



User Terminal Wideband Modem for Very High Throughput Satellites

Steven Kisseleff, Nicola Maturo, Symeon Chatzinotas, Helge Fanebust, Bjarne Rislow, Kimmo Kansanen, Matthieu Arzel, Hans Haugli

► To cite this version:

Steven Kisseleff, Nicola Maturo, Symeon Chatzinotas, Helge Fanebust, Bjarne Rislow, et al.. User Terminal Wideband Modem for Very High Throughput Satellites. ICSSC 2019: 37th International Communications Satellite Systems Conference, Oct 2019, Okinawa, Japan. hal-02433769

HAL Id: hal-02433769

<https://hal-imt-atlantique.archives-ouvertes.fr/hal-02433769>

Submitted on 9 Jan 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

User Terminal Wideband Modem for Very High Throughput Satellites

Steven Kisseleff[★], Nicola Maturo[★], Symeon Chatzinotas[★], Helge Fanebust[▲], Bjarne Rislow[▲], Kimmo Kansanen[•],
Matthieu Arzel[◇], and Hans C. Haugli[▽]

[★] Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg, Luxembourg;

[▲] WideNorth, Lysaker, Norway;

[•] Department of Electronic Systems, Norwegian University of Science and Technology, Trondheim, Norway

[◇] Département Electronique, IMT-Atlantique Bretagne-Pays de la Loire, Brest, France;

[▽] Space Norway, Oslo, Norway.

Abstract—The continuous increase of the demand for high data rate satellite services has triggered the development of new high-end satellite modems, which are capable of supporting a bandwidth of up to 500 MHz. For commercial application, the downlink from low earth orbit (LEO) sensors and observation satellites is of a special interest. Such satellites should be capable of recording gigabytes of data and transferring it to the ground stations within a few minutes, since the satellite is only visible for a short time at such low altitudes. This implies a very fast and reliable information processing at the terminal. For this, it would be beneficial to utilize the entire 1500 MHz spectrum of the extended Ka-band. In this context, the design of the modem architecture is very challenging. This problem is addressed in this paper for the first time. We develop a new terminal modem architecture, which is expected to support a data rate in the range between 25 Msps and 1400 Msps. Through this, the receiver can easily adapt to changes of the data rate according to the traffic requirements. Furthermore, a simulator tool is developed, which is used for a numerical performance evaluation of the individual components and of the whole system.

I. INTRODUCTION

The increasing demand for high data rates in commercial applications of satellite communication, such as the downlink from low earth orbit (LEO) sensors and observation satellites, motivates the design of novel terminal modems, which are capable of efficiently operating in the extended Ka-band and reliably processing the received data. Here, the main challenges for the modem design appear in the context of extremely large signal spectrum, which can be up to 1.5 GHz. The development of the terminal modem that supports up to 500 MHz for commercial high data rates has been addressed in various projects, e.g. [1], [2], [3]. In contrast, we impose even more challenging and future oriented requirements in this work by assuming a substantially wider target signal bandwidth, i.e. 1.5 GHz, and much higher user data rate of up to 1.4 Gbps (with a root-raised cosine filter and excess bandwidth of at most 5%). In this context, the design of the modem architecture faces the following challenges:

This work has been supported by European Space Agency (ESA) in the framework of project AO8908. Opinions, interpretations, recommendations and conclusions presented in this paper are those of the authors and are not necessarily endorsed by the European Space Agency.

- Parallel processing is required in order to support high baud rates. However, this may lead to large and expensive FPGAs.
- It may not be possible to utilize the optimum signal processing and synchronization algorithms due to the computational complexity and corresponding delays. Hence, a careful selection of the algorithms is required, which implies frequent trade-offs between the performance and the complexity.
- The modem should be able to operate at a low signal-to-noise ratio (SNR), e.g. at -2 dB. The modem operation in such noisy environments is very challenging for the considered scenario, since there are additional limitations of the processing power due to the high baud rates.
- The modem should support high MODCODs based on high order symbol constellations, e.g. 16APSK, 32APSK and 64APSK.

In order to avoid packet loss, the terminal synchronization needs to be very accurate and fast [4], [5]. However, due to the computational complexity, it may not be possible to apply the conventional high-performance synchronization methods here. Hence, less complex and correspondingly less accurate methods are preferable, which implies that a new receiver architecture needs to be specifically designed in order to take into account the individual constraints related to these low-complexity methods.

In this paper, we propose a new modem architecture, which is expected to support a data rate in the range between 25 Msps and 1400 Msps. This architecture enables a quick adaptation of the receiver to changes of the data rate according to the traffic requirements. In particular, we address the selection and analysis of the following signal processing components:

- timing synchronization,
- equalization,
- frequency offset compensation and tracking,
- frame synchronization,
- demodulation/decoding.

Furthermore, we develop a simulator tool, which is used for a

numerical performance evaluation of the individual components and of the whole system.

This paper is organized as follows. In Section II, the system model is described and the implications for the design of a wideband modem are explained. The design of the individual modem components including frequency, timing, and frame synchronization, equalization and demodulation is discussed in Section III. The numerical results based on the developed simulator tool are presented in Section IV. Finally, Section V concludes the paper.

II. SYSTEM MODEL

The very large bandwidth and baud rate poses many challenges for the design of the terminal modem. In addition, some of the less known effects and hardware impairments become substantial and make it difficult to fulfill the requirements of the system. In the following, we address these challenges, impairments and requirements.

A. Challenges and Impairments

The following challenges are especially crucial for the design of the modem:

- 1) parallel signal processing due to extremely high baud rates, which requires large FPGAs;
- 2) trade-offs between performance, latency and complexity;
- 3) high frequency selectivity due to varying amplitude and phase responses over the large signal bandwidth.

In addition, the system performance can degrade due to the following impairments:

- 1) Carrier frequency offset and drift,
- 2) Timing offset,
- 3) Clock frequency offset and drift,
- 4) Frequency selectivity of the cables.

For the carrier frequency offset and drift, we assume ± 3 MHz and ± 5 kHz/s, respectively. Depending on the selected baud rate, which is between 25 Msps and 1400 Msps in this work, the maximum offset can be up to 12% and 0.2%, respectively. These values indicate that in absence of an accurate frequency synchronization, the frequency error would be too large for a feasible frame lock.

The constant timing offset is modeled as a random variable, which is uniformly distributed between $-0.5 \cdot T$ and $0.5 \cdot T$, where T is the symbol duration, since this offset mostly depends on the switching time, which is uniform in this range.

The impairments 1)-2) and 4) are well-known problems addressed by various works in the past. In general, most of these impairments can be dealt with using the traditional methods of receiver synchronization or equalization. However, a clock frequency offset has rarely been considered. In fact, this offset may not only result from the hardware impairment of the on-board clock, but from the spectrum spreading as a side effect of the Doppler shift. This imperfection has rarely been addressed in the literature due to a typically assumed much lower signal bandwidth, such that the spectrum spreading is not crucial even with a very heavy Doppler shift. However, in the considered

scenario, the resulting clock frequency offset leads to a timing drift, which may lead to a significant performance degradation. According to our estimations, this clock frequency offset due to the Doppler shift is at most ± 30 ppm for the target application, i.e. LEO satellites and maximum baud rate of 1400 Msps. The respective change of the Doppler shift, which comes along with the motion of the satellite, lead to a clock frequency drift, which corresponds to a second order timing drift. This drift is expected to be below ± 1 ppm/s.

Furthermore, the frequency selectivity of the cables is not negligible due to the large bandwidth and the corresponding difference between the lowest and the highest signal frequency within the frequency band. We employ a cable slope model in order to take this effect into account. A cable carries the received downmixed signal to the baseband unit. This process can be viewed as an additional frequency-selective filtering. Here, we assume that the cable slope can be up to -10 dB over the whole frequency band. Note, that the received noise undergoes this filtering as well.

In addition, a slight performance degradation can occur due to a possible I-Q imbalance in the hardware of the terminal. However, this effect can be pre-compensated before the main operation of the terminal. Correspondingly, we omit this effect from our consideration.

B. Requirements

As mentioned earlier, we aim at providing the service at a wide range symbol rates, i.e. between 25 Msps and 1400 Msps. For a higher flexibility of the system, we need to ensure a possibility of changing the symbol rate "on the fly". Hence, the system should be able to work with any symbol rate using the same methods and possibly the same parameters.

For the air interface, we assume the DVB-S2X standard (cf. [6]) with the superframing format IV [7]. This format enables some advanced functionalities, e.g. beam hopping, which can be useful in future scenarios and applications. The main properties of this superframe format, that are important for our application, are the equidistant position of the pilot fields and an extended header. The header contains in total 1440 known symbols including an indicator of the superframe format, which can be assumed to be known as well. The pilot fields contain 36 known symbols repeated after every 1440 data symbols. Both header and pilots are modulated via QPSK modulation.

In this work, we focus on very harsh signal propagation conditions. Correspondingly, the system should in principle operate in a wide range of signal-to-noise ratios, e.g. E_s/N_0 between -10 dB and 30 dB. However, for a more practical application, we stick to the DVB-S2X standard [6] and employ the MODCODs accordingly together with the target E_s/N_0 values, which correspond to each of these MODCODs. In particular, we select the following schemes of phase-shift keying (PSK) modulation: QPSK, 8PSK and 16 APSK. The target E_s/N_0 is selected between -2 dB and 13 dB. These values are assumed in order to guarantee a sufficient reliability of symbol detection, i.e. very low frame error rates (FER) below 10^{-5} , in the case of perfect synchronization.

The residual frequency error after the offset compensation is expected to be below 5 kHz. This corresponds to $2 \cdot 10^{-4}$ and $3.6 \cdot 10^{-6}$ of the respective baud rates above. Hence, an extremely accurate frequency synchronization is required in order to almost completely mitigate the frequency offset.

For the timing error correction, there is no specific requirements. However, in order to enable both frequency and frame synchronization, the timing offset and drift compensation should be sufficiently accurate. Hence, we assume that the residual timing offset should be below 10% of the symbol interval. In this case, the impact of the symbol misalignment on the frequency and frame synchronization is relatively low. Although the 10% of the symbol interval can potentially lead to a significant performance degradation in terms of symbol detection due to intersymbol interference and magnitude decrease, this effect can be mostly compensated by introducing an equalization block.

III. MODEM DESIGN

A. Architecture

In this work, we consider both non-data-aided (NDA) methods for the initial (coarse) synchronization and data-aided (DA) methods for the fine synchronization. For the DA methods, we rely on a successful frame lock, such that the pilot symbols transmitted along with the data payload can be utilized for a more reliable parameter estimation. In order to cope with the mentioned difficulties and guarantee the fulfillment of the requirements, we propose the following modem architecture, see Fig. 1. After the Analog-to-Digital Converter (ADC), the signal is fed into the adaptive matched filter, which is connected to the timing error detector, such that the estimated timing error is used in order to adjust the coefficients of the matched filter. This method is preferable, since no interpolation of the consecutive samples is needed in order to account for the timing error. After the timing error correction, the signal is fed into the equalizer, which mitigates the distortion imposed by the frequency-selective cables, etc. During the initialization phase, the equalizer is in the NDA mode and has to be switched off, since it is typically very vulnerable to the frequency offset, which is not yet compensated at this point. Hence, the signal is guided to the coarse frequency estimator, which provides an estimate of the frequency offset. This estimate is used in order to compensate (at least) a part of the frequency offset for the subsequent symbols.

After a successful frame synchronization, it is possible to use the structure of the frame in order to enable DA synchronization. As mentioned earlier, we utilize the superframe format IV. Correspondingly, the structure of the frame is known to the receiver and incoming symbols can be de-formatted and de-scrambled. Through this, the data is separated from the pilots and guided to further processing into the FEC decoder via soft log-likelihood calculation. After the decoding, the decoded bits are assembled into a bit stream.

B. Timing synchronization

As described in the previous section, the first block of the synchronization chain is the time synchronization. Because of

the very high symbol rate and the much lower clock rate employed in the potential hardware implementation, the matched filter is implemented in a parallel fashion. The output of the matched filter is then passed to the Timing Error Detector (TED) block that will evaluate the timing error. The TED is an NDA algorithm, which means that the algorithm can be executed without any knowledge of the transmitted signal. Among the various NDA TED methods, we select the Gardner algorithm [8], which is known to perform well even in case of a relatively large frequency offset. The expression applied by Gardner TED algorithm is

$$e(k) = x \left(\left(k - \frac{1}{2} \right) T + \tau \right) [x((k-1)T + \tau) - x(kT + \tau)] + y \left(\left(k - \frac{1}{2} \right) T + \tau \right) [y((k-1)T + \tau) - y(kT + \tau)]. \quad (1)$$

In this expression, $x(kT + \tau)$ and $y(kT + \tau)$ are in-phase and quadrature components of the input signal to the timing error detector and τ is the estimated timing error.

Differently from the classical scheme, where the timing error detection is typically followed by the interpolation of the subsequent symbols, we combine the timing error detection and the matched filter in one signal processing block. The functionality of this block is as follows. Since the filter taps can be stored with a very high oversampling factor, it is possible to update the matched filter according to the estimated timing offset. For this, the matched filter is sampled with various timing offsets and the resulting filter taps are stored in the memory. Then, the index of the best set of filter coefficients is determined using the TED output. This set is loaded from the memory and used for filtering. Hence, the interpolation of the subsequent symbols is implicitly taken care of by adapting the matched filter. Correspondingly, significant complexity savings are achieved.

In order to account for the processing delay within the timing detection loop, we introduce a buffer of 16 symbols, such that the output of the TED algorithm is delayed by 16 symbols before it reaches the loop filter. Through this, we avoid overestimating the performance of the TED.

The main drawback of this approach is the limited resolution of the matched filter, such that the accuracy of the TED is bounded by the number of possible sets of coefficients. In particular, the set, which corresponds to the closest timing offset, is selected, which provides the lower bound for the performance of the error compensation. The worst case of this effect clearly happens when the output value of the TED falls exactly in between two closest sets of the stored filter taps. In this case, the residual uncompensated time offset scales with the reciprocal oversampling factor of the stored filter, e.g. the maximum offset is 1/4 with oversampling factor 2 and 1/8 with oversampling factor 4, etc. Hence, this imperfection can be mitigated by a sufficient oversampling, which leads to a trade-off between the accuracy and the available storage size. According to our investigations, the degradation of the signal quality in terms of SNR becomes negligible even with an oversampling factor as low as 8.

The convergence of the TED algorithm is shown for an example

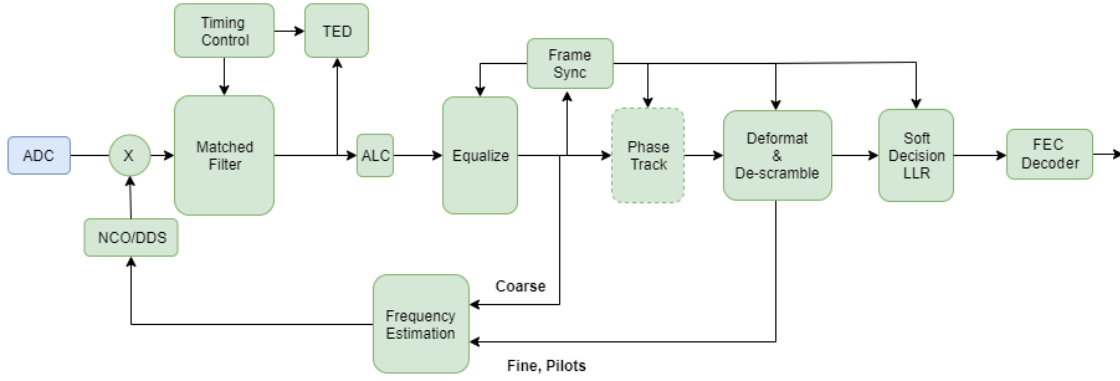


Fig. 1. Proposed modem architecture

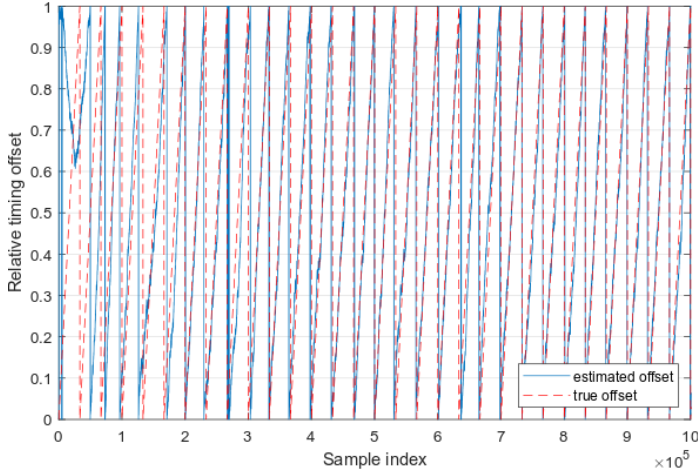


Fig. 2. Example of timing drift tracking using TED.

frame with QPSK modulation in Fig. 2. Here, the initial timing offset is set to zero and we utilize following parameters: damping factor is 3, detector gain is 4 and loop bandwidth is 10^{-3} . We observe, that after some time the algorithm starts to track the timing drift, which results from the clock frequency offset, as explained earlier. Nevertheless, there are still some deviations from the true timing offset, which produces the mentioned degradation of the signal quality. This degradation is typically below 1 dB.

C. Frequency synchronization

1) *Coarse frequency synchronizers*: The traditional method of coarse frequency synchronization is a Quadratic Correlator (QC, [9]). QC comprises two analogue filters and a subsequent correlation of the outputs of the two filters. The so-called balanced (or optimized) QC performs much better than the unbalanced QC (i.e. without optimization of the filters' coefficients). The resulting filters of the balanced QC are matched filter and its derivative. The implementation of two filters imposes a high hardware complexity, which makes the use of the QC impractical for the considered wideband scenario. In addition, QC has been shown to provide a sufficient independency of the

timing error only with oversampled signals.

Another method of coarse frequency synchronization is Delay-and-Multiply (D&M, [10]). Despite a very low complexity of this method, it has a drawback of performing well only in case of oversampling similarly to the QC. Hence, in the considered architecture the coarse frequency synchronizer needs to be deployed after the equalizer, since the output of the equalizer is already in symbol domain. In addition, the performance of both methods is not sufficient, since a very large size of the observation window is required in order to satisfy the system requirements. This leads to a long delay.

Since running the coarse frequency synchronization after the equalizer would guarantee that the frequency synch algorithm wouldn't be subject to the cable slope, we have also investigated an alternative method of coarse frequency synch using the Rife&Boorstyn algorithm, which can operate at the symbol level as well, cf. [11]. This algorithm is based on a peak search in the frequency domain. For this, a Fast Fourier Transform (FFT) method with a sufficiently fine frequency spacing is utilized. This method is substantially more complex than D&M. However, it does not require oversampling and the sufficient accuracy can be achieved with substantially lower sizes of the observation window, such that both processing delay and complexity can be kept low with this method. Also, it is possible to achieve very accurate estimation results already at the coarse synchronization stage, such that a fine frequency synchronization may even be skipped in some cases. This method shows a clear advantage compared to the other methods, such that we select the R&B method for the coarse frequency synchronization in our architecture.

2) *Fine frequency synchronizers*: D&M can be easily adjusted to a DA frequency estimator. Such a DA estimator would be applicable for a fine frequency estimation. However, according to our observation, D&M in fine frequency synchronization mode with a given number of pilots does not reach the performance requirements. Instead, the following methods have been considered first: Mengali&Morelli, Fitz, Luise&Reggiannini and Lovell&Williamson methods [10]. Among these methods, a trade-off between complexity and accuracy has been made, such that Luise&Reggiannini (L&R) method has been selected for a

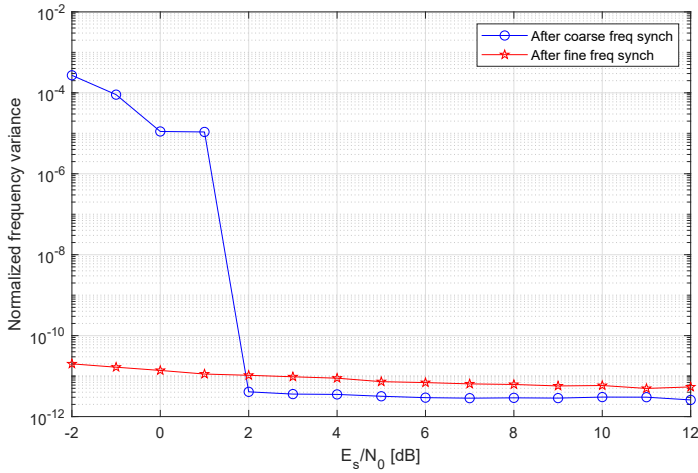


Fig. 3. Frequency error variance after coarse and fine frequency synchronization using R&B method, respectively. A cable slope of -10 dB has been assumed.

deeper analysis. As known from the literature, L&R method can reach the CRB if the number of inner summations is at least half of the window size. Unfortunately, this number of inner summations also impacts the maximum offset range, which can be compensated by the algorithm. Due to a larger residual offset after methods like D&M and QC, this parameter corresponding to the number of summations needs to be selected sufficiently low, such that the CRB cannot be reached and the performance is typically relatively bad. On the other hand, when using R&B for coarse frequency synchronization, the residual offset is usually very low, which makes L&R applicable. However, R&B can be used in DA mode as well and provides an even better performance compared to L&R especially in presence of frequency-selective channels. A distinct advantage of this strategy is that no additional complexity for the implementation of the fine frequency synchronization is required, since R&B is already assumed for the coarse frequency synchronization.

The normalized residual frequency variance after the R&B method in coarse and fine synchronization mode is shown in Fig. 3. Here, we assume a cable slope of -10 dB over the whole frequency band of 1.4 GHz. For the coarse frequency synchronization, we consider a window of 2^{13} symbols, whereas for the fine synchronization, we assume a window size of 1440 symbols, that are associated with the known symbols of the superframe header¹. Obviously, a large residual frequency variance occurs for very low SNR values, which is due to a wrong peak selection in the frequency domain. Furthermore, we observe that the coarse estimation achieves a better performance at high SNR compared to the pilot-based fine estimation. This is due to a larger window size utilized by the coarse estimation as mentioned earlier. Correspondingly, in the higher SNR regime, the use of a fine frequency synchronization is not beneficial. In addition, we observe that the frequency variance with coarse frequency synchronization is approx. $2 \cdot 10^{-12}$. Hence, the resid-

¹At the input of the FFT, we apply also zero-padding, such that the total number of input symbols including zeros is a power of 2. Through this, a better signal resolution in the frequency domain is obtained.

ual frequency offset is in the range of 10^{-6} , which corresponds to a few kHz in case of 1.4 GHz.

After the fine synchronization, we employ a phase tracker in order to compensate the phase offset and possible phase noise. For the phase tracker, we employ a second-order phase-locked loop (PLL). This phase tracker can be employed further in order to fine tune the frequency estimation by calculating the difference between the estimated phase offsets of the consecutive samples and deducing the frequency offset, which caused this relative phase difference. With this strategy, the maximum residual frequency offset reduces to extremely low values below 10^{-8} corresponding to less than 14 Hz at 1.4 GHz, such that the frequency variance is in the order of $10^{-15} - 10^{-16}$.

D. Frame synchronization

The frame synchronization is usually done via correlation of the available pilot symbols with the received signal. In presence of a frequency offset, each symbol experiences a phase rotation, which prevents from applying a straightforward correlation. However, a small frequency offset may not significantly affect the correlation performance. Hence, a non-coherent post-detection integration (NCPDI) approach can be employed (cf. [12], [13]), where a sequence of pilot symbols is split in small blocks. Each block is correlated with the received signal at the respective position of the block. Then, the results of all blocks are combined coherently. Hence, the phase rotation from block to block is eliminated, which leads to the non-coherent integration. This method has been thoroughly investigated in the past. For our design purpose, we utilize the expressions for the false alarm and missed detection probabilities, i.e. \Pr_{fa} and \Pr_{md} , respectively, as well as the so-called CHILD's rule [14], which indicates the maximum number of pilot symbols per block:

$$\Pr_{fa} = e^{-\frac{\delta KL}{P_r}} \sum_{m=0}^{L-1} \frac{1}{m!} \left(\frac{\delta KL}{P_r} \right)^m, \quad (2)$$

$$\Pr_{md} = 1 - \frac{1}{2\Delta f} \int_{-\Delta f}^{\Delta f} Q_L \left(\left| \frac{\text{sinc}(K\Delta f)}{\text{sinc}(\Delta f)} \right| \frac{\sqrt{KL}}{\sigma}, \frac{\sqrt{\delta}}{\sigma} \right) df, \quad (3)$$

$$1 \leq K \leq \frac{3}{8\Delta f T} \text{ (CHILD's rule)}, \quad (4)$$

where K and L are the number of pilots per block and the number of blocks, respectively. P_r stands for the received signal power including the noise with the variance σ^2 and $2\sigma^2 = N_0$ is the noise power density. The threshold for the decision making of the frame acquisition is denoted δ . In addition, $Q_L(\cdot)$ is the Marcum-Q function of order L and Δf is the maximum absolute frequency offset. Using these equations, it is possible to determine the optimal threshold for the frame lock and the optimal number of blocks, i.e. total required number of pilots, such that the target requirements on \Pr_{fa} and \Pr_{md} are satisfied. Since the design of the frame synchronization can be done offline, we find the optimal set of parameters, i.e. K , L and δ , via full search. The results for the total required number of pilots, i.e. $K \cdot L$, is depicted in Fig. 4 for various SNR values and baud rates. We observe, that the required number

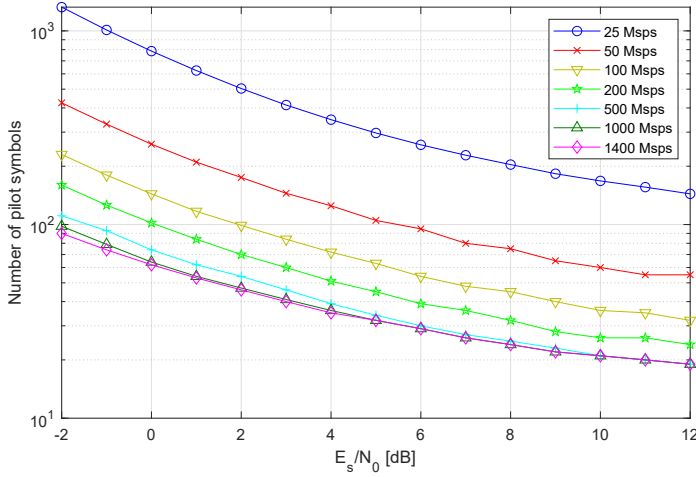


Fig. 4. The total required number of pilots for a frame lock using NCPDI under assumption of 3 MHz frequency offset.

of pilots reduces with increasing baud rate. This is due to a constant frequency offset of 3 MHz, such that the relative offset reduces. Correspondingly, the maximum value of K increases according to the CHILD's rule and the correlation of each window becomes more and more accurate. Also, the number of pilots reduces with increasing SNR, since correlation of each window becomes more accurate and reliable. For -2 dB, the total number of pilots is in the range between 96 and 1338 symbols depending on the baud rate. Fortunately, these values are lower than the number of symbols in the header of a superframe, which is 1440, such that NCPDI can be applied exclusively to the header without taking into account the pilot fields spread across the superframe. Hence, the frame synchronization is straightforward and has a relatively low complexity.

After the frame lock has been acquired, most of the NDA methods can be switched off, i.e. coarse frequency synchronization and blind equalization. Instead, the DA methods are utilized, which make use of the pilots and are therefore more reliable. In particular, fine frequency synchronization and DA equalization are switched on (if needed).

E. Equalization

The main task of the equalization for the envisioned application is to reduce the signal distortion imposed by imperfect transmit/receive filters and cable slope. In this work, the equalization component is placed after the timing error correction and before the frequency synchronization. On the one hand, the equalizer would mitigate the intersymbol interference which results from the residual timing offset. On the other hand, the equalization is needed for an accurate operation of the frequency synchronization, since the peak selection in frequency domain is sufficiently accurate only in presence of a reasonably equalized received signal.

We consider the following methods:

- Normalized Least Mean Square Algorithm (NLMS) [15],
- Normalized Block Least Mean Square Algorithm (NBLMS) [15],

- Normalized Constant Modulus Algorithm (NCMA) [16],
- Normalized Block Constant Modulus Algorithm (NBCMA) as an extension of NCMA.

All four methods are based on a gradient descend method, i.e. the filter coefficients \mathbf{f}_n are updated iteratively in each step n in the direction of the gradient ∇_n , which is calculated from the observed samples. These samples are stored in the signal vector \mathbf{x}_n . The update is done via

$$\mathbf{f}_{n+1} = \mathbf{f}_n + \mu \nabla_n, \quad (5)$$

where μ is a step size in the range between 0 and 1.

The first two methods (NLMS and NBLMS) employ pilot symbols, whereas the last two methods (NCMA and NBCMA) are blind. Furthermore, the symbol-based algorithms (NCMA and NLMS) utilize a symbolwise calculation of the stochastic gradient, whereas the block-based algorithms (NBCMA and NBLMS) employ an approximate gradient calculation, which is done over a block of symbols. Through this, a better stability and performance can be achieved, especially in the low SNR regime due to additional averaging.

The NCMA algorithm utilizes the following gradient in the n th step:

$$\nabla_n = -\frac{\mathbf{x}_n^* y_n}{\|\mathbf{x}_n\|^2} \left(1 - \frac{1}{|y_n|}\right). \quad (6)$$

For the NLMS, the gradient is given by

$$\nabla_n = -\frac{\mathbf{x}_n^* y_n}{\|\mathbf{x}_n\|^2} (y_n - p_n), \quad (7)$$

where p_n denotes the reference signal obtained from the pilot signal. For the two block-based algorithms NBCMA and NBLMS, we obtain

$$\nabla_n = -\frac{1}{L} \sum_{k=n-L}^n \frac{\mathbf{x}_k^* y_k}{\|\mathbf{x}_k\|^2} \left(1 - \frac{1}{|y_k|}\right) \quad (8)$$

and

$$\nabla_n = -\frac{1}{L} \sum_{k=n-L}^n \frac{\mathbf{x}_k^* y_k}{\|\mathbf{x}_k\|^2} (y_k - p_k), \quad (9)$$

respectively. Among these four algorithms, the block-based blind algorithm can be employed during the coarse synchronization phase, i.e. before the frame lock. After the successful frame synchronization, pilot fields can be utilized in order to further improve the performance of equalization. Here, the block-based equalization is preferred again, as explained before, and we select NBLMS.

Depending on the symbol rate, the assumed cable slope has a stronger or weaker effect on the length of the impulse response and correspondingly the equalization performance, which is shown in Fig. 5. Here, we show only the pilot-based equalization via NBLMS method after the convergence of the algorithm. Apparently, the performance degradation in terms of signal quality can be observed, if no equalization is applied, especially with a large symbol rate. Using the NBLMS equalizer, it is possible to compensate the degradation almost completely for the input SNR (before the cable) of up to 12 dB. With larger input SNR, the equalizer is capable of dramatically improving

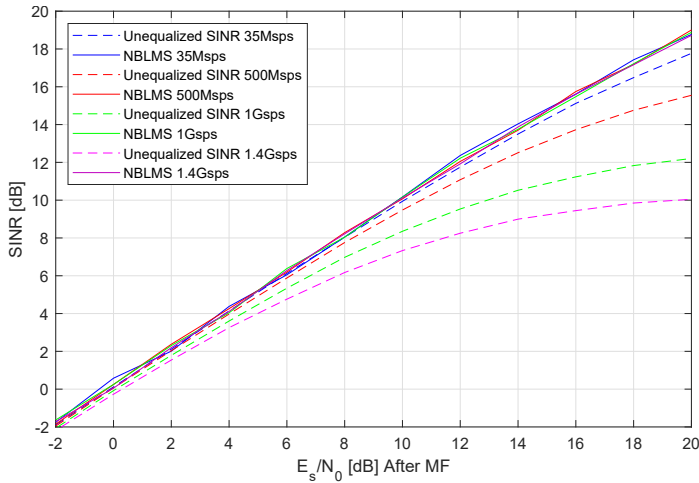


Fig. 5. Performance of NBLMS equalization with FIR filter of length 5 and cable slope of -10 dB.

the performance compared to the case with no equalization in the loop. However, it is not possible to compensate the losses completely. Fortunately, we focus on relatively low SNR values in this work, such that this performance of the equalization is sufficient.

F. Demodulation and decoding

The design of an LDPC decoder for the DVB-S2 and DVB-S2X standards is challenging especially in case of high throughput and very low bit error rate. This is a non-trivial task, especially for an implementation on a reconfigurable circuit. Even though manufacturers such as Xilinx and Intel Programmable Solutions provide more and more processing power with every generation of their chips, which enables higher and higher processing parallelism, the hardware resources still remain limited. In particular, the parallel processing may lead to frequent access conflicts, which pose the main difficulty in current FEC decoder implementations that rely on massive exchanges of information between many thousands of parallel processing units. Access conflicts are known to generate wrong or deprecated messages between processing units which produce altered messages thus letting the FEC decoder diverge from its expected correction capability. In this work, we employ the methods proposed in [17] in order to resolve the memory update conflicts while fulfilling the very low bit error rate performance requirement of the standards.

Another difficulty when targeting high throughput with parallel architectures is the routing congestion on reconfigurable targets. All the messages have to be sized so that they require as less bits as possible without degrading the correction performance, cf. [18]. Through this, a fine matching between code properties and FPGA characteristics can be established and performance degradation due to routing congestion is avoided.

IV. NUMERICAL RESULTS

In this section, the numerical results for the system performance. We assume a cable slope of 0 dB, such that no

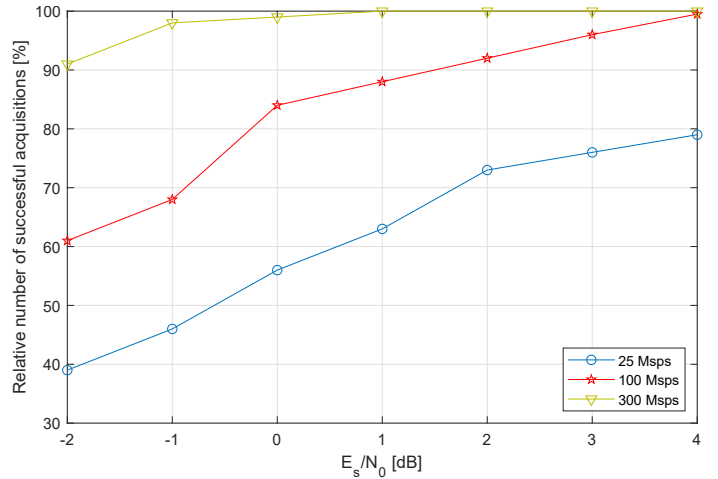


Fig. 6. Relative number of successful frame lock acquisitions.

equalizer is required during the acquisition phase. However, in the proposed architecture, the equalizer can be switched on, if the performance deviates substantially from that one shown below.

We start with the frame lock acquisition. For this we assume that the frame lock has to be acquired within three consecutive superframes, after which the system remains locked for the next five consecutive superframes. The relative number of successful acquisitions is shown in Fig. 6 for the symbol rates of 25 Mpsps, 100 Mpsps and 300 Mpsps, respectively. We can observe a monotonic increase of the number of acquisitions. With a very low signal quality and a relatively large initial frequency offset, e.g. 12% in case of 25 Mpsps, the timing error detection does not work efficiently, such that the peak of the NCPDI either falls below the assumed threshold or even corresponds to a wrong start of the superframe. In the latter case, we detect the wrong frame lock during the FEC decoding. We observe that the frame lock is obtained with 300 Mpsps in more than 90% of cases even at $E_s/N_0 = -2$ dB. On the other hand, for $E_s/N_0 = -2$ dB, no frame lock can be obtained in 40% of cases with 100 Mpsps and in 60% of cases with 25 Mpsps, i.e. a new acquisition attempt is required.

Typically, the signal quality improves substantially after the frame lock, since fine frequency synchronization and phase tracking can be applied. In order to get insight into the signal quality after the phase tracker, an estimate of the SNR is obtained using the Squared Signal-To-Noise Variance (SNV) estimator described in [19]. In our investigations, we observed a proportional increase of the SNR estimate with increasing E_s/N_0 with small fluctuations. The difference between the SNR estimate and the E_s/N_0 typically varies between 0.4 dB and 0.6 dB.

After the acquisition, the system remains locked even in case of sudden variations of the system parameters. In particular, a change of the modulation scheme, e.g. from QPSK to 8PSK or 16APSK does not impact the timing or frequency estimation. Also, we observe that E_s/N_0 needs to drop below -4 dB after

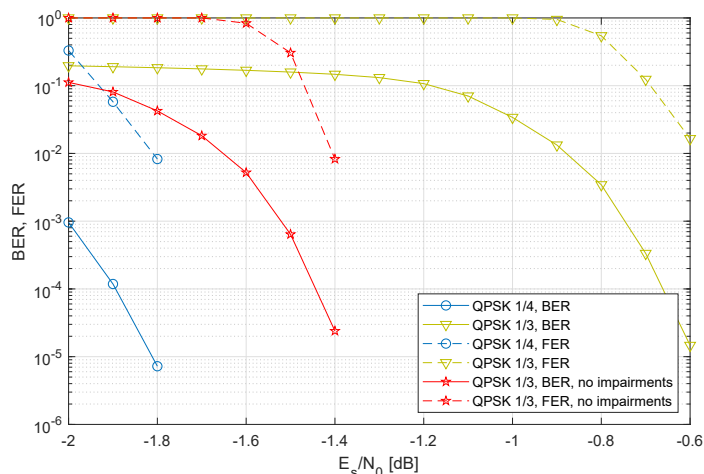


Fig. 7. BER and FER with two MODCODs in low SNR regime.

the acquisition for a loss of frame lock. In case of carrier frequency instability, the relative offset needs to change by at least 12% of the symbol rate. With a symbol rate of 25 Msps, this corresponds to 3 MHz, which is equal to the initially assumed maximum carrier frequency offset.

The decoding performance of the end-to-end superframe transmission in terms of frame error rate (FER) or bit error rate (BER) depends on the selected MODCOD. In Fig. 7, we show the performance of two MODCODs according to [20] during the tracking phase. The first MODCOD corresponds to QPSK modulation with code rate 1/4, whereas the second MODCOD has code rate 1/3. In addition, we show the performance of the second MODCOD in absence of any impairments, i.e. only the received noise has impact on the signal quality and error rate. Apparently, in absence of impairments, we obtain extremely low BER and FER for $E_s/N_0 \geq -1.3$ dB, which is well aligned with the suggested region of operation for this MODCOD, cf. [20]. In presence of both timing and frequency impairments, the required E_s/N_0 to reach the same performance is ≈ -0.5 dB, which is 0.8 dB higher than without impairments. With the first MODCOD, we obtain a significantly better performance in terms of BER and FER, such that the error probability becomes negligible for $E_s/N_0 \geq -1.7$ dB. Without impairments, this MODCOD has an extremely low BER and FER for $E_s/N_0 \geq -2.5$ dB.

V. CONCLUSION

In this paper, we design a terminal modem for very high throughput satellites and focus especially on the receiver synchronization and signal processing. Due to very harsh conditions for the information recovery and due to envisioned low complexity solution, the problem of designing a practical modem architecture appears to be very challenging. Hence, we propose a new architecture, which is capable of dealing with the assumed hardware impairments and satisfying the imposed system requirements. Our simulations have shown that a quick and reliable synchronization can be reached using the proposed architecture. Furthermore, the end-to-end losses in signal quality

are relatively low (≈ 0.8 dB), which indicates a very good performance of synchronization and information detection.

REFERENCES

- [1] "Product Leaflet. Newtec MDM5000 Satellite Modem," Tech. Rep., June 2017, rev. 3.2. [Online]. Available: <http://www.newtec.eu/frontend/files/leaflet/newtec-mdm5000-satellite-modem-r3.2.pdf>
- [2] "RT Logic HDRM High Data Rate Modem," Tech. Rep. [Online]. Available: http://www.rtllogic.com/media/datasheets/rtl_dst_hdrm.pdf
- [3] "Comtech EF Data CDM-760 Advanced High-Speed Trunking and Broadcast Modem," Tech. Rep., June 2017. [Online]. Available: <http://www.comtechefdata.com/products/satellite-modem/cdm-760/>
- [4] L. Lee, M. Eroz, and N. Becker, "Modulation, Coding, and Synchronization for Mobile and Small Satellite Terminals: An Update of the DVB-S2 Standard," in *IEEE 80th Vehicular Technology Conference (VTC2014-Fall)*, Sep. 2014, pp. 1–6.
- [5] E. Casini, R. D. Gaudenzi, and A. Ginesi, "DVB-S2 modem algorithms design and performance over typical satellite channels," *International Journal of Satellite Communications and Networking*, vol. 22, no. 3, pp. 281–318, 2004.
- [6] ETSI EN 302 307-2, "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)," Feb. 2015.
- [7] C. Rohde, H. Stadali, J. Perez-Trufero, S. Watts, N. Alagha, and R. De Gaudenzi, "Implementation of DVB-S2X super-frame format 4 for wideband transmission," in *International Conference on Wireless and Satellite Systems*. Springer, 2015, pp. 373–387.
- [8] F. Gardner, "A BPSK/QPSK Timing-Error Detector for Sampled Receivers," *IEEE Transactions on Communications*, vol. 34, no. 5, pp. 423–429, May 1986.
- [9] A. N. D'Andrea and U. Mengali, "Design of quadricecorrelators for automatic frequency control systems," *IEEE Transactions on Communications*, vol. 41, no. 6, pp. 988–997, 1993.
- [10] U. Mengali, *Synchronization techniques for digital receivers*. Springer Science & Business Media, 2013.
- [11] D. Rife and R. Boorstyn, "Single tone parameter estimation from discrete-time observations," *IEEE Transactions on Information Theory*, vol. 20, no. 5, pp. 591–598, Sep. 1974.
- [12] G. E. Corazza and R. Pedone, "Generalized and average likelihood ratio testing for post detection integration," *IEEE Transactions on Communications*, vol. 55, no. 11, pp. 2159–2171, 2007.
- [13] N. Mazzali, G. Stante, S. M. R. R. Bhavani, and B. Ottersten, "Performance analysis of noncoherent frame synchronization in satellite communications with frequency uncertainty," in *2015 IEEE Symposium on Communications and Vehicular Technology in the Benelux (SCVT)*. IEEE, 2015, pp. 1–6.
- [14] G. Corazza, R. Pedone, and M. Villanti, "Frame acquisition for continuous and discontinuous transmission in the forward link of satellite systems," *International Journal of Satellite Communications and Networking*, vol. 24, no. 2, pp. 185–201, 2006.
- [15] S. S. Haykin, *Adaptive filter theory*. Pearson Education India, 2005.
- [16] C. B. Papadias and D. T. Slock, "On the convergence of normalized constant modulus algorithms for blind equalization," in *Proc. of DSP International Conference on Digital Signal Processing*, 1993.
- [17] C. Marchand and E. Boutillon, "LDPC decoder architecture for DVB-S2 and DVB-S2X standards," in *Proc. of IEEE Workshop on Signal Processing Systems (SiPS)*, 2015, pp. 1–5.
- [18] C. Marchand, L. Conde-Canencia, and E. Boutillon, "Architecture and finite precision optimization for layered LDPC decoders," *Journal of Signal Processing Systems*, vol. 65, no. 2, p. 185, 2011.
- [19] R. Middlestead, *Digital communications with emphasis on data modems: Theory, analysis, design, simulation, testing, and applications*. John Wiley & Sons, 2017.
- [20] ETSI EN 302 307, v1.3.1, "Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications (DVB-S2)," 2013.