Link Budget Digital Format Conversion

Development and impact analysis of a LBDF Conversion Tool for Link Budget Analysis: Application to ARRAKIHS

Arthur Van Der Steichel

Delft University of Technology

TU Delft

Link Budget Digital Format Conversion

Development and impact analysis of a LBDF Conversion Tool for Link Budget Analysis: Application to ARRAKIHS

by

Arthur Van Der Steichel

to obtain the degree of Master of Science

at the Delft University of Technology,

to be defended publicly on Wednesday January 21, 2026.

Student Number:	5235227
Responsible Supervisor:	S. Speretta
Company Supervisor:	D. Jau
Thesis Committee:	S. Speretta, E. Schrama, B. Vermeersen
Project Duration:	April, 2025 - December, 2025
Faculty:	Faculty of Aerospace Engineering, Delft

Cover: Proba V, designed by Redwire Space, in orbit around Earth [1]

TU Delft REDWIRE

Preface

This thesis was written as part of Space Engineering MSc program at the Technical University of Delft and marks the completion of my studies. During this education, I have learned many invaluable lessons, both academically as personally. For this I would like to thank all the professors of the University. Especially, Stefano Speretta, who provided insights, feedback and encouragements throughout the project as my supervisor. I am also grateful for the opportunities I had to explore my interests in Space Engineering and share these experiences with my peers.

Secondly, I would like to thank Redwire Space to provide me with the opportunity to write my thesis at this company. Specifically, Dominique Jau, my company supervisor, who guided me through the project and provided me with support and insights. On top of this, I want to thank the ARRAKIHS team to include me in the project and help me out with any problems that occurred during this thesis.

Arthur Van Der Steichel
Kruibeke, January 2026

Abstract

Several satellite engineering domains already adopt a standardised digital format, however the communication subsystem still uses and outdated approach, namely link budget tables. Because of this, the concept of the Link Budget Data Format (LBDF) is introduced which is a standardised, digital format to share link budget data. This thesis presents a conversion tool to convert the satellite link budget data from Redwire Space's calculation tool to the LBDF standard. The impact of this approach is studied by taking interviews of both Redwire Space and European Space Agency (ESA) employees. These interviews show that the LBDF tool has a great impact on the time consumption of the data exchange and verification process, the error reduction due to the elimination of the manual copy-paste process and finally, cost savings. On top of this, the newly created tool provides opportunities within the company to automate their own processes such as the verification and optimisation of the link budget. A new verification tool shows a fully verified link budget of the Analysis of Resolved Remnants of Accreted galaxies as a Key Instrument for Halo Surveys (ARRAKIHS) mission, while a new optimisation tool calculates the maximum achievable data rate for different modulation and coding schemes considered for the ARRAKIHS mission. The theoretical optimal solution deemed to be 8 Phase Shift Keying (PSK) modulation with concatenated coding of Reed-Solomon and convolutional coding. Both tools use the new LBDF file as input and greatly improve the speed of these processes.

Contents

List of Figures

List of Tables

List of Symbols

The lists below describe several symbols used within the body of the document

Constants

π	Pi
c	Speed of light
k	Boltzmann constant
R_E	Earth's radius

Symbols

$\frac{C}{N_0}$	Carrier-to-noise ratio
$\frac{E_b}{N_0}$	Energy-per-bit-to-noise-density ratio
$\frac{E_s}{N_0}$	Energy-per-symbol-to-noise-density ratio
$\frac{G}{T}$	Gain-to-noise-temperature ratio
A	Area
a	Amplitude
AR_{rx}	Receiver axial ratio
AR_{tx}	Transmitter axial ratio
B	Bandwidth
f	Frequency
G_{rx}	Receiver gain
G_{tx}	Transmitter gain
h	Orbit altitude
I	Number of parity bits

k Number of bits per symbol

L_{atm} Atmospheric loss

L_{cir} Circuit loss

L_{ion} Ionospheric loss

L_{pol} Polarisation loss

L_{prop} Total propagation loss

L_p Pointing loss

N Length of encoded blocks

NF Noise figure

P_{tx} Transmitting power

r Code rate

R_b Bit rate

R_s Channel symbol rate

S Number of transmitted systematic bits

S Slant range

T_{ref} Reference temperature

T_s System noise temperature

Greek Symbols

α Roll-off factor

ϵ Elevation angle

η Efficiency

λ Wavelength

ω Angular frequency

ϕ Polarisation vector angle

ρ_{rx} Receiver circular polarisation ratio

ρ_{tx} Transmitter circular polarisation ratio

θ Signal phase

Acronyms

ACM Adaptive Coding and Modulation, 8

APSK Amplitude and Phase Shift Keying, 2

ARRAKIHS Analysis of Resolved Remnants of Accreted galaxies as a Key Instrument for Halo Surveys, ii

ASK Amplitude shift keying, 4

AWGN Additive White Gaussian Noise, 7

BER Bit Error Rate, 4

BPSK Binary Phase Shift Keying, 8

CC Convolutional Code, 49

CCSDS The Consultative Committee for Space Data Systems, 6

EIRP Effective Isotropic Radiated Power, 27

ESA European Space Agency, ii

ESOC European Space Operation Centre, 32

EUMETSAT European Organisation for the Exploitation of Meteorological Satellites, 33

EUTELSAT European Telecommunications Satellite Organization, 33

FEC Forward Error Coding, 7

FER Frame Error Rate, 11

FSK Frequency shift keying, 4

FSPL Free Space Path Loss, 37

ISS International Space Station, 3

ITU International Telecommunication Union, 21

LBDF Link Budget Data Format, ii

LDPC Low-Density Parity-Check code, 8

MBSE Model Based System Engineering, 16

OBW Occupied Bandwidth, 5

OQPSK Offset Quadrature Phase Shift Keying, 2

PAPR Peak-to-Average Power Ratio, 5

PCCC Parallel Concatenated Convolutional Code, 8

PDF Probability Density Function, 23

PDT Payload Downlink Telemetry, 22

PSK Phase Shift Keying, ii

QPSK Quadrature Phase Shift Keying, 2

RAM Random Access Memory, 12

RF Radio Frequency, 47

RFDU Radio Frequency Distribution Unit, 38

ROM Read-Only Memory, 12

RS Reed-Solomon, 45

SANA Space Assigned Numbers Authority, 16

SCCC Serial Concatenated Convolutional Code, 8

SNR Signal-to-Noise Ratio, 3

SRRC Squared Root Raised Cosine, 47

TT&C Telemetry, Tracking and Command, 22

UML Unified Modeling Language, 22

US United States, 3

VSWR Voltage Standing Wave Ratio, 39

WP Work Package, 19

XML Extensible Markup Language, 16

XSD XML Schema Definition, 16

1. Introduction

The engineering world, and specific to this thesis, the space engineering world is becoming more and more a digital world, where processes and data exchange have to be done efficiently. This means that many processes are being digitalised and automated to improve efficiency and make life easier. However, the communication subsystem of a satellite design still opts for an outdated approach for sharing link budget data. It is time for this subsystem to follow the automation trend and because of this, the LBDF concept is created by ESA.

This thesis is performed at the company Redwire Space in Kruibeke, Belgium and aims to create a conversion tool to convert the satellite link budget from the company's calculation tool to the LBDF standard and study the impact this tool has on the data exchange process. A second goal is that the output of the conversion is used to verify the satellite link budget of the Analysis of Resolved Remnants of Accreted galaxies as a Key Instrument for Halo Surveys (ARRAKIHS) mission. This is an ESA fast mission [2], which is currently being designed at Redwire Space. The last goal of this thesis is to maximise the data rate of the ARRAKIHS mission by analysing different modulation and coding schemes with the converted LBDF file as input. These objectives are defined by the three stakeholders of this thesis, namely the Technical University of Delft, Redwire Space and ESA.

This thesis starts by performing a literature review in Chapter 2 to provide the reader with the required background information and to identify any knowledge gaps that lead to the definition of the research questions of this thesis. Chapter 3 discusses how the work is split up and the methodology taken to answer the research questions. Chapter 4 explains the conversion tool and researches the impact it has on the data exchange process. Then, in Chapter 5, the ARRAKIHS link budget is verified using the results from the conversion tool as input. Next, the LBDF file is used as input for the optimisation process to identify the most suitable modulation/coding scheme for the ARRAKIHS mission in Chapter 6. Finally, in Chapter 7 the conclusions of this thesis are drawn and in Chapter 8 recommendations are given for future work related to this topic.

2. Literature review

This thesis revolves around the communication subsystem of a satellite with a main focus on creating a conversion tool for the satellite link budget data to improve the data exchange process, with the application to the ARRAKIHS mission. A short introduction of the mission is provided in Section 2.1. On top of this, the conversion tool output has other applicabilities such as being used as an input for a link budget optimisation. Before starting on this, it is important to get a good understanding of what is involved in a satellite link budget. This is explained in Section 2.2. Next, the different considered modulation types for the ARRAKIHS mission are discussed in Section 2.3. The modulation types are Quadrature Phase Shift Keying (QPSK), Offset Quadrature Phase Shift Keying (OQPSK), 8 Phase Shift Keying (PSK) and 16 Amplitude and Phase Shift Keying (APSK). Afterwards, different coding types are presented and a comparison between them is made in Section 2.4. The next section, Section 2.5, discusses the concept of the LBDF. Finally, the information gaps found during the literature review result in the definition of the research questions in Section 2.6.

2.1. ARRAKIHS mission

ARRAKIHS is an ESA mission proposed by the ARRAKIHS Mission Consortium [3]. It is a fast mission which was selected in November 2022 and is planned to be launched in the early 2030s. The goal of the mission is to investigate the accuracy of the Λ-Cold Dark Matter (ΛCDM) model compared to observations of smaller-scale structures, such as galaxy halos [4]. It will look for potential mismatches between model predictions and real life observations. These potential mismatches will determine the need of refinements or even alternative models [3]. It will achieve this by observing the number and shapes of tidal stellar structures, the shape and extent of the diffuse stellar halo and finally the abundance and locations of satellite galaxies [3]. It will analyse these across Milky Way-like galaxies beyond the Local Group, which is a cluster of over 30 galaxies including our own Milky Way [5]. To achieve these goals, a binocular telescope assembly is designed to reach ultra-low surface brightness imaging, which can be seen in Figure 2.1. It includes near-infrared and both red and blue visual bands to image nearby Milky Way-like galaxies with the focus on their faint outer regions of their halos. These contain information about on how galaxies are formed and evolve over time [3]. The instrument will be in operation for three years part of a small satellite in Sun-Synchronous Low Earth Orbit at an altitude of approximately 800 km [4].

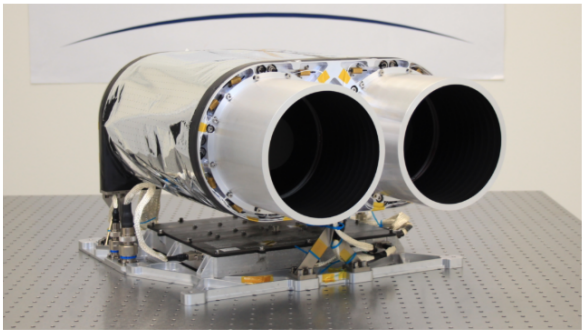

Figure 2.1: Binocular optical system developed by the Spanish micro satellites company Satlantis for the ARRAKIHS Instrument [6]

The ARRAKIHS satellite design is performed by Redwire Space, which is an American space infrastructure technology company headquartered in Jacksonville, Florida. The company has 11 facilities with eight in the United States (US) and three in Europe [7]. The design of the ARRAKIHS satellite is done at the Belgian location of the company in Kruibeke, which can be seen in Figure 2.2. This facility is focussed on the delivery of entire satellites and satellite equipment, such as on-board computers and power control and distribution units. There are two cleanrooms, one of which can be seen in Figure 2.3, and multiple laboratories where science instruments and life support systems for the International Space Station (ISS) are being build [8].

Figure 2.2: The facility of Redwire Space at Kruibeke, Belgium [8]

Figure 2.3: State-of-the-art clean room at the Kruibeke facility [9]

2.2. What is a satellite link budget?

The communication subsystem is a crucial part of a satellite which enables a reliable transfer of data between the satellite and Earth. A conceptual diagram of this link is presented in Figure 2.4. An important aspect in designing this subsystem of any space mission is the link budget. This is a quantitative assessment of the gains and losses a signal experiences as it propagates from transmitter to receiver [10]. It includes key parameters such as transmit power, antenna gains, attenuation losses, propagation losses, coding gain, the Signal-to-Noise Ratio (SNR) and others. These calculations give the engineer an estimation of how well the communication link will perform under

realistic conditions.

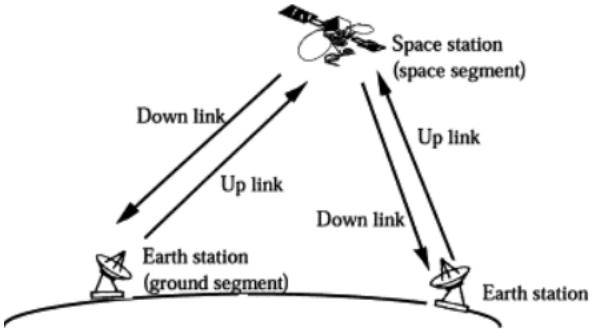

Figure 2.4: Conceptual diagram of satellite communication systems [11]

2.3. Data modulation

Sending signals directly from the satellite to Earth would create some problems such as requiring very long antennas or interference of multiple transmitters with each other [12]. To avoid this, a signal needs to be modulated onto a radio frequency carrier. This means that the bit stream is turned into an analogue signal with the following form [12]:

$$s(t) = a(t)cos(\omega t + \theta) \tag{2.1}$$

where a(t) is the amplitude, $\omega/2\pi$ the frequency and θ the signal phase. These three variables can be altered to define the modulation technique. The three basic forms that are used for transmitting digital signals are Amplitude shift keying (ASK), Frequency shift keying (FSK) and Phase Shift Keying (PSK). Where for ASK the frequency and phase remain unchanged, for FSK the amplitude and phase are not altered and for PSK the amplitude and frequency stay the same. A combination of these methods can be used for example APSK where both the amplitude and phase can change and only the frequency stays constant [12]. For the ARRAKIHS mission, multiple modulation types are considered, namely QPSK, OQPSK, 8PSK and 16APSK. A short introduction to each modulation type is provided in this section followed by a Bit Error Rate (BER) comparison between the four.

2.3.1. QPSK

The first considered modulation type is QPSK, which is the most-widely used digital modulation type [13]. QPSK contains four possible signals with a phase shift of $90°$ which is shown in the constellation map in Figure 2.5. Every signal phase represents two bits of data which means that the symbol rate is half the data rate. Each phase is represented by two bit streams, namely I and Q. When one of the components changes sign, a phase shift of $90°$ occurs, while if both components change sign, then a phase shift of $180°$ occurs. This causes additional errors due to an amplitude change

in the detected signal. This can be avoided by introducing an offset between the two bit streams I and Q, which is called OQPSK.

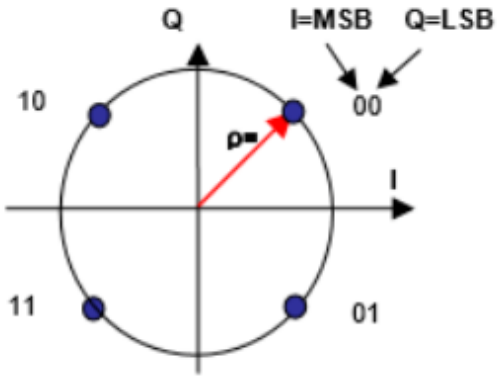

Figure 2.5: QPSK constellation map [14]

2.3.2. OQPSK

OQPSK is a variant of QPSK and differs from it by creating a delay between the odd-bit stream and the even-bit stream of half a symbol interval. This offset makes sure that during the half symbol period, either the I-signal or the Q-signal changes. This avoids the amplitude of the signal going through the ”0” point which results in minimal amplitude fluctuations because the phase never changes by $180°$ [12]. The advantage of OQPSK over QPSK is found in bandwidth limited channels where non-rectangular pulse shaping is used. The envelope fluctuations is lowered by 70% in OQPSK signals compared to QPSK [15] [16]. This means that OQPSK is a low Peak-to-Average Power Ratio (PAPR) modulation scheme, resulting in the possibility of using a non-linear power amplifier, which increases the power efficiency of the system [17].

2.3.3. 8 PSK

8PSK has four more phase states compared to QPSK and OQPSK. This means that each phase is separated by $45°$ and that each symbol contains 3 bits of information and thus that the symbol rate is a third of the data rate [18]. The advantage of 8PSK is that it can achieve a higher transmission rate compared to QPSK with the same Occupied Bandwidth (OBW). However, the disadvantage is that it requires a higher signal-to-noise ratio to obtain the same error rate as QPSK [19]. The constellation map of 8PSK can be found in Figure 2.6.

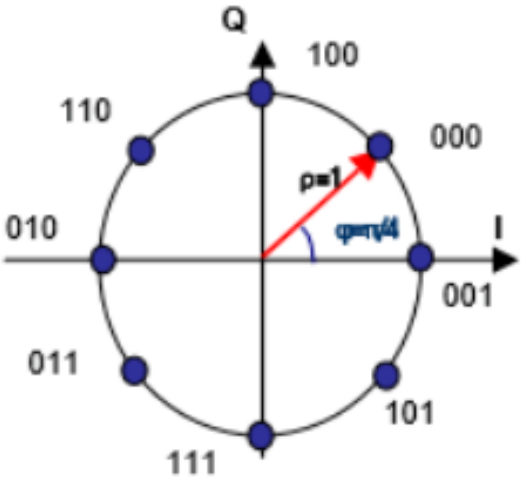

Figure 2.6: 8PSK constellation map [14]

2.3.4. 16 APSK

16APSK differs from the other three in the sense that not only the phase changes but also the amplitude of the signal. This results in an inner and outer ring with a different amplitude. Each ring contains multiple phase states where on the inner ring there are four separated by $90°$ and on the outer ring there are twelve separated by $30°$. Each symbol contains four bits of information and thus the symbol rate is one fourth of the data rate [20]. Figure 2.7 shows the constellation map of 16APSK. These rings result in amplitude variations which increases the number of errors and complexity of the modulation.

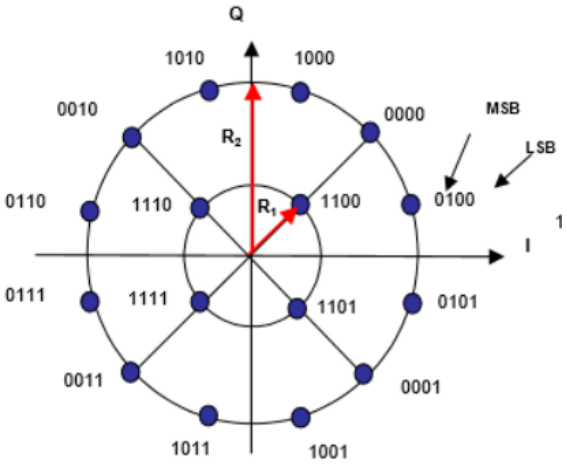

Figure 2.7: 16APSK constellation map [14]

2.3.5. BER curve comparison between modulation types

A BER curve comparison between QPSK, 8PSK and 16APSK is made by The Consultative Committee for Space Data Systems (CCSDS) [21], as can be seen in Figure 2.8.

The BER is the same for OQPSK as for QPSK and thus the graph of QPSK is a representation for both modulation types [13]. The simulations are evaluated over the Additive White Gaussian Noise (AWGN) channel. This shows that QPSK is the most reliable option of the three at low SNR. For example, at SNR of 7, QPSK has a BER value of around 10^{-2} where as 8PSK and 16APSK have a BER value of around 10^{-1}, with 8PSK slightly lower than 16APSK. At a BER of 10^{-6}, QPSK requires a $\frac{E_s}{N_0}$ of 13.35 dB, 8PSK of 18.75 dB and 21.2 dB for 16APSK. This shows again that QPSK requires a lower SNR than 8PSK who in turn also requires a lower SNR than 16APSK to achieve the same BER. However, 8PSK and 16APSK have a higher efficiency in terms of bits per symbol and thus have a higher data throughput compared to QPSK and OQPSK [21].

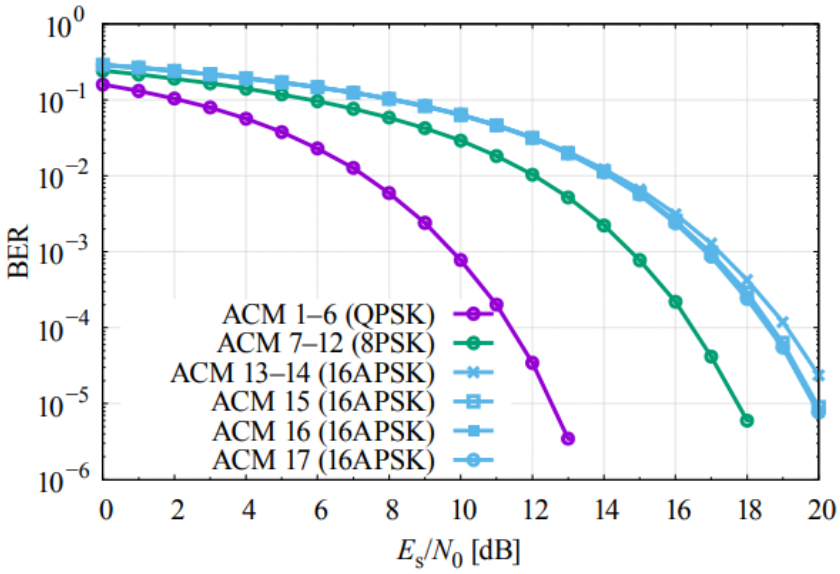

Figure 2.8: Uncoded BER on linear AWGN channel for PSK/APSK constellations [21]

2.4. Coding

During satellite communication, information can get corrupted due to noise and fading. This results in what is called bit errors [22]. A way to overcome this problem is to introduce coding. This means, adding redundant bits of information to the original data. These redundant bits are calculated from the original data using a 'code' [23]. The received message is then decoded and checked for any errors, which are repaired using the redundant bits. This process of coding in advance before sending the message is called Forward Error Coding (FEC). The advantage is thus that it detects and correct errors without the need for retransmission and it allows to operate at a lower signal-to-noise ratio. A disadvantage is that it lowers the effective data rate due to the redundant bits [22]. There are two categories of FEC, namely block codes and convolutional codes. With block codes, the message is divided in fixed-length blocks which are all encoded separately. While with convolutional codes, the data is encoded as one continuous stream of bits [23].

The following section starts by giving an introduction to Serial Concatenated Convolutional Code (SCCC) and Parallel Concatenated Convolutional Code (PCCC) in Subsection 2.4.1 and Subsection 2.4.2 respectively. This is followed by and explanation on Low-Density Parity-Check code (LDPC) and Reed-Solomon coding in Subsection 2.4.3 and Subsection 2.4.4 respectively. Afterwards, a comparison is done between the coding types itself in Subsection 2.4.5. This comparison shows why the SCCC option is preferred over PCCC and LDPC and then a comparison is made between the different considered modulation types combined with either Reed-Solomon coding or SCCC.

2.4.1. Serial Concatenated Convolutional Code (SCCC)

SCCC is a type of forward error correction coding which uses two or more convolutional codes in series. As can be seen in Figure 2.9, the information blocks are encoded by the outer convolutional encoder, followed by a fixed puncturing to a specific rate. Puncturing is a technique used to increase the code rate by selectively removing some of the parity bits produced by the encoder [24]. This punctured output shall be interleaved, which then forms the input for the inner convolutional encoder [24]. To obtain the wanted coding rate, the output of the inner encoder is punctured using two different algorithms. One is used to puncture the systematic output while the other is used for the parity output. For the upper register, systematic bits, the puncturing depends on the Adaptive Coding and Modulation (ACM) format selected. ACM involves varying the modulation and coding based on the channel conditions, user needs and system restrictions to achieve the most optimal transmission parameters. For example, with poor channel conditions, modulation techniques like Binary Phase Shift Keying (BPSK) and QPSK are preferred due to their robustness while in better conditions, 8PSK and 16APSK can be used. The coding rate can be adjusted to achieve more error correction during poor channel conditions and less during good conditions. This flexible technique makes a great trade-off between data throughput and a stable link. A disadvantage to this method is an increase in complexity and due to a delay in signal feedback, a suboptimal combination of modulation and coding can occur during fast changing channel conditions, due to weather changes for example [25].

Based on the ACM format, the number of surviving bits in each 300-bit segment after puncturing changes. The puncturing order within this 300-bit segment is fixed. The puncturing of the lower register, parity bits, is done using a rate matching algorithm. The number of parity bits deleted can be found using Equation 2.2 [24].

$$\Delta = I - (N - S - 2) \tag{2.2}$$

where I is the overall number of parity bits before puncturing, N is the length of the encoded blocks and S is the overall number of transmitted systematic bits. The punctured systematic bits followed by the punctured parity bits form the input for the row-column interleaver, which is used to pseudo-randomise the selection of bits that are allocated to one modulation symbol. The number of rows equals the length of the

encoded block (8100 symbols) and the number of columns is equal to the modulation order, which depends on the selected modulation: QPSK = 2, 8PSK = 3, 16APSK = 4, 32APSK = 5, 64APSK = 6. This process allows for a wide range of coding rates and modulation techniques [24].

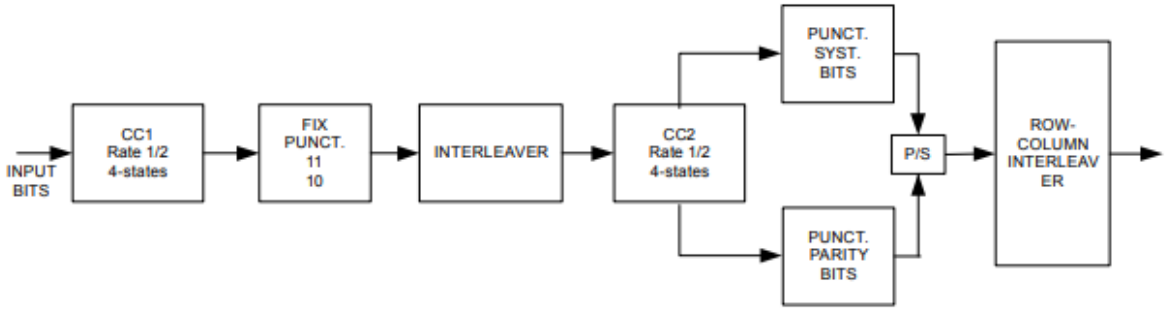

Figure 2.9: Block diagram of the SCC turbo coding scheme [24]

2.4.2. Parallel Concatenated Convolutional Code (PCCC)

PCCC is a forward error correction coding technique widely used in satellite communications and also known as 'turbo codes' [26]. It belongs to the family of concatenated codes where it combines two or more recursive systematic convolutional encoders in parallel with an interleaver in between the two [27]. The function of the interleaver is to shuffle the input data bits into the turbo encoder, which randomises burst errors and improves the correction capabilities [28]. The codeword generated consists of the parity bits (x_{k1}, x_{k2}) and the information bits (x_{ks}). A diagram of this is shown in Figure 2.10.

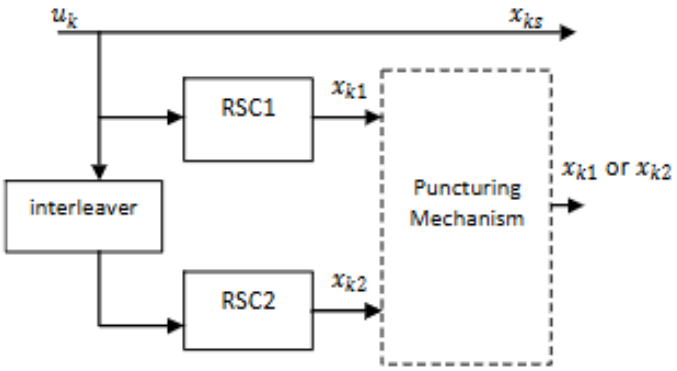

Figure 2.10: Turbo encoder block diagram [28]

The codeword output can then be punctured to change the overall code rate. The decoding process is an iterative process employing two soft-input soft-output decoders. The two decoders exchange soft information messages which represent the bit value probability. This process continues until the decoders converge [28].

2.4.3. Low-Density Parity-Check code (LDPC)

LDPC codes are a form of block codes which was proposed by Robert Gallagher in 1960 in his doctoral dissertation [29]. They were neglected for around 35 years because of their very high computational complexity [30], however they are now widely used such as in the DVB-S2 standard [31], which is a digital satellite video broadcast standard. They became more attractive in 1993 when turbo codes were introduced together with more modern hardware and technology [32]. D.McKay and R.Neal created a new class of block codes containing many features of turbo codes and this turned out to be a rediscovery of the LDPC codes introduced by Gallagher [32].

A LDPC code can be described using a graphical representation called a Tanner graph which is shown in Figure 2.11. This is a bipartite graph showing two types of nodes, namely the variables nodes ($c_0 - c_7$) which represent the symbols of the codeword, and the parity control equations are represented by the parity nodes ($f_0 - f_3$) [33].

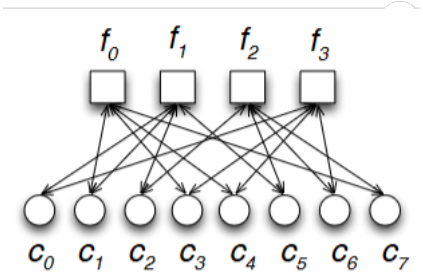

Figure 2.11: Representation of a LDPC code in a Tanner graph [34]

Another way of representing a LDPC code is with the parity check matrix H, which can be seen in Figure 2.12. The two type of nodes in the Tanner graph are connected by branches and represent the non-zero elements in the parity check matrix. The number of variable nodes indicates the number of columns in the matrix while the number of parity nodes represent the number of rows.

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

Figure 2.12: Corresponding 4×8 parity check matrix H

This is only a very small representation as in practice, the parity check matrix can have hundreds to thousands of rows and columns. This is why it is a 'low density' parity check, meaning that only a very small percentage are non-zero entries. Otherwise, the system would become too complex [33]. The main advantages of LDPC codes is that their performance is very close to the Shannon capacity [32].

2.4.4. Reed-Solomon coding

Reed-Solomon coding is an error-correction code created by Irving S. Reed and Gustave Solomon in 1960 [35]. It is a subgroup of the Bose, Chaudhuri and Hocquenghen (BCH) codes, even though their work was independent of each other. To be more specific, Reed-Solomon codes are non-binary, BCH, cyclic, linear block error correction codes [36]. A block code means that the information transmitted is split up into separate blocks of data. Every block contains a parity protection as a self-contained code word. It is thus added as a separate block without altering the original message symbols during the encoding. This can be seen in Figure 2.13. Linear means that two code words are added together to form another code word, while cyclic means that the symbols of a code word are shifted to create another code word [37].

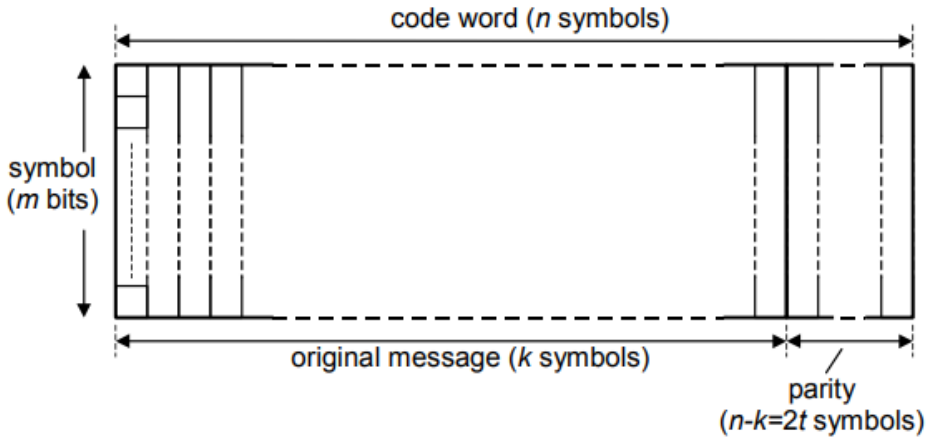

Figure 2.13: Reed-Solomon code definition [37]

2.4.5. Comparison

Now that the different coding types are introduced, a comparison can be done between them. It should be noted that Redwire Space is very familiar with Reed-Solomon as this has heritage on previous missions and this is thus definitely a contender for the ARRAKIHS mission. A comparison between SCCC, PCCC and LDPC is made to find the most suitable option of the three to contend with Reed-Solomon. SCCC, PCCC and LDPC are powerful forward error-correcting codes, but they show different performance characteristics. PCCCs generally perform better than SCCCs and LDPCs at low SNR, while SCCCs and LDPCs tend to outperform PCCCs at higher SNR. This is due to the fact that PCCC has an error floor problem, which is the flattening of BER at high SNR due to a small number of low weight codewords. This can limit the effectiveness in achieving extremely low bit error rates. An in depth comparison between the three can be found in Figure 2.14 where four characteristics are taken into account, namely performance, complexity, flexibility and maturity. For the performance and complexity criteria, a distinction between short and long block size is made. On the first row of both the short and long block size distinction in the performance section of Figure 2.14, it shows the E_b/N_0 at the target Frame Error Rate (FER) of 10^{-6} for every coding type and different code rates. The second row states the gain loss of the coding type compared to the best coding type for the three code rates. It shows that PCCC outperforms the SCCC with a gain from 0.3 to 1.05 dB (rate 1/3 to rate 9/10)

and LDPC with a gain from 0.5 to 1 dB at the same rates for the short block size, while for long block size, LDPC performs slightly better than SCCC and PCCC (gains from 0.05 to 0.25) except for PCCC at rate 9/10 due to the error floor [38]. For this latter case, the gain loss of PCCC becomes increasingly large.

Criterion		Code	PCCC - TurboΦ			SCCC			LDPC		
			1/3	5/6	9/10	1/3	5/6	9/10	1/3	5/6	9/10
Performance	Short	Eb/N0 @ Target FER	2	5,4	6,7	2,3	6	7,75	2,5	5,9	8
		Loss to best	0	0	0	0,3	0,6	1,05	0,5	0,5	>1,0
		dmin, Nmin	36-107	5-32.4	4-337	30-1	5-1	4-3	X	X	X
	Long	Eb/N0 @ Target FER	0,8	3,4	>5,0	0,95	3,4	4,35	0,75	3,25	4,1
		Loss to best	0,05	0,15	>1,0	0,2	0,15	0,25	0	0	0
		dmin, Nmin	66-1366	11-1365	7-590	66-2	7-1	7-1	X	X	X
Complexity	Short	RAM Memory	12 840	8 988	8 798	10 914	7 059	6 869	34 413	12 641	10 435
		ROM Memory	64	64	64	6 420	6 420	6 420	47 988	15 829	12 661
		SUM	1860	1730	1730	730	730	730	950	350	250
		MAX	1280	1280	1280	-	-	-	-	-	-
		MAX*	-	-	-	370	370	370	1600	900	700
		Number of iterations	8	8	8	10	10	10	50	50	50
	Long	RAM Memory	163 840	286 720	303 104	139 260	225 276	236 742	552 313	614 137	630 713
		ROM Memory	64	64	64	106 490	307 193	331 763	1 142 383	1 292 527	1 332 783
		SUM	1860	1730	1730	730	730	730	1250	550	550
		MAX	1280	1280	1280	-	-	-	-	-	-
		MAX*	-	-	-	370	370	370	2500	1600	1500
		Number of iterations	8	8	8	10	10	10	50	50	50
Other issues		Flexibility	Sensitive to interleaver optimization			High			Code must be designed for all cases		
		Maturity	High			Medium			Low		

Figure 2.14: Summary of complexity and performance comparison for the three classes of PCCC, SCCC, and LDPC [38]

When looking at the complexity, SCCC is the simplest compared to PCCC and LDPC in terms of Random Access Memory (RAM) usage while for the Read-Only Memory (ROM), PCCC requires very little and is thus superior. LDPC is by far the most complex scheme of the three. From a mathematical point of view, SCCC is two to three times less complex than PCCC and two to six times less complex than LDPC [38]. The number of iterations are similar for PCCC and SCCC with eight and ten respectively, while LDPC has five times as many iterations with 50.

The third comparison that is done in Figure 2.14 is flexibility. For this criteria, SCCC outperforms the others with PCCC close in second. SCCC can easily adapt to different code rates, modulation schemes and block sizes, which is also true for PCCC, however, it is sensitive to interleaver optimisation. LDPC is less flexible as the code

should be designed for every case [38].

The last criteria is the maturity level of the codes. PCCC easily has the highest maturity level as it is widely used before. Figure 2.14 states that LDPC has a low maturity level, however, the table was published in 2005. By now, LDPC has been used as standards in WiFi, 10GBASE-T and WiMax [39]. The maturity level of LDPC can thus be increased from low to high. SCCC remains at medium as it has not yet been implemented. An overview of the main advantages and disadvantages of the three coding types is given in Table 2.1.

Table 2.1: Advantages and Disadvantages of SCCC, PCCC, and LDPC

Code Type	**Advantages**	**Disadvantages**
SCCC	- Improved error floor over PCCC - Lower complexity compared to PCCC and LDPC	- Higher latency than PCCC - Slightly worse performance at low SNR
PCCC	- Excellent performance at low SNR - High Technology Readiness Level (TRL)	- Higher error floors - Performance degrades at high SNR
LDPC	- Excellent overall performance - Very low error floors	- High decoding complexity - Code must be designed for all cases

This comparison shows that for the ARRAKIHS mission, the SCCC option is preferred over PCCC due to a lower complexity and a better behaviour related to the error floor. Compared to the LDPC option, it is preferred due to the much lower complexity even while having a slightly worse performance.

For the ARRAKIHS mission, multiple modulation schemes are being considered at the moment. A comparison between these combined with Reed-Solomon and SCCC is presented to highlight the differences between them. The modulation schemes that are considered are QPSK, OQPSK, 8PSK and 16APSK with Reed-Solomon and SCCC coding.

In Table 2.2 a comparison for the first modulation type is done, namely QPSK, QPSK with Reed-Solomon coding and QPSK with SCCC. QPSK has a spectral efficiency of 2 bps/Hz [21], however, this decreases when coding is applied by a factor R (code rate, typical values are 1/2, 1/3, 2/3,...). If no coding is applied, then the code rate is 1 and thus the spectral efficiency remains at 2 bps/Hz. For the options with coding, the spectral efficiency decreases by a factor R and thus the occupied bandwidth increase with a factor 1/R. The envelope of the options is non-constant as during transitions, the envelope can go to zero when the amplitudes of both the in-plane and quadrature components change at the same time their polarity [40]. The option with SCCC coding

has a range for the required E_b/N_0 qs multiple code rates are link to one modulation type and thus a range of coding gains. It outperforms the options with no coding and Reed-Solomon coding as SCCC concatenates two or more convolutional codes which thus improves the error correction. A disadvantage of this is that it increases the complexity and latency of the system.

Table 2.2: Comparison between QPSK, QPSK with Reed-Solomon and QPSK with SCCC

Feature	**QPSK**	**QPSK with Reed-Solomon**	**QPSK with SCCC**
Modulation Type	Phase shift keying	Phase shift keying + Reed-Solomon	QPSK + SCCC
Spectral efficiency	2	2R	2R
Envelope Variations	Non-constant envelope	Non-constant envelope	Non-constant envelope
Error Correction	None	Strong	Excellent
Complexity	Low	Moderate	High
Required $\frac{E_b}{N_0}$ @ BER 10^{-6} [dB]	10.34	4.14	0.9-3.3
Latency	Low	Moderate	High

The difference between QPSK and OQPSK is the offset between the in-plane and quadrature component. This results in the fact that they can not change polarity at the same time which eliminates the non-constant envelope changing it to a near-constant envelope. The other considerations remain the same. This is shown in Table 2.3.

Table 2.3: Comparison between OQPSK, OQPSK with Reed-Solomon and OQPSK with SCCC

Feature	**OQPSK**	**OQPSK with Reed-Solomon**	**OQPSK with SCCC**
Modulation Type	Offset Phase shift keying	OQPSK + Reed-Solomon	OQPSK + SCCC
Spectral efficiency	2	2R	2R
Envelope Variations	Near-constant envelope	Near-constant envelope	Near-constant envelope
Error Correction	None	Strong	Excellent
Complexity	Low	Moderate	High
Required $\frac{E_b}{N_0}$ @ BER 10^{-6} [dB]	10.34	4.14	0.9-3.3
Latency	Low	Moderate	High

Table 2.4 compares 8PSK, 8PSK with Reed-Solomon and 8PSK with SCCC. The spectral efficiency of 8PSK is 3 bps/Hz [21], which is multiplied with the code rate for

the cases with coding applied, resulting in the coding options having a higher occupied bandwidth. The robustness to noise of 8PSK is lower due to a smaller phase separation compared to QPSK which makes it more error prone, which thus requires a higher E_b/N_0. However, Reed-Solomon and SCCC improve the robustness, which lowers the required E_b/N_0 again. Due to the more phases and smaller separation, the complexity increases [41]. For 8PSK, the latency is still low as there is no decoding process while for the options with coding, the latency increases due to iterative decoding.

Table 2.4: Comparison between 8PSK, 8PSK with Reed-Solomon and 8PSK with SCCC

Feature	**8PSK**	**8PSK with Reed-Solomon**	**8PSK with SCCC**
Modulation Type	8-Phase Shift Keying	8PSK + Reed-Solomon	8PSK + SCCC
Spectral efficiency	3	3R	3R
Envelope Variations	Constant envelope	Constant envelope	Constant envelope
Error correction	None	Strong	Excellent
Required $\frac{E_b}{N_0}$ @ BER 10^{-6} [dB]	13.98	7.78	2.41-6.69
Complexity	Low-Moderate	Moderate-High	High-Very High
Latency	Low	Moderate	High

Finally, Table 2.5 makes a comparison between 16APSK, 16APSK with Reed-Solomon coding and 16APSK with SCCC. The spectral efficiency of 16APSK is 4 bps/Hz [21], which decreases due to the code rate for the coding cases. The 16APSK constellation consists of two concentric rings which results in larger envelope variations [42]. Due to this constellation, the robustness to noise decreases compared to the other considered modulation schemes and the complexity increases.

Table 2.5: Comparison between 16APSK, 16APSK with Reed-Solomon and 16APSK with SCCC

Feature	**16APSK**	**16APSK with Reed-Solomon**	**16APSK with SCCC**
Modulation Type	16- Amplitude and Phase Shift Keying	16APSK + Reed-Solomon	16APSK + SCCC
Spectral efficiency	4	4R	4R
Envelope Variations	Amplitude variations	Amplitude variations	Amplitude variations
Error correction	None	Strong	Excellent
Required $\frac{E_b}{N_0}$ @ BER 10^{-6} [dB]	15.18	8.98	4.45-7.58
Complexity	Moderate	High	Very High
Latency	Low	Moderate	High

These comparisons show that the more phases a modulation type has, the less robust they are to noise and thus a higher E_b/N_0 is required. For this, the coding comes into play which lowers the required E_b/N_0 at the cost of decreasing the spectral efficiency and increasing the complexity and latency of the system. SCCC has a higher coding gain compared to Reed-Solomon but also a higher complexity and latency. The advantage of the more phases of the modulation is having a higher spectral efficiency and thus the capability of having a higher data rate for a certain given bandwidth, however this then also increases the complexity of the system. There is no clear ideal option and thus a trade-off is required which will be done later on this thesis.

2.5. LBDF concept

The satellite link budget data is till this day shared using link budget tables. The first three pages of an empty CCSDS satellite link budget table [43] are shown in Appendix A. This is the official way to share the link budget between ESA and contractors. However, this is an outdated method as it is very inefficient and error prone because the values have to be manually copy-pasted to and from the link budget tables. Therefore, the idea of the LBDF is created, which is to introduce a standardised Extensible Markup Language (XML) format for exchanging satellite link budget data between ESA and contractors. A diagram of the concept is shown in Figure 2.15. The structure of the standardised XML files is defined by XML Schema Definition (XSD) files. These files define which parameters are required or optional, the order of the parameters and the attributes required. The XSD files are set up by ESA according to their expectations on how they want and which data they want to receive. They are made available online in the Space Assigned Numbers Authority (SANA) registry [44]. This method is a Model Based System Engineering (MBSE) approach with the aim of creating a more efficient and accurate way of sharing data between ESA and contractors. MBSE is a method which uses engineering models and digital systems as primary way of exchanging information [45].

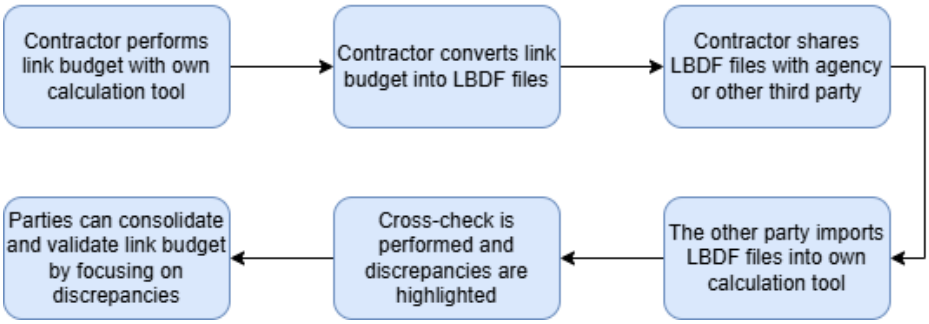

Figure 2.15: Diagram of LBDF concept in use

The ARRAKIHS mission is one of the first for which the LBDF concept will be used in practice[1]. This shows that the concept is still in an early experimental stage and the impact it has on the data exchange process is still unknown.

2.6. Research questions

The idea of introducing a standardised way of sharing satellite link budget design data was introduced by ESA. However, it is not in practical use yet, which means that the impact this new standard has on the data exchange process is still unknown. This will thus be the first goal of this thesis to create a conversion tool which converts the satellite link budget design data from the Excel file of the contractor to the LBDF standardised format for exchange and then study the impact this tool has on the data exchange process. This results in the first research question to study the impact of the LBDF approach. This can be split up in multiple sub-questions where the first one focusses on the time saved by using the tool. The idea is to automate the process more which is expected to result in saving time spent on the data exchange and verification process. The goal of the first sub-question is to identify if this is actually the case and how much exactly. The second sub-question is related to an error reduction due to the elimination of the manual copy-paste process. In the old way of sharing the link budget data, the information was copy pasted into and from link budget tables. This process often came with manual errors. The goal of the second question is thus to identify if the tool has reduced these errors. These time savings and error reductions can lead to cost savings as engineers have to spend less time on the entire process. This is always beneficial for the companies the engineers work for. The final two sub-questions are to identify the operability and adaptability of the tool to see how easy it is for the engineers to work and edit the tool. These sub-questions will answer the overall research question to study the impact the tool has on the data exchange process and the company itself. A second research question is raised with a practical use to the ARRAKIHS mission to find the most suitable modulation and coding scheme. As identified during the literature study, many combinations are a possibility for the ARRAKIHS mission and thus a detailed study is required to find the most suitable one. This is first done by looking for the coding gain of the considered coding types, resulting in the first sub-question. Secondly, the maximum data rate should be calculated using an optimisation process for every considered option, which is impor-

[1] Information obtained due to verbal communication with ESA engineer

tant to assess the performance of each option. Thirdly, the cost is an important factor which should be investigated and finally the practical availability of the combinations is required. These aspects can then be used to find the most suitable modulation and coding scheme for the ARRAKIHS mission.

1. What is the impact of the model based system engineering approach of the LBDF?
   (a) What is the time saved using the tool compared to the outdated method?
   (b) What is the error reduction due to the elimination of the manual copy-paste process of the data to and from the link budget tables?
   (c) What is the cost reduction that can be achieved with the LBDF?
   (d) How is the operability of the tool?
   (e) What is the adaptability of the tool?
2. What is the most suitable modulation and coding scheme for the ARRAKIHS mission?
   (a) What is the coding gain of the considered coding types SCCC, Reed-Solomon and concatenated coding?
   (b) What is the maximum data rate each configuration can achieve?
   (c) What is the cost difference between the coding types?
   (d) What is the availability of the modulation and coding schemes in practice?

3. Methodology

To tackle this thesis project, the work is split up in five main Work Packages (WPs) with the addition of two work packages related to the writing and finalisation of the thesis itself. A Gantt Chart is presented in Appendix B to provide a visualisation of the timeline of this thesis. This Gantt Chart has been updated throughout the project as some timings differed from the initial plan and opportunities, such as the optimisation tool, were thought of during the project itself. This addition was thus made while working on this thesis.

WP 1: Literature review:
The literature review provides the reader with a fundamental understanding of the LBDF concept and the satellite link budget with the focus on modulation and coding. The goal of the literature review is to define the research questions of this thesis by identifying the knowledge gaps.

WP 2: Conversion tool
During the literature review, it is identified that the LBDF concept is not in practical use yet and thus the impact is still unknown. To tackle this problem, it starts by creating a conversion tool to convert the satellite link budget data from the calculation tool from Redwire Space to the LBDF standard. This tool is created in Python. The idea is to define the required path in the LBDF file for every parameter and search for the parameter name in the Redwire Space calculation tool. Once the parameter is found, it extracts the corresponding information and uses the defined path to write the information in the correct place in the LBDF file. A more detailed explanation is given in Chapter 4.

WP 3: Conversion tool impact assessment
The third WP focusses on the impact of the tool to the data exchange process, which is studied by taking interviews of Redwire Space and ESA employees. The first step to these interviews is to create a set of questions that cover every aspect on the impact the tool can have. This means taking into account the advantages but also the disadvantages. But before going into these (dis)advantages, a better understanding of the old method is required. This is crucial to know in order to make an impact assessment. And thus the first set of questions identified is related to getting a better understanding of how the link budget data was shared previously:

1. How was the satellite link budget data typically shared before LBDF?
2. How long did this process take?
3. Were there any issues with inconsistencies, such as missing parameters, inconsistent units,…?
4. How often did errors occur during this process and what was their impact?

The second set of questions is related to the exact impact the tool itself has. It focusses

on the time consumption, the errors made, inconsistencies and other advantages and disadvantages of the new LBDF process.

5. How long does the data exchange and verification process take using the LBDF?
6. Has the LBDF tool reduced the number of manual errors due to the elimination of the copy-paste process?
7. Has the LBDF tool reduced misunderstandings, unclarities and inconsistencies?
8. Has the LBDF tool made it easier to verify the satellite link budget?
9. Has the LBDF tool introduced any unexpected problems that did not exist with the old method?
10. Are there any other advantages to this method of data exchange?

The final set of questions are related to the tool usage itself. These questions are directed towards the Redwire Space employees as they will be using this tool in the future for other missions. The set focusses on the operability of the tool, the limitations or bugs and the possible improvements that can be made.

11. How user-friendly do you find the LBDF tool?
12. Is training or additional documentation required to use the LBDF tool properly?
13. Have you encountered any limitations or bugs?
14. Do you see any improvements that can be made in the future?

The first ten questions are thus meant for both Redwire Space and ESA employees, while the last four are only for the Redwire Space employees. The questions will either be send out by e-mail and thus the questionnaire will be performed in writing or an actual interview will be set up. This interview can either be face-to-face or via a Teams meeting. This will depend on the time availability of the interviewees.

The results of the questions can then be used to perform a cost saving analysis. This is done by looking for the salary of an ESA employee and multiplying the hourly rate with the time saved due to the tool. This results in the costs saved during the design of one mission and it can then be estimated how much could have been saved in the past by using the tool.

WP 4: Verification tool

The impact of the tool does not end there. It creates the opportunity to make other processes within the company more efficient. A first example of this is the verification process. The generated LBDF file can be used as input for a newly created verification tool. This tool can extract the required values to recalculate the entire satellite link budget and compare the values to the original calculated ones. When there is a mismatch between the original and recalculated signal-to-noise ratio, all the intermediate results can be printed which provides the engineer with a clear overview. This makes it easier to see during which step a mistake has been made and fix it. A more

detailed explanation is provided in Chapter 5.

WP 5: Optimisation tool
A second example of making processes easier is a link budget optimisation tool. This will focus on maximising the data rate, while respecting the bandwidth and SNR constraints, for the different considered modulation and coding schemes of the ARRAKIHS mission to find the optimal configuration. The constraints for the optimisation tool are requirements set up by ESA for the ARRAKIHS mission and they are defined in Table 3.1. The requirement relating to the maximum downlink occupied bandwidth is derived from International Telecommunication Union (ITU) regulations [46], while the link margin requirement, which is the difference between the received and required $\frac{E_b}{N_0}$, is derived from the 'Model philosophy for science assessment studies' document created by ESA [47].

Table 3.1: ARRAKIHS requirements related to the maximum downlink occupied bandwidth and minimum link margin set by ESA

Req ID	Description
ARK-REQ-1	The maximum downlink occupied bandwidth (99% of power) shall not exceed 10 MHz.
ARK-REQ-2	Link budgets for all phases shall be computed with a nominal link margin of at least 3 dB.

Each modulation and coding combination has its own required $\frac{E_b}{N_0}$, code rate and number of bits per symbol. This should firstly be defined for every considered option. On top of this, other required values such as gains and losses can easily be extracted from the LBDF file. The idea is then to loop through a range of bandwidths and calculate the data rate using the code rate and number of bits per symbol. The occupied bandwidth requirement of ARRAKIHS is set to a maximum of 10 MHz and thus this value is used as the upper limit of the range. The data rate, together with the defined and extracted values, can then be used to calculate the link margin and as long as this value stays above the threshold of 3 dB, the bandwidth, data rate and achieved link margin are stored. This loop keeps running until either the link margin becomes lower than 3 dB or the occupied bandwidth reaches the limit of 10 MHz. This process will be applied to every defined option. A more detailed explanation is given in Chapter 6.

WP 6: Thesis writing
This work package focusses on documenting all the findings in a single document. This will be done throughout the project and extra focus will be allocated towards the end. This will result in a draft version of the thesis.

WP 7: Thesis finalisation
The last work package focusses on implementing the feedback received on the draft version, preparing the final presentation and handing in the final thesis.

4. Conversion tool

The new way of sharing the satellite link budget data will be through the LBDF. However, the link budget at Redwire Space is still done in Excel and thus a conversion tool is required to convert the information from Excel to LBDF. This will be presented in this chapter, starting with the purpose of the tool in Section 4.1. The next section, Section 4.2, explains how the tool is set up, followed by an explanation on how to use the tool in Section 4.3. Afterwards, the tool constraints and validation are discussed in Section 4.4 and Section 4.5 respectively. Finally the impact the tool has made on the data exchange process is detailed in Section 4.6.

4.1. Purpose

During ESA missions, several engineering domains, such as Thermal, Structure and Radiation adopt standardised and shareable digital formats and mathematical models. This is part of a broader MBSE approach, with the objective to quickly and efficiently exchange engineering information between ESA and industry. However, Telemetry, Tracking and Command (TT&C) and Payload Downlink Telemetry (PDT) engineering still today adopts an outdated approach, namely the sharing of documents with link budget tables. Although this approach guarantees an independent cross-check of analyses by ESA and by the industry, it has the drawback that engineers must import all data manually, thus being subject to a very inefficient process with potential manual errors.

The aim of LBDF is to implement a standardised and shareable digital format for link budgets allowing efficient engineering iterations between ESA and companies and among companies themselves in the field of Space Communication Subsystem. For this, a conversion tool needs to be created to convert the information from the link budget calculation tool to the standardised and shareable digital format. In the case of Redwire Space, the link budget calculation tool is based on Microsoft Excel and thus a conversion tool from Excel to LBDF standards needs to be created. The LBDF file is in XML, which means that the tool should convert the link budget data from Excel to XML format.

4.2. Tool set-up

The first step to create the conversion tool is to compare the XSD schema files provided by ESA with the link budget calculation tool of Redwire Space to see if all the required parameters by ESA are in the Redwire Space calculation tool and if the parameter names match. Missing parameters are included in the Redwire link budget and some parameter names are updated to the matching ESA equivalent. The structure of the XSD files is presented in a diagram in Unified Modeling Language (UML) format, of which an example is shown in Figure 4.1. This is only a small part of the structure, namely of the ground segment. The full structure can be found in Appendix C. Pa-

rameters marked with a '+' are considered as mandatory, while parameters with a '-' are optional. Every parameter has a certain data type assigned which indicates which values and unit are expected. This entire structure is used to check the Redwire link budget for completeness of the required values and correctness of the units and parameter names.

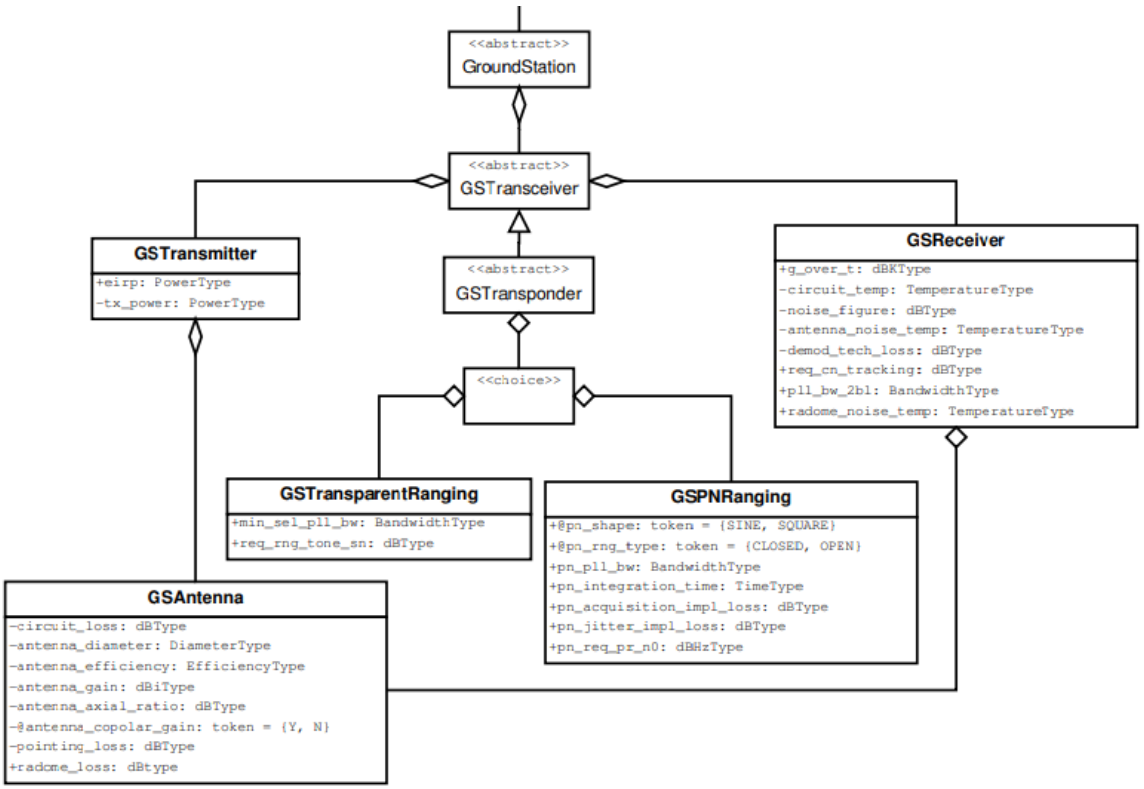

Figure 4.1: Ground segment LBDF structure in UML format [48]

To clarify how the checked and complete Redwire Space calculation tool looks like, an example is shown in Figure 4.2. This example shows the gain, pointing loss, axial ratio and the noise temperature of the spacecraft during uplink. These values are dummy values and thus does not related to any ongoing mission. The first column of the calculation tool shows the parameter names and the second column shows the unit in which the corresponding parameter is expressed. The next three columns show the nominal, adverse and favour value of the parameter. The nominal value is the design value of the parameter which serves as the reference point. However, in reality, this value might differ slightly and that is where the adverse and favour value come into play. The adverse value represent a slightly worse value compared to the nominal one, whereas the favour value represents a slightly better one compared to the nominal value. The next column shows the mean value of the parameter, which is the calculated average value of the nominal, adverse and favour value. The second to last column is the variance which represents the spread of the data values around the mean. Finally, the last column indicates the type of Probability Density Function (PDF) used for the corresponding parameter. The PDF defines the probability of a

parameter value falling within a specified range. Examples of a PDF are a Gaussian distribution (GAU), Uniform distribution (UNI) and in the case of the gain and pointing loss in the example: Triangular distribution (TRI) [49]. This completes the full lay-out of the Redwire Space calculation tool on which the conversion tool is based. This Excel file thus forms the template for the future missions designed by Redwire Space as well.

SPACECRAFT RX	Unit	NOM	ADV	FAV	MEAN	VAR	PDF
GAIN	dBi	2.00	0.50	3.50	2.00	0.38	TRI
POINTING LOSS SC	dB	0.00	0.00	0.00	0.00	0.00	TRI
AXIAL RATIO	dB	8.50	10.50	6.50			
NOISE TEMPERATURE	K	180.00	195.00	165.00			

Figure 4.2: Snapshot of a section of the calculation tool of Redwire Space

Once all the checks are done, the next step is to create the tool. This tool consists of three parts. The first part is the defining of different XML writing functions. As mentioned before, parameters have required values based on the assigned data type. However, these required values differ from parameter to parameter. Some require only one value, while others need a 'nominal', 'adverse' and 'favour' value. Other parameters need specific attributes, for example 'telemetry coding' requires the type of coding and then depending on the type of coding, it needs extra attributes related to this specific type of coding, such as the code rate for example. Because of this, multiple functions are defined such that every parameter can be written into the XML file as expected by ESA. Once it is defined which information should be read out of the Excel file, then it can be written into the XML file using the xml.etree.ElementTree package. This package can be used to parse and create XML files [50]. It was originally tried using the package 'Yattag' [51]. However, it was found using this package that not for every parameter the required format was achievable. This is when other options were explored and the xml.etree.ElementTree package was found and tried. This package is part of the standard Python library since Python 2.5 [52] and it turned out to be more successful. The second part is the definition of the parameter path where for every mandatory parameter, the XML path is defined. This makes sure that every parameter is written under the intended parent tag. In the same step, the correct writing function from step one is assigned to every parameter so that the required information is written into the XML file. The final part is the main part were everything comes together. The input from the user is asked such as the file name of the file that needs to be converted, the type of conversion, the Excel sheet that needs to be converted and the title for the output LBDF file. A more detailed explanation will be given in Section 4.3. The next step in the main part is to look for every parameter in the Excel file and write the corresponding information in the XML file under the correct path. A problem with this is that for a TT&C link budget both uplink and downlink is required. However, some parameter names occur in both which resulted in uplink values placed in the downlink section and vice versa. To resolve this, a function is created which sets boundaries for the search area for parameters in the Excel file. The tool thus automatically sets a search grid for the uplink parameters and one for the downlink parameters. It then looks for the correct parameter name, reads out the corresponding information and writes it in the XML file under the defined path. Once the conversion is completed, the tool opens both the XSD files and the converted XML file and validates the XML

against the XSD. The tool goes through the generated XML file and compares it to the XSD file to see if the generated file is first of all complete and secondly, that no errors are made. Possible errors could be the use of a wrong unit, parameters not structured in the way it is supposed to be, a value given for a parameter which is not considered possible. An example of the latter is for the option 'Convolutional coding', predefined code rate options are set to be '1/2, 2/3, 3/4, 5/6, 7/8'. When a different code rate is provided, then an error will be raised. These occurring errors are written in a separate text file, which will be created in the same folder as where the conversion tool is saved. This file can then be checked and used to fix the errors. This is the internal validation process which is further explained with a test example in Section 4.5. Once all the errors are fixed, the final converted LBDF file can be found in a separate folder which will automatically be created. A part of the final result is shown in Figure 4.3. This snapshot of the converted file corresponds with the example given of the calculation tool in Figure 4.2, and thus includes dummy values. A full example of a converted LBDF file can be found in Appendix D.

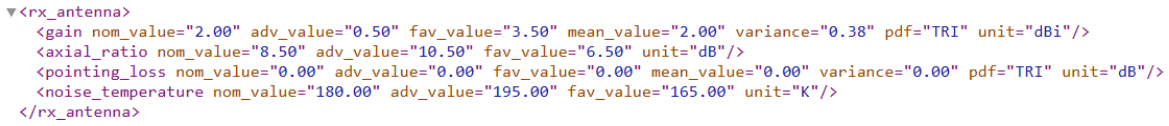

Figure 4.3: Snapshot of a section of the converted LBDF file

4.3. Tool usage

In order to use the tool, the three parts described in Section 4.2 should be placed in the same folder as the XSD files provided by ESA and together with the Excel file that needs to be converted. The next step is to run the main part of the tool which will then ask for the user input. An example of the process is shown in Figure 4.4. The first question is to fill in the name of the file that needs to be converted, which in the case of the example is 'S-Band'. It is not necessary to include the extension of the file such as '.xlsx' for example. Next, is to choose the type of conversion. The options are: PDT, Uplink and TT&C. The S-Band file consists of both an uplink and downlink and thus for this reason, it is opted for the third choice: TT&C. Thirdly, the correct Excel sheet needs to be chosen. The Excel file consists of multiple sheets and their titles are displayed. In this case, there are two relevant sheets, namely the third sheet 'Link budget - H_DL' and the sixth sheet 'Link budget - L_DL'. These are the files containing the link budget data with the first one having a high downlink data rate and the second one a low downlink data rate. The one that needs to be converted can thus be selected during this step, where in the example, the third sheet 'Link budget - H_DL' is chosen. Finally, a title should be chosen for the converted file under which it should be saved. Also here, no extension (.xml) is required as this is added automatically. Afterwards, the conversion is performed and automatically validated against the XSD files. Any occurring errors are printed in a separate text file which can then easily be checked and the errors can be fixed manually. An example of this is provided in Section 4.5. The same process can then be run again until no more errors occur and the converted file is valid against the schema files.

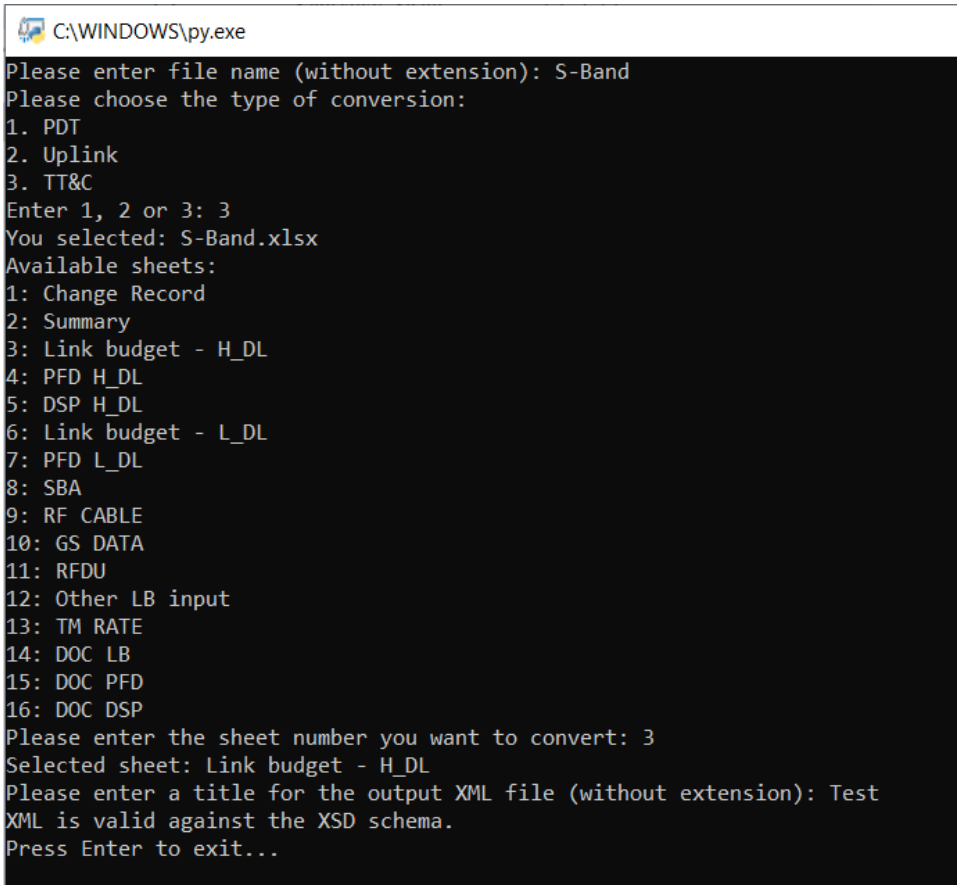

Figure 4.4: Conversion tool example

New parameters can easily be added by firstly checking the LBDF structure to find the exact required parameter name, unit and values. Secondly, the parameter can be added to the Excel file following the format of the other parameters and finally the parameters path and writing function have to be defined in the conversion tool to make sure the correct values are being read out and written in the XML file.

4.4. Tool constraints

The tool has a number of constraints that should be followed by the user in order for it to work properly. The first one is for parameter names that occur twice in the same link budget such as the 'pointing loss' of the ground station and the one of the spacecraft. When no distinction was made between the two in the beginning, the tool placed the value of the ground station pointing loss at both the ground station and the spacecraft pointing loss in the LBDF file. To solve this, a two letter marker was added to the parameter name in the Excel file. The markers are 'GS' for 'Ground Station' and 'SC' for 'Spacecraft', which are recognised by the tool and it can then make a distinction between the two parameters and place them under the correct path in the LBDF file. These markers should thus always be used for parameter names that occur twice within the same link budget.

A second constraint is that no new columns can be added without adapting the tools

code. As mentioned in Section 4.2, parameters require different values and attributes and for this, different functions are defined. This ends up in reading out the information in a fixed way with the parameter name in the first column, the unit in the second column followed by the nominal value, adverse value and favour value for most parameters. If a new column is added in between here, then the wrong information will be read out and written in the XML file. On the contrary, unlimited rows can be added to the Link Budget Excel file without having to make adaptations to the tools code.

The third constraint is that for a TT&C Link Budget, the Excel file requires an 'uplink' and 'downlink' separation with both sections marked with the title 'uplink' and 'downlink' respectively. The uplink and downlink contain a lot of the same parameters, but the values are off course different, such as the frequency, transmitter power, Effective Isotropic Radiated Power (EIRP), etc. When no distinction was made, the values of the uplink parameters were placed in the downlink section of the LBDF file. To avoid this, the section titles were introduced to create the parameter name search area. This way it can make a distinction between up- and downlink where many parameter names are the same.

4.5. Tool validation

An important process, once the tool is created, is to properly validate it. It should be checked if the tool works as intended; which means that all the required parameters should be correctly and completely be converted from the calculation tool to the LBDF file. For this, the conversion tool has two validation processes. The first one is, as mentioned in Section 4.2, an internal validation. Once the tool is run, it converts the file and automatically validates the converted LBDF file against the XSD files. When the tool recognises a difference between the two, an error is raised which is written in a separate text file and can then easily be solved by the user until the converted file is valid against the schema files. A snapshot of the text file containing the validation errors is shown in Figure 4.5. Every single error is always written on a a new line, which means that in this example, six separate errors are raised. It first specifies which element is incorrect and then secondly, it states the actual error. It can be seen that the first error is related to the element 'circuit_temp'. The error says that this element is not expected because it is expecting the element 'g_over_t'. This means that either the element 'g_over_t' is missing or the order of the elements in the converted file is incorrect. In this case the 'g_over_t' was missing. Once it was added, the error was solved. The second error is related to the element 'eirp'. The error states that the attribute 'value' is not allowed. This is because in the XSD files, it is defined that the element 'eirp' requires the attributes 'nom_value', 'adv_value' and 'fav_value'. However, these attributes are missing as only one value for the element 'eirp' is given under the incorrect attribute name 'value'. This thus raises the errors that the attribute 'value' is not allowed and that the attributes 'nom_value', 'adv_value' and 'fav_value' are missing. These errors are shown on line two, three, four and five of the text file respectively. The final error concerns the element 'tx_power', which is the transmitter power. It shows that the unit is expressed in dBW, however it is expected to be in

dBm. Once all these errors are resolved, the file is ready to be sent to ESA for an external validation check.

```
Element '{https://www.esa.int/LBDF}circuit_temp': This element is not expected. Expected is ( {https://www.esa.int/LBDF}g_over_t ).
Element '{https://www.esa.int/LBDF}eirp', attribute 'value': The attribute 'value' is not allowed.
Element '{https://www.esa.int/LBDF}eirp': The attribute 'nom_value' is required but missing.
Element '{https://www.esa.int/LBDF}eirp': The attribute 'adv_value' is required but missing.
Element '{https://www.esa.int/LBDF}eirp': The attribute 'fav_value' is required but missing.
Element '{https://www.esa.int/LBDF}tx_power', attribute 'unit': The value 'dBW' does not match the fixed value constraint 'dBm'.
```

Figure 4.5: Example of internal validation errors

The second validation is performed by ESA. The converted file, once internally valid, is sent to the ESA team working on the ARRAKIHS mission on their side. They import the file into their own calculation tool to verify the link budget. This is thus mainly to check the values itself of the link budget, however, it also serves as a first round of validation as well. If a required parameter is missing, it will raise an error on their side which can then be communicated to Redwire Space. On top of this, they generate their own LBDF file which is then used for a comparison with the one provided by Redwire Space. This is done using a tool that highlights any differences between the two files. This thus shows the differences between units, order of parameters, missing parameters and the actual values. During the validation of the X-band downlink link budget, an error was noticed where the 'antenna efficiency' was expressed as a value between 0-100 by Redwire Space. In this case as 60%. However, ESA expected this as a value between 0-1 and thus for this case as 0.6. This resulted in an antenna efficiency multiplied by a factor 100 in their calculation tool, which became clear during the verification of the link budget. Secondly, this difference is then also highlighted using the comparison tool to check the differences between the generated LBDF file from Redwire Space and ESA. Thus due to this external validation, the error was quickly communicated between the two parties and solved.

4.6. Tool impact assessment

After the tool is created and validated, an impact assessment is done. The goal of this assessment is to get a better understanding of how this conversion tool makes the link budget data exchange process easier and better. This is done by taking interviews of both Redwire Space and ESA employees that come in contact with the LBDF. In total, three interviews were done of which two with Redwire Space employees and one with an ESA employee. Both Redwire Space employees have experience in the electronic engineering department, which involves the power, command & data handling and communication subsystem. The interviewee of ESA is an experienced engineer with the focus on the communication subsystem of a satellite. The answers of these interviews are shown in Subsection 4.6.1. The conclusions drawn from these answers are discussed in Subsection 4.6.2.

4.6.1. Questionnaire impact assessment

This section is split up in three parts where the first set of questions is related to how the link budget data was shared before the introduction of the LBDF. This is done

to get a better overview of the changes it creates. The second set of questions is to evaluate the exact impact of the tool with relation to time consumption, errors made, inconsistencies and other advantages and disadvantages. The third and last set is specific for Redwire Space alone as it focusses on the use of the conversion tool itself. This results in a set of fourteen questions which are shown below, together with a combination of the answers of the interviewees.

Previously:

1. How was the satellite link budget data typically shared before LBDF?

There was not one fixed method. The most common methods were by putting snapshots of the Contractors calculation tool directly in the technical report, which would be shared during reviews, or by sending these snapshots by email. Another method is by sharing the full calculation tool with ESA. In the case of Redwire Space, this would be by sharing the full Excel file.

2. How long did this process take?

Creating a link budget analysis is a long, iterative process. During reviews, the results of this process are shared between contractor and ESA in the way described above. ESA then cross checks the link budget by manually copy pasting the values from the snapshots into their own calculation tool. The results are then compared to the ones provided by the contractor to verify that there are no mistakes. Contractors have their own calculation tool and they all differ from each other, and because parameter definitions and naming is not standard defined, it sometimes becomes hard to even locate some parameters. When the link budget of a new contractor needs to be verified, it takes around 30-40 minutes per link budget to perform the full verification [1]. There are at least 5-6 link budgets per major review that need to be checked and there are at least 6 major reviews during the design phase [1]. This means that at the very minimum, there are 30-36 link budgets that need to be verified during a mission design phase. Taking into account 40 minutes for one link budget, an engineer usually spends at least 24 hours on verifying link budgets. This process is thus time consuming.

3. Were there any issues with inconsistencies, such as missing parameters, inconsistent units,…?

As mentioned above, parameter definitions is not standard defined and thus every contractor can have a different approach to calculate some parameters. One contractor could include a certain loss into the calculation of parameter A, while another contractor includes this same loss in parameter B. In the end, it comes down to the same final

[1] Data obtained through communication with ESA employee

result but in the steps between, there is a difference. This can lead to confusion and misunderstandings. Taking into account the tens of link budgets that are shared between contractor and ESA during the lifetime of a mission, these misunderstandings add a lot to the time wasted due to this process. On top of this, some parameters can be expressed by different units, for example transmitting power can be expressed in dBm or dBW. These inconsistencies between contractors itself and ESA can result in more misunderstandings or errors in the calculations.

4. How often did errors occur during this process and what was the impact of these errors?

No exact data has been collected in the past on the number of errors, however, it became clear from the interviews that errors occurred quite often to even every time. These were considered part of the process. The first step when it is realised that the results from ESA's cross check and the contractors results do not match, is to start the process from scratch. The values are copy pasted again into the calculation tool to make sure that no error was made during this process. If there is still a mismatch between the results, then extra discussions were required between ESA and the contractor for any clarifications and ultimately finding the mistake. This again shows how time consuming this process was, especially when a mistake happened which is very likely.

Tool Impact:

5. Has the LBDF tool improved the speed of data exchange?

Yes, greatly! The importation process is now fully automated. No more values need to be copy pasted manually which saves a lot of time. As mentioned before, the old method was time consuming where at least 24 hours was spend on the verification of the link budget for every mission. While with the LBDF tool, one verification can be performed in a couple of minutes [1]. This results in a time spend of 1.8 hours on the verification using the LBDF when taking 3 minutes into account per link budget. This results in a time gain of at least 22.2 hours for every satellite design, which is a reduction of 92.5%. Secondly, ESA have a comparison tool, which compares the LBDF provided by the contractor with the LBDF generated by themselves. It immediately highlights the differences in assumptions and results. This makes the process of identifying errors and inconsistencies much easier and faster.

6. Has the LBDF tool reduced errors?

Yes! The elimination of the manual copy paste process reduces the number of errors during the cross checking process. It does not reduce the possible errors made in the calculations itself off course, however it makes it much easier to identify them as

the cross checking process goes way faster and the comparison tool highlights any differences between the generated LBDF files from the contractor and ESA.

7. Has the LBDF tool reduced misunderstandings, unclarities and inconsistencies as the required format and parameters are precisely defined?

Yes! Each parameter name is clearly defined and what the parameter represent. These definitions need to be adopted by the contractor into their own calculation tools. By creating this same standard for everyone, many of the misunderstandings are eliminated. It is now also clearly defined for every parameter in which unit they need to be expressed which reduces many inconsistencies as well.

8. Has the LBDF tool made it easier to verify the satellite link budget?

Yes, very much. This was part of the goal of the LBDF standard. The process is now automated which makes it much easier to fully verify the satellite link budget. As mentioned before, ESA can now import the generated XML file into their calculation tool to verify the link budget. They also have a second tool that compares the LBDF provided by the contractor with their own and it highlights the differences. This makes it easier to see in which parameter calculation an error could have been made when the link budget is not verified. On the side of Redwire Space, the creation of this conversion tool became a great opportunity to also create a verification tool. The generated LBDF is used as input for this from which the parameter values are extracted. These values are then used to calculate the link budget again and the final results are compared to the results from the original calculation tool. A more detailed explanation is given in Chapter 5. So also on the contractor's side, the tool created the opportunity to automate the verification process resulting in an easier and less time-consuming process.

9. Has the LBDF tool introduced any unexpected problems that did not exist with the old method?

So far, not yet. However, both on Redwire Space's as ESA's side, the tool and the entire process is very new. Redwire Space is one of the first to introduce this and thus will this be evaluated further in the future when the tool is used more often. At Redwire Space, the ARRAKIHS link budget calculation tool was adapted such that it matches certain standards set by ESA for the LBDF, namely some parameter names had to be changed and some units had to be converted to match LBDF standards. These changes now form the template for the satellite link budget calculation tool of all future missions at Redwire Space. During these future missions, the possibility exist that unexpected problems arise but during the tests performed for the ARRAKIHS mission, no problems were found.

10. Are there any other advantages to this method of data exchange?

Yes, multiple ones. The first one is that it creates an easier transition between the procurement phase and the operation phase of a mission lifetime, since the European Space Operation Centre (ESOC) is able to import LBDF files into the link budget tool used for operations. A second advantage is that the uniform format creates opportunities to make other processes, besides from the verification, easier or even introduce other processes. An example of this is the link budget optimisation at Redwire Space for the ARRAKIHS mission. This optimisation process focusses on maximising the data rate for multiple considered modulation/coding schemes. It uses the generated LBDF file as input. This is a new tool created as part of this thesis which is made possible by the conversion tool. A more detailed explanation on the optimisation is given in Chapter 6.

Tool usage:

11. How user-friendly do you find the tool?

The tool seems very user-friendly and easy to use. It is very intuitive on how to use the tool as the user inputs are asked very straightforward. Together with the user manual, the preparation before using the tool also becomes clear. The main preparation steps are to use the correct template for the link budget Excel file and to put all the required files into the same folder. Once this is all done, the tool is very straightforward. The tool asks clearly for the name of the file that needs to be converted, followed by the type of conversion that is required. Afterwards, the user should select the correct Excel sheet to convert and name under which the converted file should be saved. These steps are very clear and easy. Once the file is converted, it is simple to find the file in the newly created folder and if any errors occurred during the validation, they are clearly specified in a separate text file. This makes it easy to fix these inconsistencies. On top of this, the steps to take to add new parameters are clearly documented and straightforward.

12. Is training or additional documentation required to use the tool properly?

The most important document that is required is the Excel template. Without this document, the tool will not work properly. For the user itself, to use the tool, the user manual definitely helps to clarify how the tool works and what is required. Mainly for the preparation steps. A short training could provide some guidance on what is important to use the tool. This is also explained in the user manual, however an in-person explanation can clarify any doubts. Secondly, a training on how to edit the tool could be useful as understanding and editing someone else's code can be very tricky. It was proposed to include a ReadMe file which explains all the different parts of the code and how these can be edited without breaking the tool.

13. Have you encountered any limitations or bugs?

So far, not yet. The tool has been tested to convert the satellite link budget of ARRAKIHS without any problems. The tool has also been checked by another employee at Redwire Space and also here, no limitations or bugs have occurred yet. However, this does not mean that there are no limitations or bugs to the tool. The chance is great that there definitely are, but they are just not found yet and they will occur during future missions when the tool is used more often. The user manual and constraint document contains information on how to edit the tool and make changes. This document can thus be used to correctly improve the tool without breaking it.

14. Do you see any improvements that can be made in the future?

At the moment, not yet. During the performed tests, the tool worked fine and no bugs or limitations have been found. This means that the tool works fully as intended and thus no improvements are necessary at the moment. However, as mentioned before, this does not mean that there are no bugs or limitations. These will probably arise while using the tool more often and once these are found, the tool can be fixed and improved to avoid any errors in the future.

Cost saving:

As mentioned before, the LBDF tool has increased the speed of the data exchange and the verification process. This results in a reduction in cost as an engineer has to spend less time on these processes and can focus on other tasks during the time gained. The salary of an engineer at ESA with a master's degree in the Netherlands equals €4824.25 net monthly for an engineer with less than 4 years of experience while for one with 4-14 years of experience, it increases to €6163.63 net monthly [53]. This results in an hourly rate of €30.15 for a less experienced engineer and €38.52 for a more experienced one, when taking into account a 40-hour work week. These salaries are for 2025. This results in cost savings of €669.4 for a less experienced engineer and €855.2 for a more experienced engineer during a satellite design. This might sound insignificant, however, firstly, this is without any manual mistakes. As mentioned before during the questionnaire, using the outdated method, many mistakes were made during the copy-paste process resulting in having to start all over again. This is a waste of time only adding on to the extra costs saved. Secondly, taking into account the number of satellites in space at the moment, the amount that could have been saved goes drastically up. According to N2YO, ESA has 99 satellites in orbit [54]. Every single satellite requires a link budget analysis and by taking the previous calculated saving of €855.2, the tool could have saved €84664.8. However, this is only for the satellites fully designed by ESA themselves. They also have many partnerships with other organisations and companies, such as European Organisation for the Exploitation of Meteorological Satellites (EUMETSAT) and European Telecommunications Satellite Organization (EUTELSAT) for example. By taking their satellites

into account as well, the total amount that could have been saved equals €141963.2. A step further would be to make it a European standard to incorporate the tool. By summing the number of satellites of every European country and organisation, a total of 1322 European satellites is found. This would result in a cost saving of €1130574.4. This is all for past European satellites. The number of satellite launches per year is only increasing and thus the cost savings will only further increase. However, this should be put into perspective. On November 26 and 27 of 2025, ministers and high-level representatives of the 23 Member States, Associate Members and Cooperating States confirmed the budget for ESA to be 22.1 billion euros for the next three years (2026-2029) [55]. This is a record breaking budget with an increase of 32% compared to the budget of the last three years. This immense amount put into perspective how insignificant the cost saving of the conversion tool actually is. But, no matter how small the contribution, cost savings is always a positive thing and beneficial to any company or organisation.

4.6.2. Conclusion

To conclude from the interviews, each sub-question to evaluate the impact of the tool is addressed with its own conclusion.

(a) What is the time saved using the tool compared to the outdated method?

During the outdated method, it took a minimum of 24 hours to verify the link budgets during the mission design phase of one mission. While with the LBDF tool, it only takes around 1.8 hours, which results in a time reduction of 92.5%.

(b) What is the error reduction due to the elimination of the manual copy-paste process of the data to and from the link budget tables?

No exact data has been collected on how many errors were made in the past, however, the interviews made clear that errors occurred quite often to even every time. It was considered part of the process. This is fully eliminated using the LBDF tool as the import and exportation process is fully automated. No more manual mistakes can be made.

(c) What is the cost reduction that can be achieved with LBDF?

After taking the time savings and the salary of an ESA engineer into account, it was found that €855.2 can be saved during every mission. This does not take into account the potential errors made during the manual copy-paste method and having to start over again. However, with or without the errors made, the cost savings are insignificant compared to the total budget of ESA. So relatively speaking, the impact on the cost can be neglected.

(d) How is the operability of the tool?

It can be concluded that the tool is very straightforward and easy to use, especially with the user manual. The preparation steps are clearly defined and

the asked user inputs are very simple. An in-person training is always recommended as this can clear up any doubts.

(e) What is the adaptability of the tool?

Adding parameters seems straightforward, however an in-person explanation is again recommended to avoid any confusions. On top of this, a ReadMe file was proposed to explain what each part of the code does and how it can be edited without breaking the code. An in-person training can also be useful for this as editing a code created by someone else can be tricky.

This shows that the tool has a great impact on the time and error reduction due to the elimination of the manual copy-paste process. It also decreased misunderstandings and inconsistencies as the format and parameters are clearly defined. However, the impact on the cost is insignificant due to the immense total budget of ESA. This means that the tool just makes the life of the communication engineer much simpler with regards to the link budget data exchange and verification.

5. Link Budget Verification

The link budget calculations were already performed by Redwire Space by the start of this thesis. However, as the results are of importance because they are an integral part of the thesis, a verification of these calculations is required. For this, a verification tool is created which calculates the SNR taking into account all the gains and losses. How the tool is set-up is explained in Section 5.1 and all the different steps to calculate the SNR are explained in Section 5.2. Finally, the results of the verification are discussed in Section 5.3 for the S-band uplink, the S-band downlink with both a high and low data rate and lastly the X-band downlink.

5.1. Verification tool set-up

The tool uses the generated XML file from the conversion as input for the verification calculations. This way, the process of verifying the Link Budget calculations also results in a verification of the converted file by checking if the values written in the XML correspond with the values from the Excel file. To parse and read out the information from the LBDF file, the Python package xml.etree.ElementTree is used. A function is written which uses the parameter path and attribute name as input to extract the wanted value from the XML file. An attribute name is required as many parameters have multiple values such as the 'nominal value', 'favour value' and 'adverse value' and thus a specification is required. The next step is to define constant parameters that are not part of the LBDF file such as Earth's radius and Boltzmann constant. Thirdly, all the formulas that are required for the SNR are defined. More information on these steps is provided in Section 5.2. Finally, a verification check is performed between the calculated SNR value and the extracted value from the LBDF file. When the values, rounded to two decimals, match exactly, then only the SNR will be printed. However, when there is a difference, then the following parameters will be printed for both the calculated and extracted values: Ground station gain, Slant range, EIRP, Polarisation mismatch, Free space loss, Total propagation loss, G/T, C/N_0 and SNR. These are the intermediate calculations which can then be compared to easily find a mismatch between the two to see in which step something is going wrong. Due to the fact that the values are rounded to two decimals, there is a possibility that the calculated value differs from the original value extracted from the LBDF file by a maximum of 0.01 dB. Typical link margins adopt a policy of having a 3 dB margin [56] and others even up to 20 dB [57]. This shows that a discrepancy of 0.01 dB is much smaller than the required margins and thus can be neglected.

On top of this, the tool is integrated into the conversion tool. This means that one single tool converts the Excel file to XML, then validates the converted file against the XSD files and finally verifies the calculations. It then saves the converted file in the dedicated folder, writes errors, if there are any, in a separate text file and finally says if the link budget is verified or not. If not, it also shows the table with calculated and extracted values to easily find the mismatch.

5.2. Verification tool calculations

The first parameter to be calculated is the antenna gain G using Equation 5.1 [58], where η is the antenna efficiency, A is the physical aperture area in m^2 and λ is the wavelength in m. The power gain value is then converted to dB by taking the logarithm and multiplying by ten.

$$G[dB] = 10\log_{10}(\eta\frac{4\pi A}{\lambda^2}) \tag{5.1}$$

with

$$\lambda = \frac{c}{f} \tag{5.2}$$

where c is the speed of light in $\mathrm{m\,s^{-1}}$ and f is the frequency in Hz.

The next parameter to be calculated is the EIRP, using Equation 5.3 [59]. The first term in this equation is the transmitter power P_{tx} minus 30 as the transmitter power is expressed in dBm according to LBDF standards. However, for the calculations, it should be converted to dBW and thus the value 30 should be subtracted [60]. G_{tx} is the transmitting antenna gain, L_{cir} is the circuit loss and L_p is the pointing loss.

$$EIRP[dB] = P_{tx} - 30 + G_{tx} - L_{cir} - L_p \tag{5.3}$$

The next step is to calculate the Slant Range using Equation 5.4 [61]. R_E is Earth's radius, h represents the orbit altitude and ϵ is the minimum elevation angle.

$$S[km] = \sqrt{(R_E + h)^2 - R_E^2\cos^2\epsilon} - R_E\sin\epsilon \tag{5.4}$$

The slant range, in km, is then used to calculate the Free Space Path Loss (FSPL) as can be seen in Equation 5.5 [62]. The second variable in this equation is f which represents the frequency in MHz this time. The last factor to this equation is the value 32.44, which comes from taking the logarithm of $4\pi/c$ and then multiplied by twenty. As the slant range is expressed in km and the frequency in MHz, the speed of light must be adapted accordingly [63]. This then results to a contribution of 32.44 dB added to the FSPL.

$$FSPL[dB] = 20\log_{10}(S) + 20\log_{10}(f) + 32.44 \tag{5.5}$$

The polarisation mismatch can be calculated using Equation 5.6 [64]. θ is the angle between the polarization vectors. ρ_{tx} and ρ_{rx} are the circular polarization ratio of the transmitted wave and receiving antenna respectively. These can be found using Equation 5.7 where AR stands for the absolute axial ratio (not in dB) [64].

$$L_{pol}[dB] = 10\log_{10}\left(\frac{1+\rho_{tx}^2\rho_{rx}^2+2\rho_{tx}\rho_{rx}\cos 2\phi}{(1+\rho_{tx}^2)(1+\rho_{rx}^2)}\right) \tag{5.6}$$

where

$$\begin{aligned}\rho_{tx} &= (AR_{tx}+1)(AR_{tx}-1)\\ \rho_{rx} &= (AR_{rx}+1)(AR_{rx}-1)\end{aligned} \tag{5.7}$$

The total propagation loss can then be found by adding the atmospheric loss (L_{atm}), the ionospheric loss (L_{ion}), the polarisation mismatch loss (L_{pol}) and the FSPL. This is shown in Equation 5.8 [61].

$$L_{prop}[dB] = L_{atm} + L_{ion} + L_{pol} + FSPL \tag{5.8}$$

The system noise temperature T_s can be calculated using Equation 5.9 [63]. T_{ref} is the reference temperature and NF represents the noise figure.

$$T_s = T_{ref}\left(10^{\frac{NF[dB]}{10}}-1\right) \tag{5.9}$$

Because the system noise temperature consists of multiple factors, the Friis' formula, Equation 5.9, should be taken into account [65].

$$T_s = T_{G_1} + \frac{T_{G_2}}{G_1} + \frac{T_{G_3}}{G_1G_2} \tag{5.10}$$

where

$$G = \frac{1}{L} = \frac{1}{10^{\frac{NF[dB]}{10}}} \tag{5.11}$$

Equation 5.12 for the system noise temperature is found by combining Equation 5.9, Equation 5.10 and Equation 5.11 [63]. NF_1 is the noise figure of the Radio Frequency Distribution Unit (RFDU), NF_2 of the diplexer and NF_3 of the transceiver.

$$T_s[K] = T_{ref}\left(10^{\frac{NF_1}{10}}-1\right)+T_{ref}\left(10^{\frac{NF_2}{10}}-1\right)\left(10^{\frac{NF_1}{10}}\right)+T_{ref}\left(10^{\frac{NF_3}{10}}-1\right)\left(10^{\frac{NF_1}{10}}\right)\left(10^{\frac{NF_2}{10}}\right) \tag{5.12}$$

The system noise temperature is then used to calculate the gain-to-noise-temperature ratio using Equation 5.13 [63]. G_{rx} stands for the receiving antenna gain and T_s for the system noise temperature.

$$\frac{G}{T}[dB] = G_{rx} - 10\log_{10}(T_s) \tag{5.13}$$

The next step is to combine the gains and losses to calculate the carrier-to-noise ratio which can be seen in Equation 5.14 [63]. This includes the EIRP, the propagation loss, the Voltage Standing Wave Ratio (VSWR) loss L_{vswr}, the cable loss, the modulation loss L_{mod}, the implement loss L_{impl}, the pointing loss L_{point}, the gain-to-noise-temperature and finally the Boltzmann constant 'k' in dB.

$$\frac{C}{N_0}[dB \cdot HZ] = EIRP - L_{prop} - L_{vswr} - L_{cable} - L_{mod} - L_{impl} - L_{point} + \frac{G}{T} - k \quad (5.14)$$

Finally, the Signal-to-Noise Ratio (SNR) margin is calculated by subtracting the bit rate 'BR' and the required $\frac{E_b}{N_0}$ from the carrier-to-noise ratio [66].

$$SNR[dB] = \frac{C}{N_0} - 10\log_{10}(R_b) - \left(\frac{E_b}{N_0}\right)_{req} \quad (5.15)$$

It should be noted that the verification tool is created from scratch and independent from the Redwire Space calculation tool. This means that the formulas are found in available literature and thus could differ from the method applied in the Redwire Space calculation tool. This independent approach is crucial for the validity of the verification tool. An example of this difference is that certain losses can be taken into account in step A in the verification tool while the same loss is only taken into account in step B in the Redwire Space calculation tool. For example, the L_{mod} is included in the C/N_0 calculation during the verification, but only in the SNR margin calculation in the original calculation tool. It does come down to the same result in the end, however the intermediate results could differ.

5.3. Verification results

Once all the required parameters are extracted from the LBDF file and the calculations are defined, then the actual verification can be performed. There are four cases for which the verification is done, namely the S-band uplink, the S-band downlink with a high data rate, the S-band downlink with a low data rate and finally the X-band downlink. The first case is the S-band uplink and the results can be seen in Table 5.1. The ground station gain, slant range, EIRP, polarisation mismatch, free space loss and total propagation loss are calculated to be exactly the same as the extracted value. The first real difference is found in the gain-to-noise-temperature ratio, which can be explained by a different formula definition between the Redwire Space calculation tool and the verification tool. Redwire Space already took the cable loss and the VSWR loss into account in the G/T while in the verification calculations, these losses are only included later on in the carrier-to-noise ratio. The same happens for the carrier-to-noise ratio where there is a difference of 3.01 dB between the calculated value and the extracted one. The modulation loss and the implement loss are already taken into account in the C/N_0 in the verification while in the original calculation, these losses are only added at the SNR margin itself. This accounts for a difference of exactly 3.00 dB. The remaining 0.01 dB difference can be explained due to a rounding error. This difference of 0.01 dB is then also seen in the SNR margin value of the S-band

uplink link budget itself. It is up to the communication engineer then to notice that this is only a rounding error between the two and that it can be disregarded. This can be done by retrieving both the original and recalculated value with more significant digits, meaning with more precision. These values can be subtracted from each other and if this difference is smaller than 0.01 dB, then it could be classified as a rounding error. On top of this, the engineer can re-run the calculations again manually as a final resort.

Table 5.1: Verification results for the S-band uplink [1]

TT&C Uplink	Calculated	Original
GS gain [dBi]	41.58	41.58
Slant range [km]	2783.85	2783.85
EIRP [dB]	53.00	53.00
Polarisation mismatch [dB]	0.38	0.38
Free space loss [dB]	167.47	167.47
Total prop loss [dB]	169.36	169.36
G/T [dB/K]	-32.51	-33.57
Uplink C/N_0 [dBHz]	75.65	78.66
Uplink SNR margin [dB]	17.99	18.00

The next verification is the S-band downlink with a high data rate of which the results are shown in Table 5.2. The first difference that is noted, is in the gain-to-noise-temperature ratio where in the original calculations by Redwire Space, the pointing loss is already taken into account while in the verification, this is done later on. A second real difference is found in the carrier-to-noise ratio where the 'demodulation tech loss' results in a 1.00 dB difference. However, this difference disappears in the final SNR margin where the 'demodulation tech loss' is also taken into account in the original calculation. This means that the S-band downlink with a high data rate is fully verified with a downlink link margin of 5.04 dB.

Table 5.2: Verification results for the S-band downlink for high data rate [1]

TT&C Downlink high data rate	Calculated	Original
GS gain [dBi]	42.30	42.30
Slant range [km]	2783.85	2783.85
G/T [dB/K]	18.04	18.01
Polarisation mismatch [dB]	0.38	0.38
Free space loss [dB]	168.54	168.54
Total prop loss [dB]	170.43	170.43
EIRP [dB]	-7.56	-7.56
Downlink C/N_0 [dBHz]	67.62	68.62
Downlink SNR margin [dB]	5.04	5.04

[1] Intermediate values may differ due to a different parameter definition between the verification and original calculations.

The final S-band verification is for the downlink with low data rate, which is shown in Table 5.3. Again, a mismatch between the gain-to-noise-temperature ratio can be seen due to the pointing loss being already taken into account in the original value, but not yet in the verification. The same difference is noted for the carrier-to-noise ratio in this case as for the the high data rate case. In the end, the S-band downlink with low data rate is fully verified with a downlink margin of 11.11 dB.

Table 5.3: Verification results for the S-band downlink for low data rate [2]

TT&C Downlink low data rate	**Calculated**	**Original**
GS gain [dBi]	42.30	42.30
Slant range [km]	2783.85	2783.85
G/T [dB/K]	18.04	18.01
Polarisation mismatch [dB]	0.38	0.38
Free space loss [dB]	168.54	168.54
Total prop loss [dB]	170.43	170.43
EIRP [dB]	-9.56	-9.56
Downlink C/N_0 [dBHz]	65.62	66.62
Downlink SNR margin [dB]	11.11	11.11

The final case to verify is the X-band downlink case. The results of this verification are shown in Table 5.4. The ground station gain, polarisation mismatch, free space loss, total propagation loss and EIRP are exactly the same for the verification as for the extracted values. The first difference is noted in the gain-to-noise temperature which is due to the pointing loss being taken into account already in the original calculation and not during the verification calculation. This is taken into account in a later step. The second difference can be found in the carrier-to-noise ratio due to the 'demodulation tech loss' being taken into account in the verification but not yet in the extracted value. The final SNR comes both down to 4.61 dB and thus the X-band downlink link budget is fully verified.

Table 5.4: Verification results for the X-band downlink [2]

PDT Downlink	**Calculated**	**Original**
GS gain [dBi]	59.05	59.05
Slant range [km]	2783.85	2783.85
G/T [dB/K]	36.70	36.10
Polarisation mismatch [dB]	0.02	0.02
Free space loss [dB]	179.93	179.93
Total prop loss [dB]	181.42	181.42
EIRP [dB]	0.69	0.69
Downlink C/N_0 [dBHz]	82.46	83.96
Downlink SNR margin [dB]	4.61	4.61

[2] Intermediate values may differ due to a different parameter definition between the verification and original calculations.

6. Link Budget Optimisation

One of the most important aspects of a satellite link budget is the bit rate as this determines how much data is send back to Earth. Due to the constraints, as defined in Chapter 3 as ARK-REQ-1 and ARK-REQ-2 of having a maximum bandwidth of 10 MHz and a minimum link margin of 3 dB, an optimal data rate exist while still satisfying both constraints. For this, a link budget optimisation tool is introduced which uses the generated LBDF file as input. In Section 6.1, the coding gain of the different coding options is explained to show how this choice impacts the optimisation results. The three considered options are SCCC, Reed-Solomon and concatenated coding of Reed-Solomon and Convolutional code. Concatenated coding is only introduced here for the first time as it was not originally considered as an option for the ARRAKIHS mission. However, during the optimisation analysis, this option was investigated and deemed as a feasible option to include in the process. Section 6.2 explains how the optimisation tool is set up and finally, the results are discussed in Section 6.3.

6.1. Coding gain

Coding gain is the decrease in the required E_b/N_0 for the same BER by the use of an error-correction code compared to an uncoded system [67]. This means that the same BER can be achieved by using less power or a lower energy-per-bit is required. To define the coding gain of SCCC, firstly the required energy-per-bit to noise density ratio needs to be found for an uncoded signal. Looking at Figure 6.1 and using a Web Plot Digitizer, a value of 13.35 dB is found as $\frac{E_s}{N_0}$ at a BER of 10^{-6} for QPSK modulation. For 8PSK modulation, a value of 18.75 dB is found at a BER of 10^{-6} and for 16APSK it ranges from 20.75 to 21.2 dB. The constellation of 16APSK consists of two rings and the radius ratio of these two rings differs for the ACM formats with 16APSK modulation. This results in a range of required E_s/N_0 for the options with 16APSK modulation [21]. One problem still is that these are the E_s/N_0 and thus this needs to be converted to E_b/N_0 using Equation 6.1.

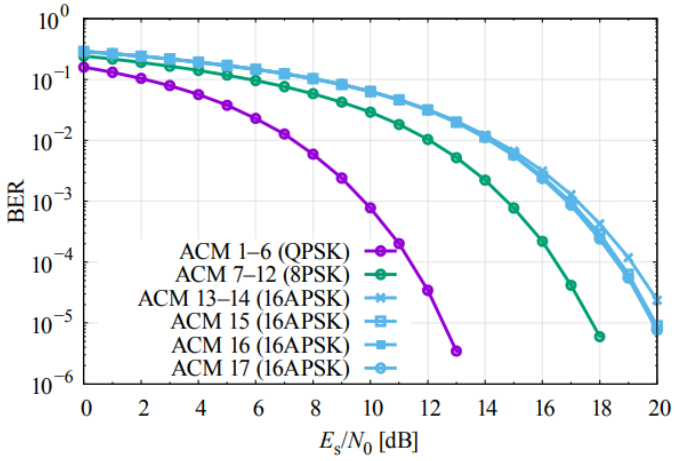

Figure 6.1: BER on Linear AWGN Channel for PSK/APSK Constellations Adopted by the ACM Formats (Uncoded BER) [21]

$$\frac{E_b}{N_0} = \frac{E_s}{N_0} - 10\log 10(k) - 10\log 10(r) \tag{6.1}$$

where $\frac{E_s}{N_0}$ is the energy-per-symbol to noise density, k is the number of bits per symbol and r represents the code rate. As for Figure 6.1, it is uncoded an thus it has a code rate of one resulting in the last term of this equation being zero. By using Equation 6.1, and the values mentioned above for the $\frac{E_s}{N_0}$ of QPSK and 8PSK, an $\frac{E_b}{N_0}$ of 10.34, 13.98 and a range of 14.73 to 15.18 dB are found for QPSK, 8PSK and 16APSK respectively at a BER of 10^{-6}.

Similarly to before, the required energy-per-bit to noise density can be found for the cases where SCCC is applied. This is shown in Figure 6.2 for the ACM formats one to twelve which corresponds with QPSK and 8PSK modulation.

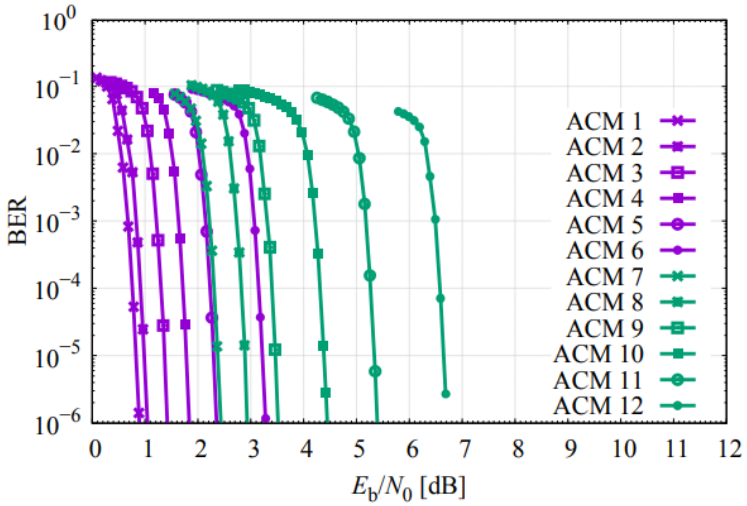

Figure 6.2: BER on Linear AWGN Channel for ACM Formats from 1 to 12 (PSK Modulations) [21]

The same needs to be done for 16APSK, which corresponds to ACM format thirteen to seventeen and can be seen in Figure 6.3. ACM formats 18 till 27, corresponding to 32APSK and 64APSK, are also shown in this figure, however these are not considered as an option for the ARRAKIHS mission due to the low robustness to noise and thus a very high required $\frac{E_b}{N_0}$.

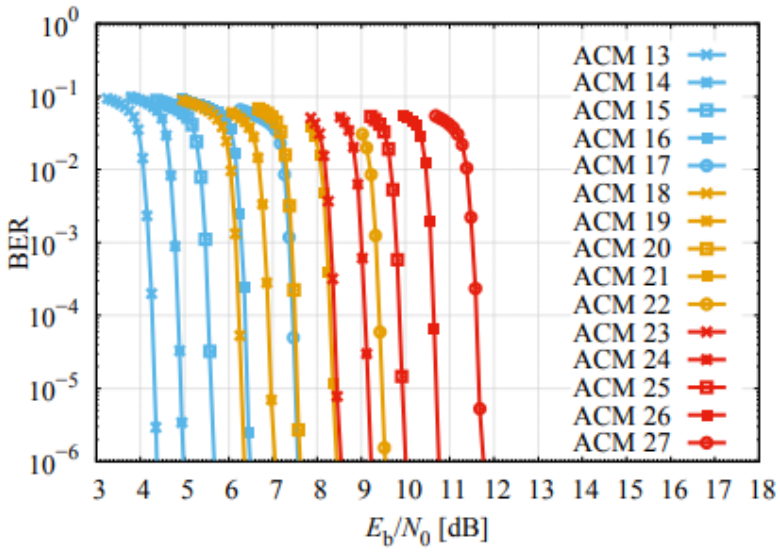

Figure 6.3: BER on Linear AWGN Channel for ACM Formats from 13 to 27 (APSK Modulations) [21]

The coding gain can then be found by taking the difference between the required energy-per-bit to noise density of the uncoded system and the SCCC coded one. This results in Table 6.1.

Table 6.1: Theoretical required E_b/N_0 and coding gain per ACM format up to ACM format 17 in Linear AWGN channel

ACM	Code rate	Req E_b/N_0 [dB]	Coding gain @ BER 10^{-6} [dB]
1	0.36	0.90	9.44
2	0.43	1.08	9.26
3	0.52	1.44	8.90
4	0.61	1.87	8.47
5	0.70	2.39	7.95
6	0.81	3.30	7.04
7	0.46	2.41	11.57
8	0.54	2.90	11.08
9	0.61	3.50	10.48
10	0.70	4.45	9.53
11	0.79	5.40	8.58
12	0.88	6.69	7.29
13	0.59	4.45	10.73
14	0.66	5.03	10.15
15	0.73	5.67	9.06
16	0.80	6.50	8.23
17	0.87	7.58	7.15

Other considered options for the ARRAKIHS mission are OQPSK and 8PSK modula-

tion with Reed-Solomon (RS) coding and OQPSK and 8PSK modulation with concatenated coding of RS + Convolutional code. The coding gain for Reed-Solomon only at BER of 10^{-6} equals 6.2 dB with a code rate of 0.88 according to ECSS-E-ST-50-01C [68], which can be seen in Table 6.2.

Table 6.2: Code rate and coding gain of (255, 223) Reed-Solomon code [68]

Coding Scheme	Code rate	Coding gain @ BER 10^{-6} [dB]
(255, 223) Reed-Solomon	0.88	6.2

The coding gain of a punctured convolutional code only with a code rate of 7/8 at BER of 10^{-6} equals only 3.9 dB [68]. This shows that Reed-Solomon achieves higher coding gains for the same code rate and on top of this, Reed-Solomon is better at correcting burst errors [69]. The advantage of convolutional coding is that it is better suited for random errors [69]. Even with this advantage, Reed-Solomon is deemed to be the superior one of the two if a choice had to be made. However, a combination of the two can exploit the advantages of both coding types and thus a concatenated code of the two is also a considered option during the optimisation. The code rates and coding gains of the concatenated options are shown in Table 6.3, which are found in ECSS-E-ST-50-01C [68].

Table 6.3: Code rate and coding gains of concatenated (255, 223) Reed-Solomon and punctured convolutional code [68]

Coding Scheme	Code rate	Coding gain @ BER 10^{-6} [dB]
(255, 223) RS and punctured convolutional rate 7/8	0.76	7.70
(255, 223) RS and punctured convolutional rate 5/6	0.73	8.40
(255, 223) RS and punctured convolutional rate 3/4	0.66	9.60

Using the previously found required $\frac{E_b}{N_0}$ for uncoded OQPSK, 8PSK and 16APSK, the required $\frac{E_b}{N_0}$ with Reed-Solomon coding only can be found and is shown in Table 6.4.

Table 6.4: Theoretical required E_b/N_O for the Reed-Solomon code options

Modulation/Coding Scheme	Req E_b/N_0 [dB]
OQPSK with Reed-Solomon	4.14
8PSK with Reed-Solomon	7.78
16APSK with Reed-Solomon	8.98

The same can be done for the concatenated coded options and the results for the required E_b/N_0 are shown in Table 6.5.

Table 6.5: Theoretical required E_b/N_0 for the concatenated RS and convolutional code options

Modulation/Coding Scheme	Req E_b/N_0 [dB]
OQPSK with RS and punctured convolutional rate 7/8	3.64
8PSK with RS and punctured convolutional rate 7/8	6.28
16APSK with RS and punctured convolutional rate 7/8	7.48
OQPSK with RS and punctured convolutional rate 5/6	1.94
8PSK with RS and punctured convolutional rate 5/6	5.58
16APSK with RS and punctured convolutional rate 5/6	6.78
OQPSK with RS and punctured convolutional rate 3/4	0.74
8PSK with RS and punctured convolutional rate 3/4	4.38
16APSK with RS and punctured convolutional rate 3/4	5.58

This shows that SCCC can achieve higher coding gains compared to Reed-Solomon coding and concatenated coding, however this is also due to having a lower code rate. This thus means that more coding is applied which results in a higher occupied bandwidth. This proves the necessity of an optimisation process to find the best combination.

6.2. Optimisation set-up

The idea of the optimisation tool is presented in Figure 6.4, where the blue boxes are the inputs, the orange one is the optimisation process itself and the green one is the result.

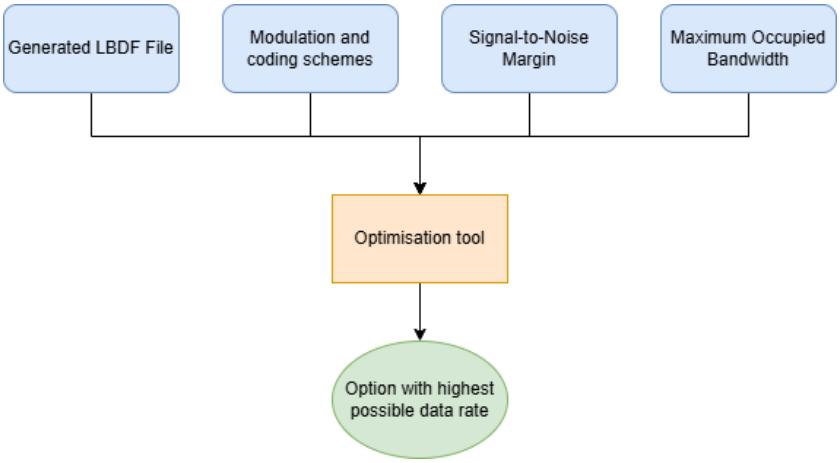

Figure 6.4: Diagram of the optimisation tool set-up

The optimisation tool uses the generated LBDF file as an input and thus the values are extracted from this file in the same way as it is done for the verification process. Secondly, the required energy-per-bit to noise density and code rate have to be defined for every considered option. These values are presented in Section 6.1. The final inputs to set are the constraints which are a maximum OBW of 10 MHz and a minimum link margin of 3 dB. Once all the inputs are set, the actual optimisation can be started. There are 29 considered options of which 17 with SCCC coding, three with Reed-Solomon coding and nine with concatenated coding. For every considered option, the tool loops through a range of bandwidths starting at 1 Hz until the constraint of a maximum occupied bandwidth of 10 MHz in ten thousand steps. This means that a step increase of 999.99 Hz is applied. This range starts at 1 Hz instead of 0 Hz because an occupied bandwidth of 0 Hz would result in a data rate of 0 bps. To calculate the SNR, the data rate is converted to dB using a logarithmic function as can be seen in Equation 5.15. This conversion is not possible when the data rate would be 0 Hz and this is why the range starts at 1 Hz. The ten thousand steps taken in this range is found iteratively. On the first try, a thousand steps was chosen, however this resulted in some options having a link margin of 3.01 dB. This shows that not the full limits are used and there is room for more optimisation. The accuracy of the tool should thus be improved by decreasing the step size and thus increasing the number of steps. The number of steps was increased with a factor of 10 and 100. In the latter case, the tool took eight times longer to run compared to the factor increase of 10, while the performance was the exact same rounded to four decimals. This shows that having ten thousand steps results in a high accuracy while being much faster than the option with one hundred thousand steps and this is why it was opted to have ten thousand steps.

The next step is to take Radio Frequency (RF) filtering into account. A Squared Root Raised Cosine (SRRC) filter is used for pulse shaping with as goal to minimise inter-symbol interference and to control the spectral width [70]. The bandwidth occupied by the transmitted signal is restricted by filtering and thus less data can be transmitted when a high roll off factor is used. The roll-off factor indicates how much filtering is

applied and should be a value between 0 and 1 with typical values used between 0.2 and 0.5. The CCSDS often opts for 0.35 [43] which is also used in this case. This results in the following relation between the bandwidth, symbol rate and the roll-off factor as can be seen in Equation 6.2 [71].

$$B = R_s * (1 + \alpha) \tag{6.2}$$

For every bandwidth in the loop, the symbol rate can be calculated using a roll-off factor of 0.35. From this, data rate can be calculated by multiplying the symbol rate by the code rate and the number of bits per symbol, as can be seen in Equation 6.3.

$$R_b = R_s * r * k \tag{6.3}$$

The bit rate is then used to calculate the SNR margin using Equation 6.4. Every time that the link margin is below the threshold of 3 dB as specified in ARK-REQ-2, the value for the best data rate is updated to the corresponding one. This loop continues until either the occupied bandwidth limit of 10 MHz (ARK-REQ-1) is reached or the link margin goes below the threshold of 3 dB (ARK-REQ-2). The optimisation process is thus a brute force optimisation which is deemed feasible because the number of options considered is very limited.

$$SNR[dB] = \frac{C}{N_0} - 10\log_{10}(R_b) - \left(\frac{E_b}{N_0}\right)_{req} \tag{6.4}$$

6.3. Optimisation results

The results of the optimisation process for the options with SCCC coding are shown in Table 6.6. The table presents the modulation type, coding applied, the code rate, the maximum achievable data rate, the occupied bandwidth, the required $\frac{E_b}{N_0}$ and finally the achieved signal-to-noise ratio margin. It can be seen that for ACM formats 1 till 11 together with formats 13 and 14, the occupied bandwidth is the limiting factor. For the other considered options using SCCC, the SNR margin is the limiting factor of the optimisation.

Table 6.6: Results of the optimisation showing the maximum possible data rate for every considered option using SCCC coding

ACM	Modulation	Coding	Code rate	Max data rate [Mbps]	OBW [MHz]	$ReqE_b/N_0$ [dB]	SNR margin [dB]
1	QPSK	SCCC	0.36	5.33	10.00	0.90	13.82
2	QPSK	SCCC	0.43	6.37	10.00	1.08	12.87
3	QPSK	SCCC	0.52	7.70	10.00	1.44	11.68
4	QPSK	SCCC	0.61	9.04	10.00	1.87	10.56
5	QPSK	SCCC	0.70	10.37	10.00	2.39	9.44
6	QPSK	SCCC	0.81	12.00	10.00	3.30	7.90
7	8PSK	SCCC	0.46	10.22	10.00	2.41	9.48
8	8PSK	SCCC	0.54	12.00	10.00	2.90	8.30
9	8PSK	SCCC	0.61	13.56	10.00	3.50	7.17
10	8PSK	SCCC	0.70	15.56	10.00	4.45	5.62
11	8PSK	SCCC	0.79	17.56	10.00	5.40	4.15
12	8PSK	SCCC	0.88	16.98	8.68	6.69	3.00
13	16APSK	SCCC	0.59	17.48	10.00	4.45	5.11
14	16APSK	SCCC	0.66	19.56	10.00	5.03	4.05
15	16APSK	SCCC	0.73	21.48	9.93	5.67	3.00
16	16APSK	SCCC	0.80	17.74	7.48	6.50	3.00
17	16APSK	SCCC	0.87	13.83	5.37	7.58	3.00

The results of the optimisation for the options with Reed-Solomon coding or concatenated coding of Reed-Solomon and Convolutional Code (CC) are shown in Table 6.7. For the OQPSK with Reed-Solomon coding, again the occupied bandwidth restricts the maximum data rate while for the 8PSK and 16APSK with Reed-Solomon, it is again the SNR margin. The options with concatenated coding are all bandwidth limited except for the ones with 16APSK modulation and a code rate of 0.73 and 0.76. Those are link margin limited.

Table 6.7: Results of the optimisation showing the maximum possible data rate for every considered option using Reed-Solomon or Reed-Solomon concatenated with Convolutional code

Modulation	Coding	Code rate	Max data rate [Mbps]	OBW [MHz]	$ReqE_b/N_0$ [dB]	SNR margin [dB]
OQPSK	RS	0.88	12.92	10.00	4.14	6.74
8PSK	RS	0.88	13.22	6.82	7.78	3.00
16APSK	RS	0.88	10.02	5.17	8.98	3.00
OQPSK	RS+CC	0.76	11.30	10.00	3.64	7.82
8PSK	RS+CC	0.76	16.96	10.00	6.28	3.42
16APSK	RS+CC	0.76	14.16	6.26	7.48	3.00
OQPSK	RS+CC	0.73	10.81	10.00	1.94	9.71
8PSK	RS+CC	0.73	16.22	10.00	5.58	4.31
16APSK	RS+CC	0.73	16.63	7.69	6.78	3.00
OQPSK	RS+CC	0.66	9.75	10.00	0.74	11.36
8PSK	RS+CC	0.66	14.62	10.00	4.38	5.96
16APSK	RS+CC	0.66	19.50	10.00	5.58	3.51

From the two tables, it can be seen that the option with the highest possible data rate is ACM format 15, corresponding to 16APSK modulation with SCCC coding with a code rate of 0.73. The details of this option are shown in Table 6.8.

Table 6.8: The optimal modulation/coding scheme showing the maximum possible data rate

Optimal modulation/coding scheme	
ACM	15
Modulation	16APSK
Coding	SCCC
Code rate	0.73
Maximum data rate [Mbps]	21.48
Occupied bandwidth [MHz]	9.93
Required E_b/N_0 [dB]	5.67
SNR margin [dB]	3.00

However, this is just by looking purely at the maximum performance. But this is not the only aspect that should be taken into account. Other factors to look at are the availability and the cost. To get a better understanding of this, a X-band transmitter supplier was contacted. After discussions with this supplier about the requirements and specifications needed for the ARRAKIHS mission, two proposals were made. One for OQPSK and 8PSK with Reed-Solomon coding or concatenated coding and another proposal for QPSK and 8PSK with SCCC coding. It should be noted that neither proposal includes the option for 16APSK modulation. To include this option, hardware changes to the transmitter are required which could only be proposed in a later stage. These changes will off course introduce extra costs. This means that the options with modulation 16APSK are eliminated. The second best option, in terms of performance,

is ACM format eleven, corresponding to 8PSK modulation and SCCC coding with a code rate of 0.79, which now becomes the best option with the elimination of 16APSK. The newly selected option has a highest possible data rate of 17.56 Mbps and the details are shown in Table 6.9. It shows that this option is bandwidth limited.

Table 6.9: The optimal modulation/coding scheme showing the maximum possible data rate excluding the 16APSK options

Optimal modulation/coding scheme excluding 16APSK	
ACM	11
Modulation	8PSK
Coding	SCCC
Code rate	0.79
Maximum data rate [Mbps]	17.56
Occupied bandwidth [MHz]	10.00
Required E_b/N_0 [dB]	5.40
SNR margin [dB]	4.15

The third factor that was mentioned to be included is the cost. The proposals showed a significant difference in cost in favour of the Reed-Solomon coding or concatenated coding option compared to SCCC[1] [2]. This cost increase is discussed internally within the ARRAKIHS team and is deemed to be too significant. This means that options including SCCC coding are eliminated, resulting in a choice reduction to either an option with Reed-Solomon coding or concatenated coding. The best performing one of these is the option of 8PSK modulation with concatenated coding of Reed-Solomon and convolutional code with a combined code rate of 0.76. This option has a maximum data rate of 16.96 Mbps. It achieves a link margin of 3.42 dB while occupying the maximum allowed bandwidth of 10 MHz. The details of this option are shown in Table 6.10.

Table 6.10: The optimal modulation/coding scheme showing the maximum possible data rate taking into account the cost

Optimal modulation/coding scheme including cost	
Modulation	8PSK
Coding	RS + convolutional
Code rate	0.76
Maximum data rate [Mbps]	16.96
Occupied bandwidth [MHz]	10.00
Required E_b/N_0 [dB]	6.28
SNR margin [dB]	3.42

[1] Data obtained through communication with X-band transmitter supplier.
[2] The exact values can not be specified due to commercial reasons.

7. Conclusions

This thesis set out to develop a conversion tool to convert the satellite link budget data from Redwire Space's calculation tool to the LBDF standard and study the impact of this tool.

What is the impact of the model based system engineering approach of the LBDF?

Once the tool was successfully created, tested and validated, the actual impact could be studied by interviewing Redwire Space and ESA employees.

(a) What is the time saved using the tool compared to the outdated method?

During the outdated method, it took a minimum of 24 hours to verify the link budgets during the mission design phase of one mission. While with the LBDF tool, it only takes around 1.8 hours which results in a time reduction of 92.5%. More details can be found in question 2 and 5 of Subsection 4.6.1.

(b) What is the error reduction due to the elimination of the manual copy-paste process of the data to and from the link budget tables?

No exact data has been collected on how many errors were made in the past, however, the interviews made clear that errors occurred quite often to even every time. It was considered part of the process. This is fully eliminated using the LBDF tool as the import and exportation process is fully automated. No more manual mistakes can be made. More details can be found in question 4 and 6 of Subsection 4.6.1.

(c) What is the cost reduction that can be achieved with LBDF?

After taking the time savings and the salary of an ESA engineer into account, it was found that €855.2 can be saved during every mission. This does not take into account the potential errors made during the manual copy-paste method and having to start over again. However, with or without the errors made, the cost savings are insignificant compared to the total budget of ESA. So relatively speaking, the impact on the cost can be neglected. More details can be found in Subsection 4.6.1.

(d) How is the operability of the tool?

It can be concluded that the tool is very straightforward and easy to use, especially with the user manual. The preparation steps are clearly defined and the asked user inputs are very simple. An in-person training is always recommended as this can clear up any doubts. More details can be found in question 11 and 12 of Subsection 4.6.1.

(e) What is the adaptability of the tool?

Adding parameters seems straightforward, however an in-person explanation is again recommended to avoid any confusions. On top of this, a ReadMe file was proposed to explain what each part of the code does and how it can be edited without breaking the code. An in-person training is definitely recommended for this as editing a code created by someone else can be tricky. More details can be found in question 11 and 12 of Subsection 4.6.1.

It can be concluded that the tool has a great impact on the time and error reduction. It also reduced the number of misunderstanding and unclarities by having a fixed format and clearly defined parameters. The impact on the cost is concluded to be insignificant as it is very small compared to the immense budget of ESA. This means that the tool mainly makes the life of the engineers much easier with regards to the link budget data exchange and verification process due to the automatisation.

Secondly, the output of the conversion tool is used as an input for the verification of the ARRAKIHS link budget. The S-band uplink, S-band low data rate downlink, S-band high data rate downlink and the X-band downlink are all successfully verified. This tool is integrated into the conversion tool and thus a single tool converts the satellite link budget, validates the generated LBDF file against the XSD files and finally verifies the calculations. This means that if all steps are successful, the file is ready to be sent to ESA for a cross-check.

What is the most suitable modulation and coding scheme for the ARRAKIHS mission?

Another goal was to find the theoretical optimal modulation and coding scheme for the ARRAKIHS mission. The theoretical exercise showed that the most suitable option was 8PSK modulation with concatenated coding of Reed-Solomon and convolutional code with code rate of 7/8, however this does not necessarily mean that this option is baselined for the ARRAKIHS mission. This option provides a maximum possible data rate of 16.96 Mbps, has an occupied bandwidth of 10.00 MHz and achieves a link margin of 3.42 dB. This was found by answering the following sub-questions.

(a) What is the coding gain of the considered coding types SCCC, Reed-Solomon and concatenated coding?

The coding gain of SCCC ranges from 7.04 dB to 11.57 dB depending on the ACM format. The coding gain for Reed-Solomon is 6.2 dB with a code rate of 0.87 while the coding gain for concatenated coding of Reed-Solomon and convolutional code ranges from 7.70 to 9.60 dB depending on the code rate of the convolutional code. More details can be found in Section 6.1.

(b) What is the maximum data rate each configuration can achieve?

The maximum data rate achieved ranges from 5.33 to 21.48 Mbps for the SCCC options while it ranges from 9.75 to 19.50 Mbps for the Reed-Solomon and concatenated code options. More details can be found in Section 6.3.

(c) What is the cost difference between the coding types?

Two proposals from a X-band transmitter supplier showed that the option including SCCC was significantly more expensive compared to the option with Reed-Solomon or concatenated coding. More details can be found in Section 6.3.

(d) What is the availability of the modulation and coding schemes in practice?

The same proposals showed that the options with 16APSK could not be proposed at the moment of request. Hardware changes were required for this and could only be proposed in later stages. This would also add extra costs. More details can be found in Section 6.3.

This shows that the impact of the conversion tool does not stop at only the direct impact of time saved, cost saved and errors reduced during the data exchange process. It also creates opportunities to use the generated LBDF file as an input for other tools within the company itself. These tools can then make the life of the engineers at Redwire Space easier, processes faster and results more accurate.

8. Recommendations

The first recommendation is that the conversion tool was designed with the primary goal of working for the ARRAKIHS link budget Excel file. This Excel file has been adapted in such a way that it works smoothly and a perfect conversion can be executed and thus this file has become the template for other satellite link budges. The tool has been tested by an independent person at Redwire Space by using this template for a different mission. However, this is only for two missions that the tool has actually been used. Further testing, by using the tool for other missions, is thus definitely recommended to make sure that there are no bugs in the conversion tool.

A second recommendation relating to the conversion tool is that the tool can be extended further. As mentioned, the tool was created with the primary goal of working for the ARRAKIHS mission. For this, only three types of conversion are included in the tool, namely 'Uplink', 'TT&C' and 'PDT'. However, there exist more types of link budgets such as an intersatellite link for example. As this was not applicable to ARRAKIHS, it is not included as an option at the moment. However, for future missions, it could become relevant to have this as an option and thus it is recommended that the tool is expanded to include these possibilities.

The third recommendation is that the optimisation tool is created for the ARRAKIHS mission and thus includes only modulation/coding options that were considered for this mission. This means that only QPSK, 8PSK and 16APSK with SCCC, corresponding to ACM formats 1-17, are included together with OQPSK, 8PSK and 16APSK with Reed-Solomon coding and concatenated coding. This could be extended to other modulation or coding types to increase the capabilities of the tool for other missions. An extra addition that could be made is to ask the user which modulation and coding types should be included in the analysis. This creates the opportunity for the user to make an easy selection. Right now it includes all the options defined, but this should not always be the case.

References

[1] Belgian Earth Observation. *Proba-V*. Accessed 15-12-2025. 2025. URL: https://eo.belspo.be/en/satellites-and-sensors/proba-v.

[2] C. Corral van Damme et al. “ARRAKIHS: ESA’s new fast-implementation science mission”. In: *Space Telescopes and Instrumentation 2024: Optical, Infrared, and Millimeter Wave*. Ed. by Laura E. Coyle, Shuji Matsuura, and Marshall D. Perrin. Vol. 13092. International Society for Optics and Photonics. SPIE, 2024, 130920Q. DOI: https://doi.org/10.1117/12.3020186.

[3] Rafael Guzmán. *ARRAKIHS ESA F-Mission*. Instituto de Física de Cantabria, 2022.

[4] ARRAKIHS Mission Consortium. *ARRAKIHS Mission*. Accessed 11-09-2025. URL: https://www.arrakihs-mission.eu/mission/.

[5] J.D. Myers. *The Local Group*. Accessed 11-09-2025. 2020. URL: https://imagine.gsfc.nasa.gov/features/cosmic/local_group_info.html.

[6] ARRAKIHS Mission Consortium. *Science Instrument*. Accessed 28-11-2025. 2025. URL: https://www.arrakihs-mission.eu/instrument/.

[7] Redwire Space. *About Redwire*. Accessed 11-09-2025. 2025. URL: https://rdw.com/about/.

[8] Redwire Space. *Locations + Facilities, Kruibeke, Belgium*. Accessed 11-09-2025. 2025. URL: https://rdw.com/locations/belgium/.

[9] Redwire Space. *Redwire Prepares for a Major European Satellite Delivery in the Fourth Quarter as International Customer Demand Increases*. Accessed 03-12-2025. 2024. URL: https://rdw.com/newsroom/redwire-prepares-for-a-major-european-satellite-delivery-in-the-fourth-quarter-as-international-customer-demand-increases/.

[10] Tony J. Rouphael. “Chapter 4 - High-Level Requirements and Link Budget Analysis”. In: *RF and Digital Signal Processing for Software-Defined Radio*. Burlington: Newnes, 2009, pp. 87–122. DOI: https://doi.org/10.1016/B978-0-7506-8210-7.00004-7.

[11] Takashi Iida and Hiromitsu Wakana. “Communications Satellite Systems”. In: *Encyclopedia of Physical Science and Technology (Third Edition)*. Ed. by Robert A. Meyers. Third Edition. New York: Academic Press, 2003, pp. 375–408. ISBN: 978-0-12-227410-7. DOI: https://doi.org/10.1016/B0-12-227410-5/00882-6.

[12] Vijay K. Garg and Yih-Chen Wang. “4 - Modulation and Demodulation Technologies”. In: *The Electrical Engineering Handbook*. Ed. by WAI-KAI CHEN. Burlington: Academic Press, 2005, pp. 971–981. DOI: https://doi.org/10.1016/B978-012170960-0/50072-4.

[13] Ali Grami. “Chapter 7 - Passband Digital Transmission”. In: *Introduction to Digital Communications*. Boston: Academic Press, 2016, pp. 299–355. DOI: https://doi.org/10.1016/B978-0-12-407682-2.00007-7.

[14] Krishn Gupt et al. “Smart Home realization through Wireless communication system”. In: *International Journal of Computer Science Engineering (IJCSE)* 5 (Sept. 2016), pp. 240–253.

[15] Jimmy Nsenga et al. “Comparison of OQPSK and CPM for communications at 60 GHz with a nonideal front end”. In: *EURASIP Journal on Wireless Communications and Networking* 2007.1 (2007), p. 086206.

[16] Leon W Couch, Muralidhar Kulkarni, and U Sripati Acharya. *Digital and analog communication systems*. Vol. 8. Pearson Upper Saddle River, 2013.

[17] Ângelo da Luz et al. “On the design of spreading sequences for CDMA systems with nonlinear OQPSK-type modulations”. In: *2014 4th International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems (VITAE)*. IEEE. 2014, pp. 1–5.

[18] M. W Cao, Guangguo Bi, and Xiufeng Jin. “Bandwidth Efficient Three-User Cooperative Diversity Scheme Based on Relaying Superposition Symbols”. In: *Wireless Sensor Network* 1 (Jan. 2009), pp. 1–6. DOI: 10.4236/wsn.2009.11001.

[19] Rastislav Róka and Martin Mokrán. “Modeling of the PSK Utilization at the Signal Transmission in the Optical Transmission Medium”. In: *International Journal of Communication Networks and Information Security* 7 (Dec. 2015), pp. 187–196. DOI: 10.17762/ijcnis.v7i3.1471.

[20] Michael Rice, Chad Josephson, and Erik Perrins. *OPTIMIZING CODED 16-APSK FOR AERONAUTICAL MOBILE TELEMETRY*. 2017. URL: http://hdl.handle.net/10150/627010.

[21] *SCCC—Summary of Definition and Performance*. 2^{nd}. Washington, D.C.: CCSDS 131.2-B-2., May 2023.

[22] Yasir Ahmed. “Modulation and Coding”. In: *Recipes for Communication and Signal Processing*. Singapore: Springer Nature Singapore, 2023, pp. 25–54. ISBN: 978-981-99-2917-7. DOI: 10.1007/978-981-99-2917-7_2.

[23] S.L. Sathiya Narayanan et al. “Implementation of forward error correction for improved performance of free space optical communication channel in adverse atmospheric conditions”. In: *Results in Optics* 16 (2024), p. 100689. ISSN: 2666-9501. DOI: https://doi.org/10.1016/j.rio.2024.100689.

[24] *Flexible Advanced Coding and Modulation Scheme for High Rate Telemetry Applications*. 2^{nd}. Washington, D.C.: CCSDS 131.2-B-2., February 2023.

[25] Fabian Ogenyi et al. “A comprehensive review of adaptive modulation and coding techniques for spectrum efficiency and interference mitigation in satellite communication”. In: *KIU Journal of Science Engineering and Technology* 3 (Feb. 2025), pp. 39–50.

[26] Xinyu Lei. “The Comparison of RS Codes, LDPC Codes, And Turbo Codes”. In: *Highlights in Science, Engineering and Technology* 81 (Jan. 2024), pp. 496–504. DOI: 10.54097/8tn0ya36.

[27] Sergio Benedetto and Guido Montorsi. “Design of parallel concatenated convolutional codes”. In: *IEEE Transactions on Communications* 44.5 (2002), pp. 591–600.

[28] Imed Amamra and Nadir Derouiche. “A stopping criteria for turbo decoding based on the LLR histogram”. In: *2012 16th IEEE Mediterranean Electrotechnical Conference*. IEEE. 2012, pp. 699–702. DOI: 10.1109/MELCON.2012.6196527.

[29] R. Gallager. “Low-density parity-check codes”. In: *IRE Transactions on Information Theory* 8.1 (1962), pp. 21–28. DOI: 10.1109/TIT.1962.1057683.

[30] Sarah J Johnson. “Introducing low-density parity-check codes”. In: *University of Newcastle, Australia* 1 (2006), pp. 1–83.

[31] John Dielissen, Andries Hekstra, and Vincent Berg. “Low cost LDPC decoder for DVB-S2”. In: *Proceedings of the Design Automation & Test in Europe Conference*. Vol. 2. IEEE. 2006, pp. 1–6.

[32] Sukhleen Bindra Narang, Kunal Pubby, and Hashneet Kaur. “Low-density parity check (LDPC) codes: A new era in coding”. In: *Signal And Image Processing* 1.1 (2016), pp. 7–14.

[33] Latifa Mostari and Abdelmalik Taleb-Ahmed. “High performance short-block binary regular LDPC codes”. In: *Alexandria Engineering Journal* 57.4 (2018), pp. 2633–2639. ISSN: 1110-0168. DOI: https://doi.org/10.1016/j.aej.2017.09.016.

[34] Rene Cumplido et al. “Towards a reconfigurable hardware architecture for implementing a LDPC module suitable for software radio systems”. In: (2010), pp. 1–8. DOI: https://doi.org/10.4230/DagSemProc.10281.13.

[35] Bernard Sklar. “Reed-solomon codes”. In: *Downloaded from URL http://www. informit. com/content/images/art. sub.–sklar7. sub.–reed-solomo-n/elementLinks/art. sub.–sklar7. sub.–reed-solomon. pdf* (2001), pp. 1–33.

[36] William A Geisel. *Tutorial on reed-solomon error correction coding*. Vol. 102162. National Aeronautics and Space Administration, Lyndon B. Johnson Space Center, 1990.

[37] C.K.P. Clarke. *Reed-Solomon error correction*. BBC, 2002.

[38] S. Benedetto et al. “MHOMS: high-speed ACM modem for satellite applications”. In: *IEEE Wireless Communications* 12.2 (2005), pp. 66–77. DOI: 10.1109/MWC.2005.1421930.

[39] Muhammad Awais and Carlo Condo. “Flexible LDPC decoder architectures”. In: *VLSI Design* 2012 (June 2012). DOI: 10.1155/2012/730835.

[40] Unknown. *Constant Envelope OQPSK vs QPSK*. Accessed 14-05-2025. 2025. URL: https://electroagenda.com/en/constant-envelope-oqpsk-vs-qpsk/.

[41] Unknown. *QPSK vs. 8-PSK: A Comparative Analysis of Modulation Techniques*. Accessed 14-05-2025. 2025. URL: https://www.rfwireless-world.com/terminology/qpsk-vs-8psk-modulation.

[42] Viasat. *HI-BEAM Transceiver DVB-S2 High-Speed Modem*. Accessed 15-05-2025. 2017. URL: https://www.viasat.com/content/dam/us-site/products/semiconductors/documents/hi-beam_modem_127_web_125864.pdf.

[43] *Radio Frequency and Modulation Systems—Part 1: Earth Stations and Spacecraft*. 32^{nd}. Washington, D.C.: CCSDS 401.0-B-32, October 2021.

[44] CCSDS. *Link Budget Digital Format XML*. Accessed 05-06-2025. 2025. URL: https://beta.sanaregistry.org/r/link_budget_digital_format_xml/.

[45] Olaf Kath. *Model-based Systems Engineering (MBSE), Explained*. Accessed 05-11-2025. 2025. URL: https://www.ansys.com/blog/model-based-systems-engineering-explained.

[46] International Telecommunication Union. *Radio Regulations*. ITU, 2024. URL: https://www.itu.int/hub/publication/r-reg-rr-2024/.

[47] ESA ESTEC. *Margin philosophy for science assessment studies*. European Space Agency, 2012.

[48] CCSDS. *Link Budget Digital Format XML - Link Budget Model*. Accessed 05-06-2025. 2025. URL: https://beta.sanaregistry.org/files/link_budget_digital_format_xml/LinkBudgetModel.pdf.

[49] Joseph H Yuen. *Deep space telecommunications systems engineering*. Springer Science & Business Media, 2013.

[50] Python Documentation. *xml.etree.ElementTree — The ElementTree XML API*. Accessed 24-07-2025. 2025. URL: https://docs.python.org/3/library/xml.etree.elementtree.html.

[51] Unknown. *How to Convert Excel to XML Format in Python?* Accessed 09-05-2025. 2025. URL: https://www.geeksforgeeks.org/python/how-to-convert-excel-to-xml-format-in-python/.

[52] Stephan Behnel. *ElementTree*. Accessed 05-01-2026. 2011. URL: https://wiki.python.org/moin/ElementTree#:~:text=Both%20the%20original%20implementation%20and,etree)%20since%20Python%202.5..

[53] European Space Agency. *What we offer*. Accessed 07-11-2025. 2025. URL: https://www.esa.int/About_Us/Careers_at_ESA/What_we_offer#salaries.

[54] N2YO. *Satellites by Country and Organization*. Accessed 11-11-2025. 2025. URL: https://www.n2yo.com/satellites/?c=&t=country.

[55] ESA. *ESA Member States commit to largest contributions at Ministerial*. Accessed 28-11-2025. 2025. URL: https://www.esa.int/About_Us/Corporate_news/ESA_Member_States_commit_to_largest_contributions_at_Ministerial.

[56] K Cheung. "The role of margin in link design and optimization". In: *2015 IEEE Aerospace Conference*. IEEE. 2015, pp. 1–11.

[57] Department of ECE. *Link Budget and Link Margin*. Rohini College of Engineering and Technology.

[58] Dirk Giggenbach, Marcus Knopp, and Christian Fuchs. “Link budget calculation in optical LEO satellite downlinks with on/off□keying and large signal divergence: A simplified methodology”. In: *International Journal of Satellite Communications and Networking* 41 (Apr. 2023), pp. 460–476. DOI: 10.1002/sat.1478.

[59] Mir Hassan et al. “Optimization of Satellite Link to Earth Station Using Satellite Tool Kit (STK)”. In: *International Journal of Electronics* (Dec. 2018).

[60] Frank H Sanders. *Derivations of relationships among field strength, power in transmitter-receiver circuits and radiation hazard limits*. Tech. rep. Institute for Telecommunication Sciences, 2010.

[61] Dario Di Francesco. “Development and application of Link Budget Calculator for a Spacecraft mission: GSn design and Link Budget analysis for SROC support”. Master’s thesis. Turin, Italy: Politecnico di Torino, 2022. URL: https://webthesis.biblio.polito.it/25148/.

[62] Md Akhtaruzzaman et al. “Link Budget Analysis in Designing a Web-application Tool for Military X-Band Satellite Communication”. In: *Mist International Journal of Science and Technology* 8 (July 2020), pp. 17–33. DOI: 10.47981/j.mijst.08(01)2020.174(17-33).

[63] Louis J. Ippolito Jr. “The RF Link”. In: *Satellite Communications Systems Engineering*. John Wiley and Sons, Ltd, 2017. Chap. 4, pp. 49–74. ISBN: 9781119259411. DOI: https://doi.org/10.1002/9781119259411.ch4.

[64] Rastislav Galuscak and Pavel Hazdra. *Circular polarization and polarization losses*. 2006. URL: https://api.semanticscholar.org/CorpusID:15344097.

[65] Armin W. Doerry. *Noise and Noise Figure for Radar Receivers*. Tech. rep. Sandia National Laboratories (SNL-NM), Albuquerque, NM (United States), Sept. 2016. DOI: 10.2172/1562649. URL: https://www.osti.gov/biblio/1562649.

[66] Sergi Gallardo Marquina. “Study and simulation of communication links in a LEO satellite constellation based on Link Budget calculations”. Bachelor’s thesis. Barcelona, Spain: Polytechnic University of Catalonia, 2022. URL: https://upcommons.upc.edu/entities/publication/7ea594f8-dcb9-4c39-8b5f-a87dda267eba.

[67] Ivan J. Fair Vijay K. Bhargava. “Forward Error Correction Coding”. In: *The Communications Handbook*. Vol. 2. CRC Press. 2002. DOI: https://doi.org/10.1201/9781420041163.

[68] European Cooperation for Space Standardization. *ECSS-E-ST-50-01C: Space data links - Telemetry synchronization and channel coding*. Noordwijk, Netherlands : ECSS Secretariat, 2008.

[69] K Suman. “Concatenation and Implementation of Reed-Solomon and Convolutional Codes”. In: *International Journal on Recent and Innovation Trends in Computing and Communication* 5.6 (), pp. 978–983.

[70] S Vinay, R Mahaanth, and N Shreyas. “Implementation of SRRC Filter Using Variable Alpha”. In: (2025). DOI: 10.21203/rs.3.rs-8046725/v1.

[71] Michael Zoltowski. “Equations for the raised cosine and square-root raised cosine shapes”. In: *Communication Systems Division* (2013).

A. Satellite link budget table

1	Owner CCSDS Agency	
2	Name of Mission	
3	Name of Spacecraft	
4	Mission Category a. A = Alt.<2,000,000 km b. B = Alt.>2,000,000 km	
5	Link Budget Number	
6	Revision No. / Conditions	
7	Date	
8	File Name	
9	Project Cognizant Person Name: Title: Address: Telephone No: Fax No: Email:	
10	Network Cognizant Person Name: Address: Telephone No: FAX No: Email:	

Figure A.1: CCSDS Link Design Control Table General Information [43]

Page 2

EARTH-SPACE LINK			SPACE-EARTH LINK		
E/S TRANSMITTING RF CHANNEL:			***S/C TRANSMITTING RF CHANNEL:***		
1	RF Carrier Modulation		11	RF Carrier Modulation	
	a. Ch 1 Type			a. Ch 1 Type	
	b. Ch 1 Format			b. Ch 1 Format	
	c. Ch 2 Type			c. Ch 2 Type	
	d. Ch 2 Format			d. Ch 2 Format	
E/S TRANSMITTING DATA CHANNEL:			***S/C TRANSMITTING DATA CHANNEL:***		
2	Baseband Data		12	Baseband Data	
	a. Ch 1 Bit Rate, b/s			a. Ch 1 Bit Rate, kb/s	
	b. Ch 1 Bit Error Rate			b. Ch 1 Bit Error Rate	
	c. Ch 2 Bit Rate, b/s			c. Ch 2 Bit Rate, kb/s	
	d. Ch 2 Bit Error Rate			d. Ch 2 Bit Error Rate	
3	Data Coding		13	Data Coding	
	a. Ch 1 Type			a. Ch 1 Rate	
	b. Ch 1 No. Info Bits			b. Ch 1 Constraint Length	
	c. Ch 1 Block Length			c. Ch 1 Concatenated Code	
				d. Ch 1 Data/Total Bits	
	d. Ch 2 Type			e. Ch 2 Rate	
	e. Ch 2 No. Info Bits			f. Ch 2 Constraint Length	
	f. Ch 2 Block Length			g. Ch 2 Concatenated Code	
				h. Ch 2 Data/Total Bits	
4	Subcarrier		14	Subcarrier	
	a. Ch 1 Waveform			a. Ch 1 Waveform	
	b. Ch 1 Frequency			b. Ch 1 Frequency	
	c. Ch 1 Mod Type			c. Ch 1 Modulation Type	
	d. Ch 2 Waveform			d. Ch 2 Waveform	
	e. Ch 2 Frequency			e. Ch 2 Frequency	
	f. Ch 2 Mod Type			f. Ch 2 Modulation Type	
E/S TRANSMITTING RNG CHANNEL:			***S/C-E/S RNG CHANNEL:***		
5	a. System Type		15	a. Code Regeneration	
	b. Tone/Code Wavfrm			b. Coherent Ops Reqd	
	c. Highest Frequency			c. Required Accuracy (m)	
	d. Lowest Frequency			d. Bandwidth T/C 1 (Hz)	
	e. Total Comp No.			e. Bandwidth T/C 2 (Hz)	
EARTH-TO-SPACE PATH PERFORMANCE:			***SPACE-TO-EARTH PATH PERFORMANCE:***		
6	a. Weather Avail (%)		16	a. Weather Avail (%)	
	b. S/C Distance (km)			b. S/C Distance (km)	

Figure A.2: CCSDS Link Design Control Table Communication System Operating Conditions [43]

Page 3

MISSION AND SPACECRAFT		UNITS	CHANNEL 1			CHANNEL 2		
			DESIGN VALUE	FAV TOL	ADV TOL	DESIGN VALUE	FAV TOL	ADV TOL
E/S TRANSMITTING RF CARRIER CHANNEL PARAMETERS:								
1	Transmitter Power	dBW						
2	Transmitter Frequency	MHz						
3	Antenna Gain	dBi						
4	Antenna Circuit Loss	dB						
5	Antenna Pointing Loss	dB						
E/S TRANSMITTING DATA CHANNEL PARAMETERS:								
6	Information Bit Rate	b/s						
7	Subcarrier Frequency	kHz						
8	Subcarrier Waveform	Sin-Sq						
9	RF Modulation Index	Rad-pk						
E/S TRANSMITTING RANGING CHANNEL PARAMETERS:								
10	Simultaneous With Data	Yes-No						
11	Ranging Waveform	Sin-Sq						
12	a. Mod Index Tone/Code 1	Rad-pk						
	b. Mod Index Tone/Code 2	Rad-pk						
EARTH-TO-SPACE PATH PARAMETERS:								
13	Topocentric Range	km						
14	Atmospheric Attenuation	dB						
15	Ionospheric Loss	dB						
16	Antenna Elevation Angle	deg						
S/C RECEIVING RF CARRIER CHANNEL PARAMETERS:								
17	Antenna Gain	dBi						
18	Polarization Loss	dB						
19	Antenna Pointing Loss	dB						
20	Antenna Circuit Loss	dB						
21	Carrier Circuit Loss	dB						
22	Total Noise Temperature	K						
	a. Receiver Operating Temp	K						
	b. Feed Through Noise	K						
	c. Hot Body Noise	K						
23	Threshold Loop Noise BW	Hz						
24	Reqd Threshold SNR in 2 B_{LO}	dB						
S/C RECEIVING DATA CHANNEL PARAMETERS:								
25	Phase Jitter Loss	dB						
26	Demodulator / Detector Loss	dB						
27	Waveform Distortion Loss	dB						
28	Max Rng Interference to Data	dB						
29	Reqd Data E_b/N_o	dB						
S/C RECEIVING RNG CHANNEL PARAMETERS:								
30	Ranging Demodulator Loss	dB						
31	Ranging Filter Bandwidth	MHz						
32	Required Tone/Code 1 SNR	dB						
33	Required Tone/Code 2 SNR	dB						

Figure A.3: CCSDS Link Design Control Table Earth-Space Link Input Data Sheet [43]

B. Project Gantt Chart

Figure B.1: Thesis Gantt Chart

C. Link Budget Digital Format Structure

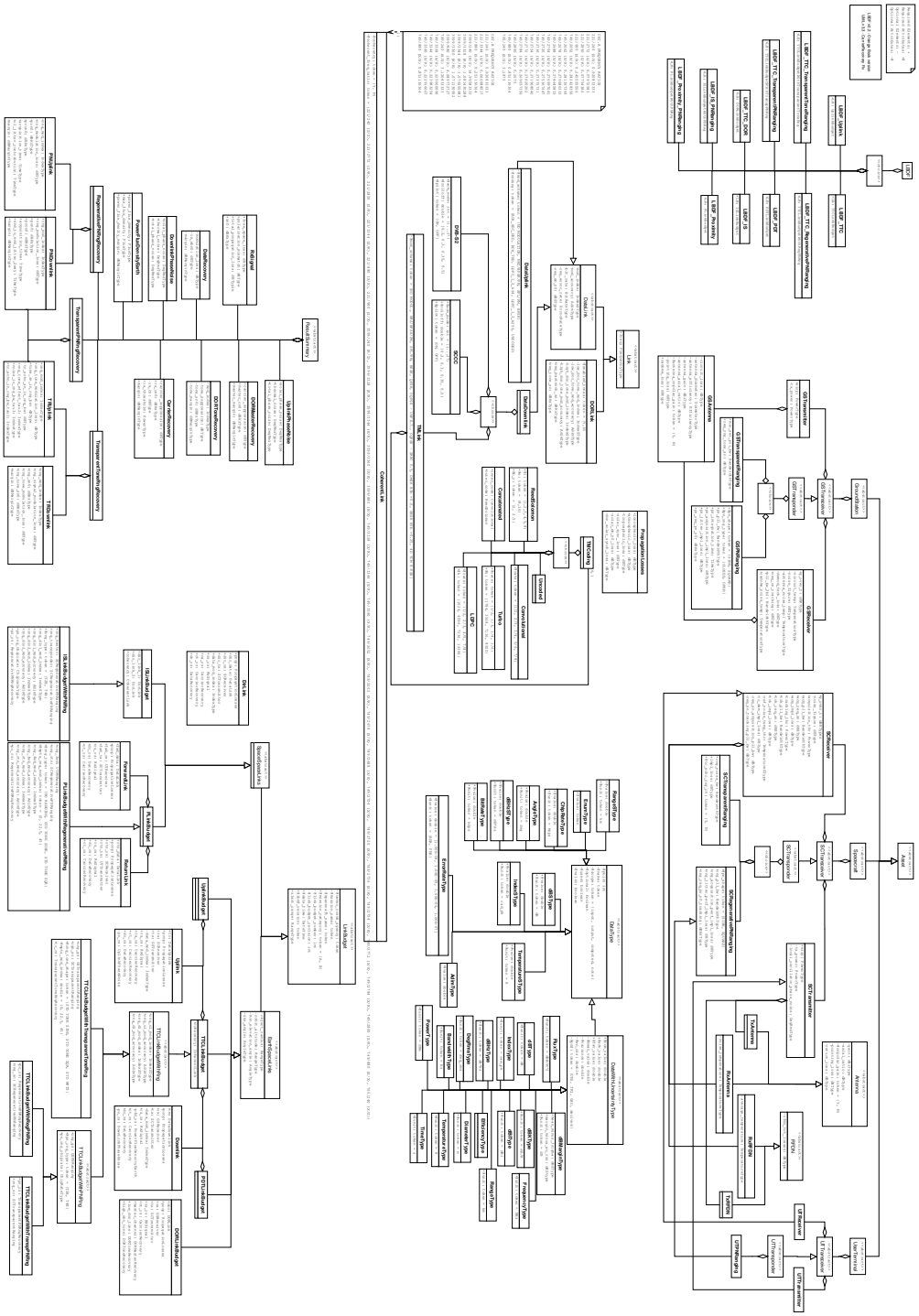

Figure C.1: Link Budget Digital Format Structure in UML [48]

D. Converted LBDF file

This example shows a TT&C S-band link budget in LBDF format. This is a dummy file and thus does not correspond to any ongoing mission.


```
<lbdf xmlns="https://www.esa.int/LBDF">
  <ttc_link_budget spacecraft_name="" mission_category="A" date_time="2026-01-03T16:02:24+01:00" lbdf_version="1.0.0">
    <slant_range nom_value="2469.02" adv_value="2489.19" fav_value="2448.77" mean_value="2469.02" unit="km"/>
    <orbit_altitude value="850.00" unit="km"/>
    <min_elevation_angle value="10.00" unit="deg"/>
    <coherency coherent_freq_ratio="221/240 (S/S)"/>
    <uplink>
      <tc mod_scheme="OQPSK - SRRC 0.5" coding="BCH_SEC_DED">
        <freq nom_value="2200.00" adv_value="2200.00" fav_value="2200.00" unit="MHz"/>
        <bit_rate value="100.00" unit="kbps"/>
        <req_error_rate value="1e-07" unit="FER"/>
        <req_eb_n0 value="8" unit="dB"/>
      </tc>
      <prop>
        <ionospheric_loss nom_value="0.00" adv_value="0.10" fav_value="0.00" mean_value="0.05" variance="0.00" pdf="GAU" unit="dB"/>
        <atmospheric_loss nom_value="1.30" adv_value="2.00" fav_value="1.00" mean_value="1.50" variance="0.01" pdf="GAU" unit="dB"/>
      </prop>
      <rx>
        <g_over_t nom_value="-32.29" adv_value="-34.74" fav_value="-29.77" mean_value="-32.26" unit="dB/K"/>
        <noise_figure nom_value="3.30" adv_value="3.50" fav_value="3.10" unit="dB"/>
        <acquisition_thr nom_value="-124.20" adv_value="-124.00" fav_value="-124.40" unit="dBm"/>
        <acq_pll_bw nom_value="500.00" adv_value="600.00" fav_value="400.00" unit="Hz"/>
        <acq_req_sn value="10.00" unit="dB"/>
        <acq_impl_loss nom_value="1.50" adv_value="1.70" fav_value="1.30" mean_value="1.50" variance="0.01" pdf="TRI" unit="dB"/>
        <tracking_thr nom_value="-107.89" adv_value="-107.39" fav_value="-108.39" unit="dBm"/>
        <trk_pll_bw nom_value="0.00" adv_value="0.00" fav_value="0.00" unit="Hz"/>
        <trk_req_sn value="0.00" unit="dB"/>
        <trk_impl_loss nom_value="0.00" adv_value="0.00" fav_value="0.00" unit="dB"/>
        <rx_antenna>
          <gain nom_value="2.00" adv_value="0.50" fav_value="3.50" mean_value="2.00" variance="0.38" pdf="TRI" unit="dBi"/>
          <axial_ratio nom_value="8.50" adv_value="10.50" fav_value="6.50" unit="dB"/>
          <pointing_loss nom_value="0.00" adv_value="0.00" fav_value="0.00" mean_value="0.00" variance="0.00" pdf="TRI" unit="dB"/>
          <noise_temperature nom_value="180.00" adv_value="195.00" fav_value="165.00" unit="K"/>
        </rx_antenna>
        <rx_rfdn>
          <rfdn_tot_losses nom_value="3.35" adv_value="3.55" fav_value="3.15" mean_value="3.35" variance="0.01" pdf="UNI" unit="dB"/>
          <vswr_loss nom_value="0.18" adv_value="0.24" fav_value="0.12" mean_value="0.18" variance="0.00" pdf="TRI" unit="dB"/>
        </rx_rfdn>
      </rx>
      <tx>
        <eirp nom_value="84.52" adv_value="84.05" fav_value="84.99" mean_value="84.52" variance="0.04" unit="dBm"/>
        <tx_power nom_value="48.00" adv_value="47.93" fav_value="48.07" mean_value="48.00" variance="0.00" pdf="TRI" unit="dBm"/>
        <gs_antenna>
          <circuit_loss nom_value="5.00" adv_value="5.10" fav_value="4.90" mean_value="5.00" variance="0.00" pdf="UNI" unit="dB"/>
          <antenna_gain nom_value="41.92" adv_value="41.72" fav_value="42.12" mean_value="41.92" variance="0.01" pdf="UNI" unit="dBi"/>
          <antenna_axial_ratio nom_value="1.60" adv_value="2.00" fav_value="0.50" unit="dB"/>
          <pointing_loss nom_value="0.40" adv_value="0.50" fav_value="0.30" mean_value="0.40" variance="0.00" pdf="UNI" unit="dB"/>
          <radome_loss nom_value="0.00" adv_value="0.00" fav_value="0.00" mean_value="0.00" variance="0.00" unit="dB"/>
        </gs_antenna>
      </tx>
      <data_mod_index nom_value="1.00" adv_value="1.05" fav_value="0.95" unit="rad_pk"/>
      <rs_rs>
        <free_space_loss nom_value="167.15" adv_value="167.22" fav_value="167.08" mean_value="167.15" unit="dB"/>
        <polarization_mismatch nom_value="0.37" adv_value="0.56" fav_value="0.09" mean_value="0.32" variance="0.02" pdf="UNI" unit="dB"/>
        <total_propagation_loss nom_value="168.82" adv_value="169.87" fav_value="168.17" mean_value="169.02" unit="dB"/>
        <sn0 nom_value="82.01" adv_value="78.03" fav_value="85.65" mean_value="81.84" variance="0.45" unit="dB*Hz"/>
      </rs_rs>
```


Figure D.1: Part 1 of the dummy TT&C S-band link budget in LBDF format

```
  <ca_rs>
    <rx_sn nom_value="12.07" adv_value="11.93" fav_value="11.66" unit="dB"/>
    <margin nom_value="33.27" adv_value="27.15" fav_value="38.00" mean_value="32.75" variance="0.70" unit="dB">
      <mean_minus_3sigma value="30.24" unit="dB"/>
      <margin_minus_wc_rss value="30.89" unit="dB"/>
    </margin>
  </ca_rs>
  <ct_rs>
    <carrier_suppression nom_value="2.00" adv_value="2.58" fav_value="2.08" mean_value="2.22" variance="0.02" pdf="TRI" unit="dB"/>
    <rx_cn nom_value="51.52" adv_value="45.97" fav_value="56.25" unit="dB"/>
    <rx_cn0 nom_value="82.01" adv_value="78.03" fav_value="85.65" mean_value="81.84" variance="0.45" unit="dB*Hz"/>
    <margin nom_value="37.44" adv_value="33.76" fav_value="40.72" mean_value="31.04" variance="0.43" unit="dB">
      <mean_minus_3sigma value="29.06" unit="dB"/>
      <margin_minus_wc_rss value="35.73" unit="dB"/>
    </margin>
  </ct_rs>
  <dr_rs>
    <modulation_loss nom_value="1.50" adv_value="1.24" fav_value="1.79" mean_value="1.51" variance="0.01" pdf="TRI" unit="dB"/>
    <rx_ebn0 nom_value="29.51" adv_value="24.29" fav_value="33.06" unit="dB"/>
    <margin nom_value="21.51" adv_value="14.69" fav_value="23.46" mean_value="19.30" variance="0.59" unit="dB">
      <mean_minus_3sigma value="16.99" unit="dB"/>
      <margin_minus_wc_rss value="19.12" unit="dB"/>
    </margin>
  </dr_rs>
</uplink>
<downlink>
  <tm>
    <freq nom_value="2200.00" adv_value="2200.00" fav_value="2200.00" unit="MHz"/>
    <bit_rate value="900.00" unit="kbps"/>
    <req_error_rate value="1e-07" unit="FER"/>
    <req_eb_n0 value="3.1" unit="dB"/>
    <standard_mod_coding mod_scheme="OQPSK - SRRC 0.5">
      <tm_coding>
        <reed_solomon i="5" e="16" b_c="2"/>
      </tm_coding>
    </standard_mod_coding>
  </tm>
  <prop>
    <ionospheric_loss nom_value="0.00" adv_value="0.10" fav_value="0.00" mean_value="0.05" variance="0.00" pdf="GAU" unit="dB"/>
    <atmospheric_loss nom_value="1.30" adv_value="2.00" fav_value="1.00" mean_value="1.50" variance="0.01" pdf="GAU" unit="dB"/>
  </prop>
  <rx>
    <g_over_t nom_value="17.70" adv_value="17.29" fav_value="18.51" mean_value="17.90" variance="0.02" unit="dB/K"/>
    <noise_figure nom_value="2.08" adv_value="2.28" fav_value="1.88" unit="dB"/>
    <antenna_noise_temp nom_value="58.00" adv_value="58.00" fav_value="58.00" unit="K"/>
    <demod_tech_loss nom_value="1.00" adv_value="1.00" fav_value="0.90" mean_value="0.97" variance="0.00" pdf="TRI" unit="dB"/>
    <req_cn_tracking nom_value="17.00" adv_value="17.00" fav_value="17.00" mean_value="17.00" unit="dB"/>
    <pll_bw_2bl nom_value="200.00" adv_value="200.00" fav_value="200.00" mean_value="200.00" unit="Hz"/>
    <radome_noise_temp nom_value="0.00" adv_value="0.00" fav_value="0.00" unit="K"/>
    <gs_antenna>
      <circuit_loss nom_value="0.70" adv_value="0.90" fav_value="0.50" unit="dB"/>
      <antenna_gain nom_value="41.92" adv_value="41.72" fav_value="42.12" mean_value="41.92" variance="0.01" pdf="UNI" unit="dBi"/>
      <antenna_axial_ratio nom_value="1.60" adv_value="2.00" fav_value="0.50" unit="dB"/>
      <pointing_loss nom_value="0.02" adv_value="0.04" fav_value="0.01" mean_value="0.02" variance="0.00" pdf="UNI" unit="dB"/>
      <radome_loss nom_value="0.00" adv_value="0.00" fav_value="0.00" unit="dB"/>
    </gs_antenna>
  </rx>
```


Figure D.2: Part 2 of the dummy TT&C S-band link budget in LBDF format

```
▼<tx>
   <eirp nom_value="21.85" adv_value="18.88" fav_value="24.60" mean_value="21.74" variance="0.70" unit="dBm"/>
   <tx_power nom_value="25.00" adv_value="24.50" fav_value="25.50" mean_value="25.00" variance="0.04" unit="dBm"/>
  ▼<tx_antenna>
     <gain nom_value="2.00" adv_value="0.50" fav_value="3.50" mean_value="2.00" variance="0.38" pdf="TRI" unit="dBi"/>
     <axial_ratio nom_value="8.50" adv_value="10.50" fav_value="6.50" unit="dB"/>
     <pointing_loss nom_value="0.00" adv_value="0.00" fav_value="0.00" mean_value="0.00" variance="0.00" pdf="TRI" unit="dB"/>
   </tx_antenna>
  ▼<tx_rfdn>
     <rfdn_tot_losses nom_value="4.97" adv_value="5.88" fav_value="4.28" unit="dB"/>
     <vswr_loss nom_value="0.18" adv_value="0.24" fav_value="0.12" unit="dB"/>
   </tx_rfdn>
 </tx>
▼<rs_rs>
   <free_space_loss nom_value="167.15" adv_value="167.22" fav_value="167.08" mean_value="167.15" unit="dB"/>
   <polarization_mismatch nom_value="0.37" adv_value="0.56" fav_value="0.09" mean_value="0.32" variance="0.02" pdf="UNI" unit="dB"/>
   <total_propagation_loss nom_value="168.82" adv_value="169.87" fav_value="168.17" mean_value="169.02" unit="dB"/>
   <sn0 nom_value="69.33" adv_value="64.90" fav_value="73.54" mean_value="69.22" variance="0.70" unit="dB*Hz"/>
 </rs_rs>
▼<pfd_rs>
   <power_flux_density nom_value="-170.53" adv_value="-173.57" fav_value="-167.72" mean_value="-170.64" unit="dBW/m^2"/>
   <max_flux_density nom_value="-151.50" adv_value="-151.50" fav_value="-151.50" unit="dBW/m^2"/>
  ▼<power_flux_density_margin nom_value="19.03" adv_value="22.07" fav_value="16.22" mean_value="19.14" variance="0.00" unit="dB">
     <mean_minus_3sigma value="19.14" unit="dB"/>
     <margin_minus_wc_rss value="14.66" unit="dB"/>
   </power_flux_density_margin>
 </pfd_rs>
▼<ct_rs>
   <carrier_suppression nom_value="0.00" adv_value="0.00" fav_value="0.00" mean_value="0.00" variance="0.00" pdf="TRI" unit="dB"/>
   <rx_cn nom_value="51.52" adv_value="45.97" fav_value="56.25" unit="dB"/>
   <rx_cn0 nom_value="82.01" adv_value="78.03" fav_value="85.65" mean_value="81.84" variance="0.45" unit="dB*Hz"/>
  ▼<margin nom_value="29.32" adv_value="24.89" fav_value="33.53" mean_value="29.21" variance="0.70" unit="dB">
     <mean_minus_3sigma value="26.70" unit="dB"/>
     <margin_minus_wc_rss value="27.30" unit="dB"/>
   </margin>
 </ct_rs>
▼<dr_rs>
   <modulation_loss nom_value="0.00" adv_value="0.00" fav_value="0.00" mean_value="0.00" variance="0.00" pdf="TRI" unit="dB"/>
   <rx_ebn0 nom_value="8.79" adv_value="4.36" fav_value="13.09" mean_value="8.71" unit="dB"/>
  ▼<margin nom_value="5.69" adv_value="1.26" fav_value="9.99" mean_value="5.61" variance="0.70" unit="dB">
     <mean_minus_3sigma value="3.10" unit="dB"/>
     <margin_minus_wc_rss value="3.67" unit="dB"/>
   </margin>
 </dr_rs>
</downlink>
</ttc_link_budget>
</lbdf>
```


Figure D.3: Part 3 of the dummy TT&C S-band link budget in LBDF format