



CHALMERS

Chalmers Publication Library

Practical Detection Schemes for Power Efficient Modulation Formats

This document has been downloaded from Chalmers Publication Library (CPL). It is the author's version of a work that was accepted for publication in:

Advanced Photonics Congress (SPPCom) 2013

Citation for the published paper:

Johannisson, P. ; Karlsson, M. (2013) "Practical Detection Schemes for Power Efficient Modulation Formats". Advanced Photonics Congress (SPPCom) 2013(SPM3D.4),

Downloaded from: <http://publications.lib.chalmers.se/publication/181060>

Notice: Changes introduced as a result of publishing processes such as copy-editing and formatting may not be reflected in this document. For a definitive version of this work, please refer to the published source. Please note that access to the published version might require a subscription.

Chalmers Publication Library (CPL) offers the possibility of retrieving research publications produced at Chalmers University of Technology. It covers all types of publications: articles, dissertations, licentiate theses, masters theses, conference papers, reports etc. Since 2006 it is the official tool for Chalmers official publication statistics. To ensure that Chalmers research results are disseminated as widely as possible, an Open Access Policy has been adopted. The CPL service is administrated and maintained by Chalmers Library.

(article starts on next page)

Practical Detection Schemes for Power Efficient Modulation Formats

Pontus Johannisson and Magnus Karlsson

Photonics Laboratory, Chalmers University of Technology, SE-412 96 Göteborg, Sweden

pontus.johannisson@chalmers.se

Abstract: We discuss the detection process for a selection of power efficient formats. The receiver subsystems of dynamic equalization, phase estimation, and data detection are described with emphasis on the case with differential encoding.

OCIS codes: (060.1660) Coherent communications; (060.4080) Modulation; (060.2330) Fiber optics communications

1. Introduction

Coherent optical transmission brings a large freedom in the choice of modulation format and type of forward error correction (FEC). Polarization-multiplexed quaternary phase-shift keying (PM-QPSK) is popular for many reasons but of great importance is the (relative) simplicity of both transmitter and receiver hardware, as well as of the receiver digital signal processing (DSP). Traditionally, the use of Reed–Solomon FEC has been assumed, leading to systems being evaluated as acceptable at a bit error rate (BER) below 3.8×10^{-3} .

However, an optical signal is inherently 4-dimensional (4D) and this has recently been the focus of many studies. Bülow reported POL-QAM [1] and Agrell and Karlsson reported, e.g., the most power efficient 4D modulation format, the polarization-switched (PS-) QPSK [2]. The fundamental parameters for these formats are the spectral efficiency (SE) and the asymptotic power efficiency. However, as the latter corresponds to a low BER, the power efficiency depends on the target BER and the FEC. Therefore, the choice of modulation format and FEC are connected. Unfortunately, the FEC raises an array of new trade-off questions about, e.g., system complexity and DSP power dissipation.

In this paper, we compare a selection of 4D modulation formats in terms of the receiver DSP. In particular we discuss data encoding/detection and the important subsystems of dynamic equalization and phase estimation. We do not address the ambiguity problem, i.e., that the phase and polarization of the received signal is unknown and we assume no use of pilot data or segmentation of the data into blocks.

2. Modulation Formats

We compare five different formats that we have investigated experimentally: PM-QPSK, PM-16-QAM, PS-QPSK [3], 128-SP-QAM [4], and PM-2-PPM-QPSK [5]. All these modulation formats are found in Fig. 1, which shows the SE per polarization as a function of the sensitivity penalty ($1/\gamma$), i.e., the sensitivity loss compared to PM-QPSK at low BER [2]. PS-QPSK and 128-SP-QAM can be constructed by set partitioning [6] of PM-QPSK and PM-16-QAM, respectively. By this operation, the minimum distance between constellation points is increased by $\sqrt{2}$ and the number of constellation points is reduced by 50%, i.e., one bit is lost. Compared to the generating constellations, the asymptotic power efficiency is increased with 1.76 dB for PS-QPSK and 2.43 dB for 128-SP-QAM.

Pulse-position modulation (PPM) increases the dimensionality, which can be used to improve the sensitivity at the cost of a reduced SE [7]. For example, using K -PPM-PM-QPSK, a symbol consists of a PM-QPSK symbol transmitted in one selected slot out of K consecutive ones. However, applying PPM to a 4D symbol is *not* the same as applying polarization multiplexing to a 2D format that uses PPM. We here discuss PM-2-PPM-QPSK which, by individually selecting the position of two (2D) QPSK symbols, carries 3 bits per polarization and pair of neighboring symbols. In fact, PM-2-PPM-QPSK is equivalent to PS-QPSK in terms of SE and γ . We remark that PM-QPSK, PM-16-QAM, and PM-2-PPM-QPSK all can be described as $2 \times 2D$, as they consist of two polarization-multiplexed channels carrying a 2D signal.

2.1. Differential Data Encoding

PM-QPSK and PM-16-QAM can use Gray bit-to-symbol mapping and since errors typically occur between neighboring symbols, a symbol error will cause a single bit error. Neither PS-QPSK nor 128-SP-QAM can be Gray coded since each symbol has too many closest (equidistant) neighbors. In PS-QPSK the number of closest neighbors is six and a 128-SP-QAM symbol has up to 24 closest neighbors. However, in practice we must also address the possibility of cycle slips, as the introduction of an unknown $n \times \pi/2$ phase rotation in one of the polarizations otherwise causes

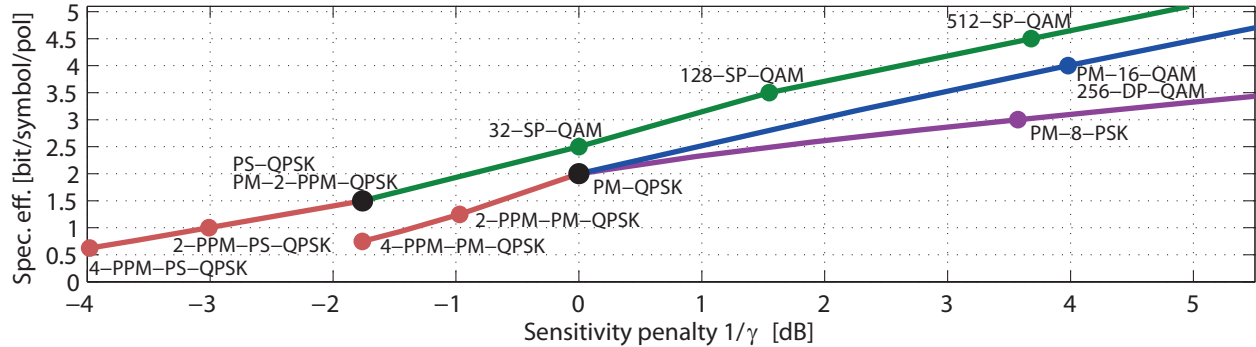


Fig. 1: The SE as a function of the sensitivity penalty at low BER (compared to PM-QPSK) [2]. Included are rectangular QAM (blue), SP formats (green), phase-shift keying (purple), and PPM of the formats marked by black dots (orange). The 32-SP-QAM was recently implemented [8]. In practice, many more parameters are important: Different formats have differently complicated transmitter/receiver implementation and different tolerances to hardware nonidealities, FEC strongly influences both the performance and the DSP complexity etc.

the following data to be lost. We address the cycle slip problem by differential encoding, which is straightforward for PM-QPSK and PM-16-QAM. In the first case this is accomplished with a Gray coded differential rotation between quadrants. In the latter case, two bits are encoded in a rotationally invariant way (using Gray coding within each quadrant) and two bits are encoded in the change of quadrant just as for PM-QPSK [9]. Viewing 128-SP-QAM as a subset of PM-16-QAM, a pragmatic choice is to use the same scheme as for PM-16-QAM [9]. As discussed below, the decoding scheme is compatible with this choice. The case for PS-QPSK is somewhat more difficult. It is true that PS-QPSK can be obtained from PM-QPSK using set partitioning and an alternative name for PS-QPSK is therefore 8-SP-QAM. This is a different but equivalent way of viewing PS-QPSK and it may seem like we can proceed in an analogous way as we did for 128-SP-QAM. However, the conventional constant modulus algorithm (CMA) [10] does not work for PS-QPSK which has led to the development for alternative approaches [11, 12]. Using these algorithms, the resulting signal is in the PS-QPSK configuration, not in the 8-SP-QAM configuration, and conversion between these configurations requires knowledge of the (unknown) relative phase between the polarizations. One alternative way is to introduce a relative time delay between the two polarizations [13]. In this way, the statistics of PS-QPSK become identical to PM-QPSK and the conventional CMA can be used. Notice that the delay must then be longer than the number of FIR filter taps in the CMA in order to remove all correlation. One way to do differential encoding of PS-QPSK is as described in [14], i.e., one bit is encoded in the selection of the polarization and two bits are differentially encoded in the phase of consecutive 4D symbols. A similar approach can be used for PM-2-PPM-QPSK, i.e., one bit is encoded in the selection of the time slot and two bits are encoded differentially in the phase.

3. Detection

At the transmitter side, a selected modulation format can either be realized in hardware or software. For example, PS-QPSK was realized with a special transmitter in [3], while combining, say, a general PM-16-QAM modulator with the capability of generating also other modulation formats leads to a more flexible transmitter. On the receiver side, however, there is no need to modify the hardware as a typical double 90° hybrid followed by balanced detectors can be used for all modulation formats. The difference instead lies in the DSP and in the following, we compare dynamic equalization, phase estimation, and data decoding for the selected modulation formats.

3.1. Polarization Demultiplexing and Dynamic Equalization

Polarization demultiplexing/dynamic equalization is for PM-QPSK typically carried out using the CMA [10], which also works for PM-16-QAM although the performance, in terms of convergence/tracking properties, is reduced. Therefore, CMA is then often only used initially to enable a switch to decision-directed mode. We have used this approach for 128-SP-QAM. However, the conventional CMA does not work for PS-QPSK and the reason is that there is no unique demultiplexing filter that achieves the constant modulus property. Instead, the PS-CMA [11] or the delay scheme described in [13] can be used. For PM-2-PPM-QPSK, we used a variation of the algorithm described in [12]. Thus, for each symbol and polarization a decision is made if a symbol is present. Then, as opposed to PS-QPSK, there

may be a 2D symbol in both, none, or one of the polarizations of each 4D symbol. The so obtained algorithm works but is sensitive to the initial configuration [12]. The reason for this is that the detection of power can be viewed as a partial decision, and the algorithm will not converge if the frequency of incorrect decisions becomes large.

3.2. Phase Estimation

Phase estimation is for PM-QPSK typically carried out using the Viterbi and Viterbi (V&V) algorithm [15] and this can be done separately for the two polarizations. The same algorithm can be used for PM-16-QAM, but the performance is degraded significantly since only half of the symbols have a data phase that is removed by the raising to the fourth power. The impact of the remaining symbols then is reduced by averaging or by not including these symbols by a partial decision. While there are a number of well-performing methods it is also important to avoid feedback for implementation purposes. We have used the method based on test angles described in [16] for our PM-16-QAM and 128-SP-QAM experiments. Cycle slips are handled by the detection as described below. For PM-2-PPM-QPSK, on average one out of two symbols carries power, which means that again only half of the symbols contribute to the phase estimation. We can here choose to either average over all symbols or make a partial decision on each symbol and find the phase based on only the power-carrying symbols. With these adaptations, the V&V algorithm can be used. The same can be said for PS-QPSK, but in this case there is the additional complication that the phases in the x - and y -polarizations have to be locked to each other. When a cycle slip occurs in one of the polarizations, all following phase transitions between QPSK symbols transmitted in different polarization states otherwise become incorrect. With the differential data encoding described above it is therefore necessary to perform a joint phase estimation of the polarization-multiplexed signals. To make this possible the relative phase between these signals must be identified and this was achieved by applying the Viterbi-Viterbi algorithm on a block of data from each tributary, see also [13]. We remark that such joint estimation is possible to perform also for the other modulation formats considered here. This reduces the variance of the estimated phase, but comes at a cost of extra complexity.

3.3. Data Detection

Data detection for PM-QPSK, PM-16-QAM, and 128-SP-QAM is performed using the algorithm described by Conway and Sloane [17]. Thus, as PM-QPSK and PM-16-QAM is a subset of a 4D integer lattice, Z_4 , the detection is a simple rounding operation followed by the symbol-to-bit mapping. Performing set partitioning transforms a Z_4 to a D_4 lattice and vice versa. Thus 128-SP-QAM is decoded using the algorithm for D_4 in [17]. In short it works as follows: Find two candidate symbols, the first as for Z_4 and the second by inverting the most uncertain bit. This is done by selecting the component that has the largest rounding error and then applying a modified rounding operation that “rounds the wrong way”. By inspecting the bit-to-symbol mapping described above, it is found that these two candidate symbols have different parity and we select the one that has even parity. This works both with absolute and differential encoding, and in the latter case cycle slips will be acceptable. For PS-QPSK, one bit is recovered by deciding which was the selected launch polarization [3]. By investigating the Voronoi regions, it is found that this is optimally done by comparing the sum of the absolute values of the I- and Q-channels for the two polarizations, respectively [3]. The remaining two bits are recovered from the differential phase between consecutive QPSK symbols. The situation for PM-2-PPM-QPSK is similar, with the roles of the two polarization states replaced by neighboring time slots.

4. Conclusion

In conclusion, we have compared the detection of a number of different power efficient formats. Starting from the typical DSP for PM-QPSK and PM-16-QAM, only small modifications are necessary. We have also described how differential detection can be implemented for all formats included in the comparison.

References

- [1] H. Bülow, OFC 2009, p. OWG2.
- [2] E. Agrell and M. Karlsson, *J. Lightw. Technol.* **27**, 5115–5126 (2009).
- [3] M. Sjödin *et al.*, *Opt. Express* **19**, 7839–7846 (2011).
- [4] T. A. Eriksson *et al.*, OFC 2013, p. OTu3B.2.
- [5] M. Sjödin *et al.*, OFC 2013, p. OTu2B.7.
- [6] G. Ungerboeck, *IEEE Trans. Inform. Theory* **IT-28**, 55–67 (1982).
- [7] M. Karlsson and E. Agrell, ECOC 2011, p. Tu.6.B.6.
- [8] J. Renaudier *et al.*, OFC 2013, p. OTu3B.1.
- [9] M. Sjödin *et al.*, *Opt. Express* **20**, 8356–8366 (2012).
- [10] D. N. Godard, *IEEE Trans. Commun.* **28**, 1867–1875 (1980).
- [11] P. Johannisson *et al.*, *Opt. Express* **19**, 7734–7741 (2011).
- [12] D. S. Millar and S. J. Savory, *Opt. Express* **19**, 8533–8538 (2011).
- [13] J. Renaudier *et al.*, *J. Lightw. Technol.* **30**, 1312–1318 (2012).
- [14] M. Sjödin *et al.*, *Opt. Express* **20**, 7544–7554 (2012).
- [15] A. J. Viterbi and A. M. Viterbi, *IEEE Trans. Inform. Theory* **IT-29**, 543–551 (1983).
- [16] T. Pfau *et al.*, *J. Lightw. Technol.* **27**, 989–999 (2009).
- [17] J. H. Conway and N. J. A. Sloane, *IEEE Trans. Inform. Theory* **IT-28**, 227–232 (1982).