Hardware Precoding Demonstration in Multi-Beam UHTS Communications under Realistic Payload Characteristics

Juan Duncan, Jorge Querol, Nicola Maturo, Jevgenij Krivochiza, Danilo Spano, Norshahida Saba, Liz Marrero, Symeon Chatzinotas, Björn Ottersten[1]

[1] *SnT, University of Luxembourg, 29 Avenue JF Kennedy L-1855 Luxembourg*

October 4, 2019

Abstract

In this paper, we present a new hardware test-bed to demonstrate closed-loop precoded communications for interference mitigation in multi-beam ultra high throughput satellite systems under realistic payload and channel impairments. We build the test-bed to demonstrate a real-time channel aided precoded transmission under realistic conditions such as the power constraints and satellite-payload non-linearities. We develop a scalable architecture of an SDR platform with the DVB-S2X piloting. The SDR platform consists of two parts: analog-to-digital (ADC) and digital-to-analog (DAC) converters preceded by radio frequency (RF) front end and Field-Programmable Gate Array (FPGA) backend. The former introduces realistic impairments in the transmission chain such as carrier frequency and phase misalignments, quantization noise of multichannel ADC and DAC and non-linearities of RF components. It allows evaluating the performance of the precoded transmission in a more realistic environment rather than using only numerical simulations. We benchmark the performance of the communication standard in realistic channel scenarios, evaluate received signal SNR, and measure the actual channel throughput using LDPC codes.

1 Introduction

The 5th generation of mobile radio communications systems should provide a high level of integration and flexibility between different types of telecommunication networks. Terrestrial and satellite systems historically evolved independently of each other, which results in technological diversity between the networks. The launched 5GPPP research program co-funded by the European Commission is set to work towards a definition of new common standards for 5G networks [1]. The objective of the project METIS 2020 as a part of 5GPPP is to build the foundation for a future mobile and wireless communications system for 2020 and beyond [2]. These standards will allow seamless joint operation of mobile cellular communications and satellite systems as a single service. The use cases of modern satellite communications (SATCOM) systems in 5G networks include increasing coverage of conventional terrestrial cells, facilitating caching through multicast/broadcast data transmission and providing off-load backhauling for unicast user traffic [3, 4]. The DVB-S2X [5] was developed to complement new scenarios for flexible SATCOM integration into 5G and beyond networks.

MIMO precoding techniques are based on the closed-loop approach by employing the retrieved channel state information (CSI) from the user terminals (UTs), requiring a feedback channel from UT back to Gateway (GW). Due to the time-varying channel, the gateway has only access to a delayed version of the channel state information (CSI), which can eventually limit the overall system performance [6–9]. However, in contrast to general multiuser multiple-input multiple-output terrestrial systems, the CSI degradation in multibeam mobile applications has a very limited impact for typical fading channel and system assumptions. Under realistic conditions, the numerical results [7] show that precoding can offer an attractive gain in the system throughput compared with the conservative frequency reuse allocations.

MIMO precoding techniques, which are defined as convex optimisation problems, have to be solved by time-consuming iterative convex optimisation (CVX) or Non-negative least squares (NNLS) solving methods that must fit into a relevant time frame. Recent research advanced on the reduction of the processing times to meet channel requirements [10–12]. ZF precoding was practically demonstrated in [13–15] using real-time signal processing and transmission.

Academic research shows that Precoding techniques in SATCOM potentially allow more efficient spectral utilisation and substantially higher service availability [16–18]. To enable the efficient utilisation of satellite transponders, multiple carriers have to be relayed through a single HPA. However, the non-linear nature of the HPA results in adjacent channel interference and increased Peak-to-average power-ratio (PARP), which limits the expected performance gains [19, 20]. The symbol-level precoding design proposed in [21] allows controlling the instantaneous per-antenna transmit power, thus leading to a reduction of the power peaks, which are detrimental for the aforementioned non-linearity problem. It should be mentioned that this is not possible in the channel-level approach, where the precoder is designed for an entire codeword, including several symbols, hence the transmitted power can be controlled only in average and not symbol by symbol. In the context of non-linear channels, it is also worth mentioning more advanced symbol-level precoding schemes [21–23] which aim at reducing the peak-to-average power ratio (PAPR) of the transmitted waveforms, thus considerably improving their robustness.

In this work, we focus on the implementation of the hardware demonstrator for the closed-loop precoded SATCOM. We describe the design and functionality of the multi-beam DVB-S2X compliant GW, the satellite MIMO Channel Emulator and the set of UTs. We validate the design requirements using reasonable software and hardware resources.

We implemented the physical layer of the DVB-S2X standard using software-defined radio (SDR) techniques using commercial SDR platforms. Developing on SDR allows us to rapidly prototype and deploy the precoded transmission in a more realistic environment rather than using only numerical simulations.

Notation: Upper-case and lower-case bold-faced letters are used

to denote matrices and column vectors. The superscripts $(\cdot)^H$ and $(\cdot)^{-1}$ represents Hermitian matrix and inverse operations.

2 Hardware Demonstrator

In this paper, we present a hardware test-bed to demonstrate closed-loop precoded communications for interference mitigation in multi-beam ultra-high throughput satellite (UHTS) systems. We build the test-bed to demonstrate real-time precoded communications under realistic environments. For this matter, we designed a scalable architecture of the gateway and UTs compatible with the DVB-S2X superframe structure. Fig. 1 shows the block diagram of the demonstrator.

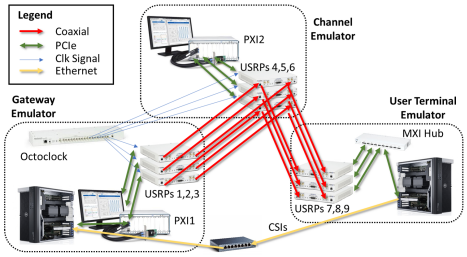

Figure 1: Block diagram of the Hardware Demonstrator

We use the commercially available SDR platform developed by National Instruments (NI). The platform consists of two NI PXI (PCI EXtension for Instruments) 1085 chassis, which allow centralised connection of the set of the NI USRP (Universal Software Radio Peripheral) 2954R and FlexRIO (Reconfigurable IO) 7976R. The NI USRP and FlexRIO have integrated an FPGA (Field-Programmable Gate Array) module Kintex-7 from Xilinx.

The gateway simultaneously transmits 6 precoded signals towards 6 user terminals through a 6×6 multi-beam satellite channel emulator. The channel emulator acquires the gateway signals, applies the impairments of the satellite payload, Gaussian noise, and the multi-beam interference and transmits the signals to the UTs. The UTs estimate the CSI based on the DVB-S2X standard pilots and report the estimated values to the gateway through a dedicated feedback channel over an Ethernet link. The gateway uses this CSI data to compute a Precoding matrix. The table 1 summaries the current capabilities and the final targets of the demonstrator.

Table 1: Parameters of the hardware demonstrator

Parameter	Current
Gateway Channels	6
Sampling frequency	40 MHz
Oversampling factor	4
Gateway TX freq.	1.14 GHz
Channel Emulator RX freq.	1.14 GHz
Channel Emulator TX freq.	1.386 MHz
User Terminal RX freq.	1.386 MHz
Filter roll-off factor	0.2, 0.15, 0.1, 0.05
Forward Error Correction	LDPC
LDPC code rate	1/2, 2/3, 3/4, 4/5

2.1 System Model

We consider a system model, which focuses on the forward link of a multi-beam satellite system. We assume a full frequency reuse scenario, in which all the beams transmit in the same frequency and time. The multi-user interference is mitigated by using the signal precoding technique. The defined the number of transmitting antenna is equal to the total number of users in the coverage area. In this scenario, we consider a 6×6 MIMO channel. In the specified MIMO channel model, the received signal at the i-th terminal is given by $y_i = \mathbf{h}_i^\dagger f(\mathbf{x}) + n_i$, where $\mathbf{h}_i^\dagger$ is a 1×6 vector representing the complex channel coefficients between the i-th terminal and the 6 antennas of the transmitter, $\mathbf{x}$ is defined as the 6×1 vector of the transmitted symbols of DVB-S2X superframe at a certain symbol period and n_i is the independent complex circular symmetric (c.c.s.) independent identically distributed (i.i.d) zero mean Additive White Gaussian Noise (AWGN) inserted to the i-th terminal's receive signal. The function $f(\mathbf{x})$ represents the non-linear behaviour of the satellite channel.

Looking at the general formulation of the received signal, which includes the whole set of terminals, the signal model is

$$\mathbf{y} = \mathbf{H} f(\mathbf{x}) + \mathbf{n} = \mathbf{H} f(\mathbf{W}\mathbf{s}) + \mathbf{n}, \tag{1}$$

where $\mathbf{y} \in \mathbb{C}^{6\times 1}$, $\mathbf{n} \in \mathbb{C}^{6\times 1}$, $\mathbf{x} \in \mathbb{C}^{6\times 1}$, and $\mathbf{s} \in \mathbb{C}^{6\times 1}$ and $\mathbf{H} \in \mathbb{C}^{6\times 6}$. We define the block-level precoding matrix $\mathbf{W} \in \mathbb{C}^{6\times 6}$ as

$$\mathbf{W} = \hat{\mathbf{H}}^H \cdot (\mathbf{H} \cdot \mathbf{H}^H)^{-1}. \tag{2}$$

We consider the data symbols $\mathbf{s}$ to be unit variance complex vectors $|s_k| = 1$ for every $k = 1 \dots 6$.

2.2 Gateway

The gateway uses 3 dual-channel NI USRP RIO devices connected to a NI FlexRIO FPGA. A central Host computer configures the RF hardware, flashes the FPGAs, performs clock synchronization and controls the communication buses. To generate 6 coherent IF streams of data synchronized in time and frequency, the 3 USRP RIO devices are connected to a reference clock source. Fig. 2 shows the data flow through the different hardware blocks of the DVB-S2X gateway. The source of data to be transmitted can be selected between a virtual network interface (TUN/TAP) at the host computer (e.g. video broadcast or data transmission) or a Pseudo-Random Data Generator (PRDG) used for testing and validation purposes. The 6 data streams are sent to the 3 USRP RIO and the type of source can be selected independently for each channel. The data interface format of this DVB-S2X gateway follows the "Mode Adaptation input interface" defined in [24], where complete BBFRAME packets are provided together with the corresponding MODCOD configuration. All the functional blocks of the DVB-S2X gateway (scrambling, encoding, framing, modulation, pulse-shaping, etc.), together with the Precoding algorithm, are running in real-time at either the USRPs or the FlexRIO FPGA as shown in Fig. 2. Once the DVB-S2X waveform is conformed and up-converted, it is ready to be transmitted to an actual satellite, which turns to be the satellite Channel Emulator in this work.

Fig. 3 shows the detailed block diagram of a single data stream of the DVB-S2X Gateway. The output of the precoder block is connected to the up-sampling, pulse-shaping and up-conversion stages at the USRP FPGAs. Light color boxes indicate where the block is implemented: yellow at Host, green at USRP FPGA, and read at FlexRIO FPGA. Grey dotted boxes indicate that the blocks have been implemented as a single HLS / VHDL block. Grey solid arrows correspond to data bits, whereas black solid arrows are complex-valued modulated symbols. Black dotted arrows are parameters controlled by the Host. All blocks are DVB-S2X compliant except for the "Encoder Manager", which controls the I/O buffering of the FEC encoder. The Precoder block received the 6 data streams to be able to apply the Precoding matrix before transmitting the beam data to the Channel Emulator.

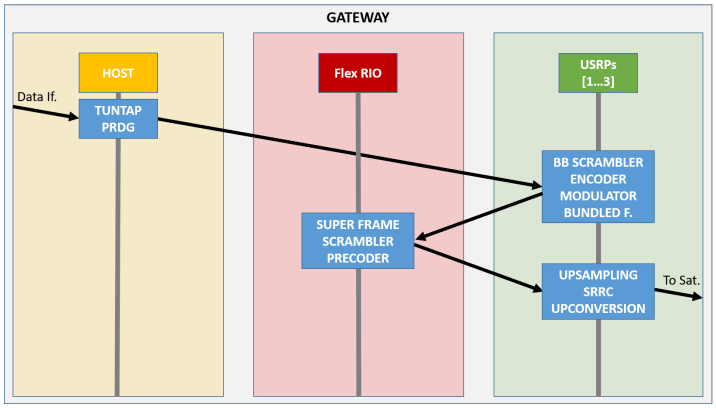

Figure 2: Flow diagram of the 6 data stream DVB-S2X Gateway.

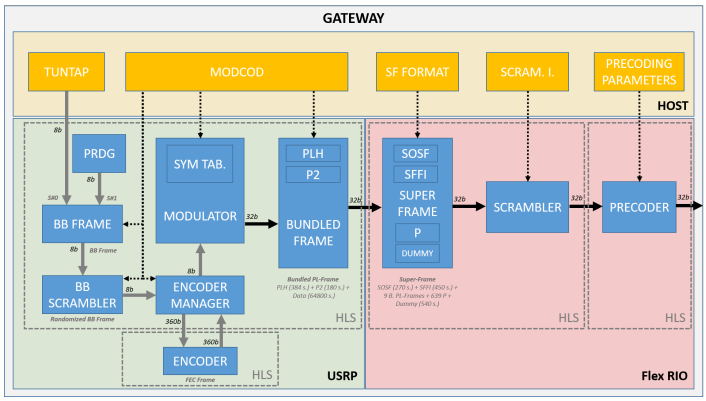

Figure 3: Block diagram of the DVB-S2X Gateway per data stream. The output of Precoder is connected to the up-sampling, pulse-shaping and up-conversion stages at the USRP FPGAs.

The 6 data streams can be independently configured through a dedicated graphical interface as shown in Fig. 4 [13]. The control parameters follow the DVB Standards [24, 25] and they are the following: MODulation and CODing Mode (MODCOD), Super-Frame Format Indicator (SFFI), Index Stream, Index of the Walsh-Hadamard (WH) matrix, scramble flag for Pilots and Start Of Super-Frame (SOSF). We include an extra parameter indicating the Precoding type, which describes the type of the Precoding technique allowed to precode each stream. Rank 0 indicates no Precoding is applied, 1 - Channel based Zero Forcing (ZF) or the Minimal Means Square Error (MMSE) Precoding are used [26], 2 - reserved for the future use, 3 - Symbol-Level Precoding (SLP) techniques [27] are used if possible, otherwise - ZF and MMSE.

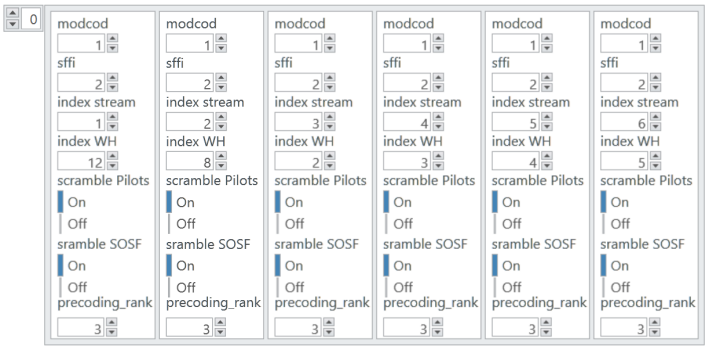

Figure 4: DVB-S2X Gateway configuration graphical interface [13].

As mentioned above, the 6 Precoded beam streams are sent to the 3 NI USRP RIO, where the signals are up-sampled and pulse-shaped with the Square-Root Raised-Cosine (SRRC) function with configurable roll-off factor. The roll-off factor of the filter response is configurable according to the DVB-S2X standard. The implemented roll-off factors are 0.2, 0.15, 0.1 and 0.05. The pulse-shaped signals are then up-converted and transmitted to the RF domain at the configured carrier frequency.

2.3 Channel Emulator

The hardware architecture used for the implementation of the satellite Channel Emulator is similar to the one used for the Gateway. The NI FlexRIO module is in charge of the central signal processing, whereas 3 NI USRP devices take care of the RF processing plus some specific processing for each channel. The block diagram depicted in Fig. 5 describes the different impairments implemented in the Channel Emulator.

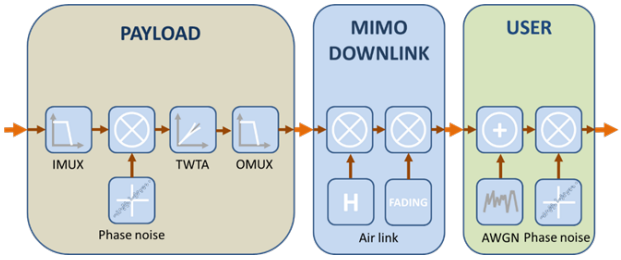

Figure 5: Functional block diagram of the satellite Channel Emulator [28].

The data flow in the Channel Emulator is the following:

- Each NI USRP device acquires 2 of the 6 streams generated by the Gateway at a determined intermediate frequency.
- Each stream is sampled and digital down-converted (DDC) to base-band.
- All the effects that occur in the actual K-/Ka-band are emulated in the payload emulator. This emulation is applied for each stream in the NI USRP FPGA in real-time. The implemented payload impairments are the following:
  - Input Multiplexing (IMUX) and Output Multiplexing (OMUX) filtering FPGA IP blocks. The input and output characteristics of the filters are shown in Fig. 6.
  - Traveling-Wave Tube Amplifier (TWTA) non-linearities. The input and output characteristics of the TWTA IP block correspond to the DVB-S2 standard specifications and are depicted in Fig. 7.
  - Phase noise emulation at K-/Ka-band.
- After the payload emulation, data is sent to the central NI FlexRIO FPGA for the channel processing.
- The channel matrix ($\mathbf{H}$) is jointly applied to all the streams by the MIMO downlink block. The 6×6 channel matrix of complex coefficients is fully configurable. However, a realistic satellite beam pattern illustrated in Fig. 8 is usually running. This pattern is called ESA71 (after the origin and the number of beams). ESA71 makes use of the Ka-band exclusive band 19.7 to 20.2 GHz. We consider a scenario of full frequency reuse, where the same frequency band is used at every beam. We can simultaneously select up to 6 user terminals in the coverage area and generate realistic channel coefficients.
- The resulting signals with channel interference are sent back to the corresponding NI USRP device.
- Back to the NI USRP FPGA, user impairments such as Additive White Gaussian Noise (AWGN) and Low-Noise BLock (LNB) phase noise are emulated. We designed and build a customized AWGN generator [29] with configurable amplitude.

- Once the user impairments are applied, the signal is upsampled and sent through the RF outputs. The RF inputs and outputs of the channel emulator operate at different carrier frequencies depending on the desired configuration.

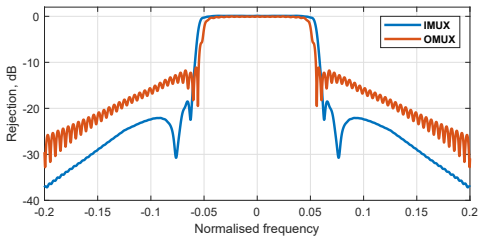

Figure 6: IMUX and OMUX filter characteristics of the Channel Emulator [13].

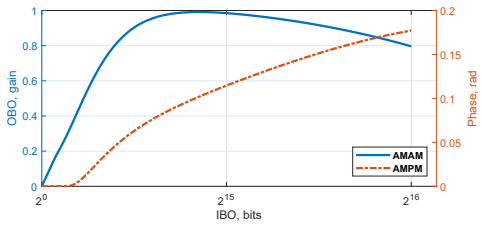

Figure 7: TWTA AM/AM and AM/PM characteristics of the Channel Emulator [13].

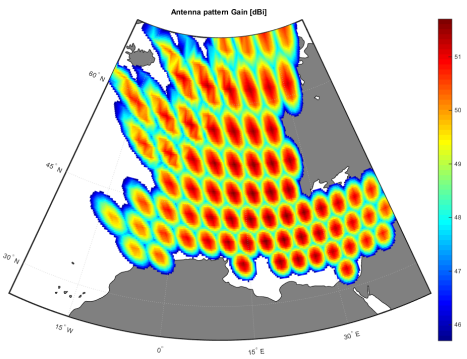

Figure 8: ESA71 beam pattern and antenna performance [13].

2.4 User Terminal

The User Terminals (UT) are the are ground-based users which collect the transmitted waveforms and recover the transmitted data bits. The UTs are capable of recovering format 2 and 3 DVB-S2X super-frames. The UTs are implemented using the same SDR platform used in for the Gateway and the Channel Emulator, the USRP RIO NI2944. We use each of these USRP to implement two independent UTs using the two RF inputs of the device. The FPGA inside the SDR platform performs signal processing for the two UT chains and communicates with the host computer [30]. Fig. 9 shows a simplified functional block diagram of the DVB-S2X Receiver IP block used in the UTs [30].

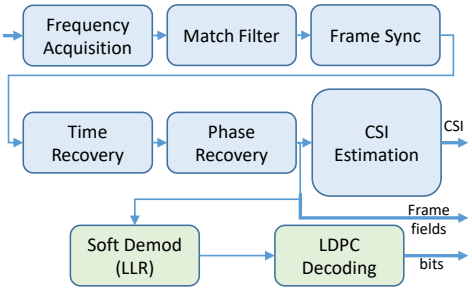

Figure 9: Functional diagram of the DVB-S2X receiver block in the UT

As explained in [30], these processes include frequency acquisition, matched filtering, time synchronization, frame (including Super-Frame) synchronization, fine phase tracking, and CSI estimation. The UT provides the CSI information to the central Gateway using a custom feedback channel. For robust communication under noisy channels, the UT performs the Forward Error Correction (FEC) of the recovered DVB-S2X encoded payloads symbols. As shown in Fig. 9 the information bits are recovered using a Logarithmic Likelihood (LLR) for the incoming symbols and a Low Density Parity-Check Decoder.

2.4.1 LLR demapper

The LLR values required by the Decoder IP block are generated in another independent FPGA IP block. As it is well known the classical formulation of the LLR is:

$$LLR_j = \ln\left[\frac{\sum_{b:b_j=0}\exp\left(\frac{-\|r-c(b)\|^2}{2\sigma^2}\right)}{\sum_{b:b_j=1}\exp\left(\frac{-\|r-c(b)\|^2}{2\sigma^2}\right)}\right], \quad (3)$$

where σ^2 is the noise variance, r is the received complex symbol, and c is the nominal constellation complex point. Thus, to compute the j-th bit LLR from r, one may compute the squared distance to each of the constellation points, separating those constellation points that have a 0 in bit j from those that have a 1. A common approximation to the LLR is to approximate each sum in 3 by its largest term, i.e., by using only the nearest constellation point that has $b_j = 0$ in the numerator, and the nearest neighbor that has $b_j = 1$ in the denominator. This approximation is mathematically expressed in the following equation:

$$c_0 \triangleq c\left(\arg\min_{b:b=0}\|r-c(b)\|^2\right), c_1 \triangleq c\left(\arg\min_{b:b=1}\|r-c(b)\|^2\right) \quad (4)$$

Substituting 4 in 3 and after some algebraic computations we get the approximate LLR expression as

$$LLR_j = \frac{1}{2\sigma^2}\left(\|r-c_1\|^2 - \|r-c_0\|^2\right), \quad (5)$$

Equation 5 is what is used in the FPGA. We evaluated the use of the precise LLR formulation, however, we did not implement this in practice since its added complexity does not payoff in terms of the final error correction performance. Currently, the LLR block is capable of computing the LLR values from modulations ranging from QPSK up to 32APSK. The LLR values obtained from these computations are quantized into 6-bit fixed-point values, represented as five bits for the integer part (one of these 5 bits is used for the sign) and one bit for the fractional part of the LLR, so obtaining values that range from -15.5 up to +15.5. For this particular case, the quantization is implemented in a saturated and symmetrical fashion, to avoid the -16 minimum value, which represents a bias in the calculation. In a conventional full-precision (double float) quantization of the LLR values, the estimation of the noise variance does not have a strong impact on the decoder performance. However, for highly quantized LLR values (as in the current case where 6 bits are used), the estimation of the noise variance has a big impact in terms of scaling purposes that is translated directly into error correction performance.

Fig. 10 shows a plot of LLR values obtained from a 32ASPK constellation with an SNR of 14dB and 6 dB respectively from the FPGA block.

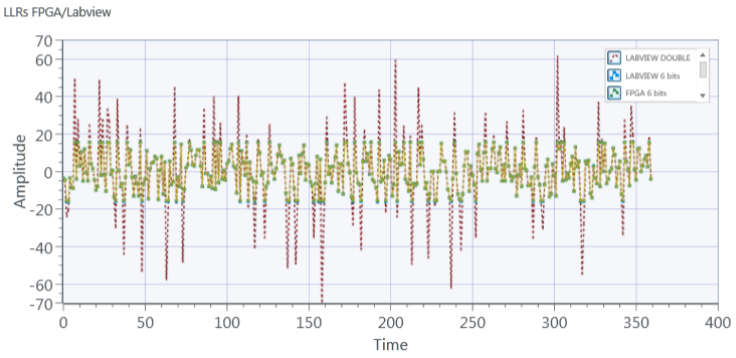

Figure 10: LLR outputs of the FPGA LLR block for a 32APSK constellation at 14dB SNR, compared to simulations in quantized and non quantized versions.

2.4.2 LDPC Decoder

The LDPC Decoder has been implemented with the layered offset MinSum processing scheme. It uses a partial block parallel architecture and can support a sub-matrix size Z=360 for structured-block LDPC codes. The decoder receives the noisy codeword from the demapper in the form of LLR values and performs the decoding procedure. The decoder supports the input of Z=360 parallel LLRs per clock cycle. The fixed-point representation of LLRs is notated as (wl,fr). The wl indicates the total number of bits, including the sign bit, to perform the quantization while fr accounts for the number of bits used for the fractional part of the number. For this FPGA implementation, the decoder is configured to receive LLRs with a representation of (6,1) meaning that total word length is 6 bits, 1 bit is used for sign, 4 bits are used for the integer part and 1 bit is used for the fractional part. This feature is generic, strongly related to the specific decoding algorithm implemented and can be adjusted based on the requirements of BER or FER performance, area cost and throughput requirements. An early termination mechanism has been installed and may be activated by an enable input port. When the mechanism is enabled the decoder controller will terminate the decoding process before the maximum amount of iterations is performed if the current decoded codeword is able to pass parity check test. Table 2.4.2 shows a list of the available LDPC modes with their corresponding code rates and output datarates.

Mode	Mode Description (R = K/N)	Iteration Interval		Datarate (Mbps @50MHz clock)	
		Permut	**Decode**	**I=10**	**I=40**
0	DVB S2 R=1/2 R = 32400/64800	8100	651*I +202	218	94
1	DVB S2 R=1/2 R = 7200/16200	1875	259*I + 67	178	65
2	DVB S2 R=2/3 R = 43200/64800	3600	620*I+ 208	323	113
3	DVB S2 R=2/3 R = 10800/16200	900	320*I + 73	194	58
4	DVB S2 R=3/4 R = 48600/64800	4050	686*I + 216	291	102
5	DVB S2 R=3/4 R = 11880/16200	720	338*I + 79	194	56
6	DVB S2 R=5/6 R = 54000/64800	1800	1210*I + 232	229	64
7	DVB S2 R=5/6 R = 13320/16200	576	342*I + 91	198	56

Table 2: List of available LDPC decoding modes, code rates and output datarates

The LDPC Decoder supports also double buffering at the input. Because of the layered offset MinSum processing scheme, an additional codeword permutation is required, before the actual decoding can begin. Depending on the time needed to perform this permutation and on the number of decoding iterations performed, the total decoder processing delay may vary. Because of the double buffering mechanism at the input, for the first input codeword there is a fixed processing delay penalty due to the permutation circuit, which then for high enough number of decoding iterations is masked for all the subsequent codewords after the first one. To provide a more explanatory example let us consider the Mode 0 of Table 2.4.2. In this particular case, the permutation circuit has a fixed processing delay of 8100 clock cycles, since the first codeword must first be completely uploaded before it is permutated to perform the layered decoding. The actual decoding time, once the permutation is completed, is equal to 651*I+202 clock cycles (with I indicating the actual number of decoding iterations). So it means that the first decoded word comes out after 8100 + 651*I + 202 clock cycles. For the second codeword, if the number of decoding iteration I is greater than 12, since the permutation circuit works in parallel with the actual decoding process, it is possible to save the permutation time and so the decoded word will come out after 651*I+202 clock cycles (the actual decoding time). The permutation time can be reduced implementing a higher level of parallelism, but this will imply the need to use much more hardware resources. The LDPC Decoder was tested for a single link under a SNR controlled environment and the Frame Error Performance was verified. The results are summarized in Table 2.4.2.

MODCOD	Es/No (@ FER = 10^{-5})
QPSK R=1/2	1.25
QPSK R=3/4	4.14
QPSK R=5/6	5.29
8PSK R=1/2	5.86
8PSK R=2/3	7
8PSK R=3/4	8.11
8PSK R=5/6	9.61
16APSK R=1/2	9.16
16APSK R=2/3	10.35
16APSK R=3/4	11.50

Table 3: Experimentally validated LDPC performance in terms of Es/No for different MODCODs for a FER operating point of 10^{-5}.

2.5 Resource Occupation in FPGAs

The described communication equipment is heavily based on FPGA code acceleration and parallel computing. The FPGA resources in the used equipment (NI USRP 2954-R and NI FlexRIO 7976R) are limited, thus the design of the functional FPGA blocks is a compromise between the functionality, the resource occupation, and the data throughput. Table 4 is the summary of the actual FPGA resource occupation of each part of the hardware demonstrator. The occupation percentage of the Digital Signal Processors (DSP), Block RAMs (BRAMS), lookup tables (LUT) is manageable for the planned functionality. Hence, we can notice that the Slices (Each slice contains four LUTs and eight flip-flops) occupation is very hight at the user terminal. The complexity of the terminal is much higher than the one with a single receiver. It is evident that if we want to include the LDPC decoder functionality of the terminal we need to deploy only one DVB-S2X receiver per USRP node.

Table 4: FPGA Xilinx Kintex-7 (410TFFG-2) Resource Occupation

	DSP48E	BRAM	LUT	Slices
Gateway node	15%	64%	25%	42%
Gateway FlexRIO	34%	23%	26%	41%
Channel Emu. node	34%	56%	32%	49%
Channel Emu. FlexRio	9%	27%	31%	43%
UT 1 RX	31%	22%	36%	63%
UT 2 RXs	40%	23%	57%	83%
UT 1 RX w. LDPC	31%	64%	78%	95%

3 Conclusion

In this paper, we present the hardware demonstration of the fully DVB-S2X standard-compliant communication system. The hardware demonstrator is a full-chain closed-loop communication system with multi-beam gateway transmitter, MIMO channel emulator and receiver terminals with real-time CSI estimation. With the hardware demonstrator, we can experimentally validate the new communication standard in the realistic scenario and to demonstrate the full frequency reuse in multi-beam satellite communications.

4 Acknowledgments

This research was partially supported by the European Space Agency Research Project: Live Satellite Precoding Demonstration (LiveSatPreDem); ESA Contract Number 4000122451/18/NL/NR, and by the Fonds National de la Recherche (FNR), Luxembourg, FNR projects: FNR11689919 "COHESAT: Cognitive Cohesive Networks of Distributed Units for Active and Passive Space Applications" and AFR-PPP grant No. FNR11481283 "End-to-end Signal Processing Algorithms for Precoded Satellite Communications".

References

[1] 5G-Infrastructure-Association. EC H2020 5G Infrastructure PPP - Prestructuring Model RTD & INNO Strands. 2014.

[2] 5G-PPP. Mobile and Wireless Communications Enablers for Twenty-twenty (2020) Information Society. 2014.

[3] 3GPP TR 38.811 V15.0.0. 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Study on New Radio (NR) to support non terrestrial networks (Release 15). June 2018.

[4] B. Evans, O. Onireti, T. Spathopoulos, and M. A. Imran. The Role of Satellites in 5G. In *2015 23rd European Signal Processing Conference (EUSIPCO)*, pages 2756–2760, Aug 2015.

[5] Morello Alberto and Mignone Vittoria. DVB-S2X: The New Extensions to The Second Generation DVB Satellite Standard DVB-S2. *International Journal of Satellite Communications and Networking*, 34(3):323–325.

[6] M. Medra, Y. Huang, W. Ma, and T. N. Davidson. Low-Complexity Robust MISO Downlink Precoder Design Under Imperfect CSI. *IEEE Transactions on Signal Processing*, 64(12):3237–3249, June 2016.

[7] M. À. Vàzquez, M. R. B. Shankar, C. I. Kourogiorgas, P. Arapoglou, V. Icolari, S. Chatzinotas, A. D. Panagopoulos, and A. I. Prez-Neira. Precoding, Scheduling, and Link Adaptation in Mobile Interactive Multibeam Satellite Systems. *IEEE Journal on Selected Areas in Communications*, 36(5):971–980, May 2018.

[8] A. Tato, S. Andrenacci, S. Chatzinotas, and C. Mosquera. Link Adaptation and Carriers Detection Errors in Multibeam Satellite Systems with Linear Precoding. In *2018 9th Advanced Satellite Multimedia Systems Conference and the 15th Signal Processing for Space Communications Workshop (ASMS/SPSC)*, pages 1–8, Sep. 2018.

[9] A. Haqiqatnejad, F. Kayhan, and B. Ottersten. Robust Design of Power Minimizing Symbol-Level Precoder under Channel Uncertainty. In *2018 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6, Dec 2018.

[10] Alireza Haqiqatnejad, Farbod Kayhan, and Bjorn Ottersten. Robust SINR-Constrained Symbol-Level Multiuser Precoding with Imperfect Channel Knowledge. *arXiv preprint arXiv:1903.03371*, 2019.

[11] Alireza Haqiqatnejad, Farbod Kayhan, and Bjorn Ottersten. An Approximate Solution for Symbol-Level Multiuser Precoding Using Support Recovery. *arXiv preprint arXiv:1903.03377*, 2019.

[12] J. Krivochiza, J. Merlano Duncan, S. Andrenacci, S. Chatzinotas, and B. Ottersten. FPGA Acceleration for Computationally Efficient Symbol-Level Precoding in Multi-User Multi-Antenna Communication Systems. *IEEE Access*, 7:15509–15520, 2019.

[13] Juan Duncan, Jevgenij Krivochiza, Stefano Andrenacci, Symeon Chatzinotas, and Björn Ottersten. Hardware Demonstration of Precoded Communications in Multi-Beam UHTS Systems. In *36th International Communications Satellite Systems Conference (ICSSC)*, Oct 2018.

[14] Jevgenij Krivochiza, Juan Carlos Merlano-Duncan, Stefano Andrenacci, Symeon Chatzinotas, and Bjrn Ottersten. Computationally and Energy Efficient Symbol-Level Precoding Communications Demonstrator. *Physical Communication*, 28:108 – 115, 2018.

[15] J. C. Merlano-Duncan, J. Krivochiza, S. Andrenacci, S. Chatzinotas, and B. Ottersten. Computationally Efficient Symbol-Level Precoding Communications Demonstrator. In *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pages 1–5, Oct 2017.

[16] N. Mazzali, S. Boumard, J. Kinnunen, B. Shankar M. R., M. Kiviranta, and N. Alagha. Enhancing Mobile Services with DVB-S2X Superframing. *International Journal of Satellite Communications and Networking*, 0(0).

[17] S. Jiang, F. Gong, and X. Chen. A Low-complexity Soft Demapper for 128APSK of DVB-S2X. In *2016 8th International Conference on Wireless Communications Signal Processing (WCSP)*, pages 1–4, Oct 2016.

[18] J. Lucciardi, P. Potier, G. Buscarlet, F. Barrami, and G. Mesnager. Non-Linearized Amplifier and Advanced Mitigation Techniques: DVB-S2X Spectral Efficiency Improvement. In *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, pages 1–7, Dec 2017.

[19] Shree Krishna Sharma, Symeon Chatzinotas, and Pantelis-Daniel Arapoglou. *Satellite Communications in the 5G Era (Telecommunications)*. The Institution of Engineering and Technology, 2018.

[20] Symeon Chatzinotas, Bjorn Ottersten, and Riccardo De Gaudenzi. *Cooperative and Cognitive Satellite Systems*. Academic Press, Inc., Orlando, FL, USA, 1st edition, 2015.

[21] Danilo Spano, Symeon Chatzinotas, Stefano Andrenacci, Jens Krause, and Bjrn Ottersten. Per-Antenna Power Minimization in Symbol-Level Precoding for the Multibeam Satellite Downlink. *International Journal of Satellite Communications and Networking*, 37(1):15–30, 2019.

[22] D. Spano, M. Alodeh, S. Chatzinotas, and B. Ottersten. Papr Minimization Through Spatio-Temporal Symbol-Level Precoding for the Non-Linear Multi-User MISO Channel. In *2018*

IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 3599–3603, April 2018.

[23] D. Spano, M. Alodeh, S. Chatzinotas, J. Krause, and B. Ottersten. Spatial PAPR Reduction in Symbol-Level Precoding for the Multi-Beam Satellite Downlink. In *2017 IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 1–5, July 2017.

[24] ETSI. Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications; Part 1 (DVB-S2). 2014.

[25] ETSI. Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications; Part 2: DVB-S2 Extensions (DVB-S2X). 2015.

[26] C. B. Peel, B. M. Hochwald, and A. L. Swindlehurst. A Vector-Perturbation Technique for Near-Capacity Multiantenna Multiuser Communication-Part I: Channel Inversion and Regularization. *IEEE Transactions on Communications*, 53(1):195–202, Jan 2005.

[27] M. Alodeh, D. Spano, A. Kalantari, C. G. Tsinos, D. Christopoulos, S. Chatzinotas, and B. Ottersten. Symbol-Level and Multicast Precoding for Multiuser Multiantenna Downlink: A State-of-the-Art, Classification, and Challenges. *IEEE Communications Surveys Tutorials*, 20(3):1733–1757, thirdquarter 2018.

[28] Nicola Maturo, Juan Duncan, Jevgenij Krivochiza, Jorge Querol, Danilo Spano, Symeon Chatzinotas, and Björn Ottersten. Demonstrator of Precoding Technique for a Multi-Beams Satellite System. In *8th ESA International Workshop on Tracking, Telemetry and Command Systems for Space (TTC)*, Sep 2019.

[29] C. Politis, S. Maleki, J. M. Duncan, J. Krivochiza, S. Chatzinotas, and B. Ottesten. SDR Implementation of a Testbed for Real-Time Interference Detection With Signal Cancellation. *IEEE Access*, 6:20807–20821, 2018.

[30] J. Duncan, J. Krivochiza, S. Andrenacci, S. Chatzinotas, and B. Ottersten. Hardware demonstration of precoded communications in multi-beam uhts systems. In *36th International Communications Satellite Systems Conference (ICSSC 2018)*, pages 1–5, Oct 2018.