



## **ASIC Design Exploration of Phase Recovery Algorithms for M-QAM Fiber-Optic Systems**

Downloaded from: <https://research.chalmers.se>, 2025-05-17 13:04 UTC

Citation for the original published paper (version of record):

Börjeson, E., Fougstedt, C., Larsson-Edefors, P. (2019). ASIC Design Exploration of Phase Recovery Algorithms for M-QAM Fiber-Optic Systems. Optics InfoBase Conference Papers.  
<http://dx.doi.org/10.1364/OFC.2019.W3H.7>

N.B. When citing this work, cite the original published paper.

# ASIC Design Exploration of Phase Recovery Algorithms for M-QAM Fiber-Optic Systems

Erik Börjesson, Christoffer Fougstedt and Per Larsson-Edefors

*Dept. of Computer Science and Engineering,  
Chalmers University of Technology, SE-412 96 Göteborg, Sweden  
erikbor@chalmers.se*

**Abstract:** We develop circuit implementations and explore design optimizations for one blind and one pilot-based carrier phase-recovery algorithm, where the former algorithm is shown to dissipate 1.8–4.5 pJ/bit and the latter 0.5–0.3 pJ/bit, using 16 to 256QAM.

**OCIS codes:** (060.0060) Fiber optics and optical communication; (060.1660) Coherent communications

## 1. Introduction

Coherent optical transmission is one of the enabling technologies for high-speed long-haul fiber-optic communications as it permits encoding of data on both the amplitude and phase of the signal and on both polarizations. Increasing the spectral efficiency, and thereby the data throughput, of a coherent system requires the use of higher-order modulation formats. Transmissions using these formats are, however, more sensitive to the impairments introduced by the fiber transmission and in a coherent fiber-optic receiver, digital signal processing (DSP) is used to compensate for these impairments. DSP functionality is typically realized as an application-specific integrated circuit (ASIC), whose power dissipation can represent a large part of the overall system power. As coherent schemes, which have been most prevalent in long-haul transmission, become more common in metro networks, there is an increasing need to reconcile increasing throughput requirements with power-efficient DSP implementations, to curb operator costs and pave the way for densely packed equipment.

One of the impairments caused by the fiber-optic system are the carrier phase noise, which is handled by the carrier phase estimation (CPE) of the DSP. For simpler formats, like quadrature phase-shift keying (QPSK), relatively straightforward CPE algorithms are available. The scalability to higher-order modulation formats is, however, limited due to the multi-level amplitudes of the constellation points of these formats. The CPE approaches suggested for formats such as 16QAM and higher can be divided into blind methods, which process the data symbols to recover the phase, and pilot-aided methods, which rely on known pilot symbols inserted in the data stream. These CPE approaches are thoroughly described in the floating-point domain, but studies on fixed-point implementation aspects, their impact on the performance of the algorithms and ASIC power dissipation, are missing.

In this work, we present ASIC implementations of two CPE algorithms (see Section 2) suitable for realization in hardware. Considering three different modulation formats, 16QAM, 64QAM, and 256QAM, we develop hardware descriptions (VHDL) of the CPE algorithms and perform algorithm-hardware co-optimizations. We consider fixed-point aspects that affect the resulting bit error rate (BER) and evaluate power dissipation and gate count for implementations in a 28-nm ASIC technology.

## 2. Phase Recovery Algorithms

The blind phase search (BPS) algorithm was introduced in [1] and suggested for use in fiber-optic systems in [2]. When applying BPS, each input symbol is rotated with  $R$  number of test phases after which the distances from the rotated symbols to the closest constellation point are calculated. A sliding-window average is used to reduce the impact of white noise and the phase with the minimum average distance is selected. The transmitted symbol is recovered by using the complex conjugate of the recovered phase to rotate the input symbol, which has been delayed  $D$  cycles. When using BPS for carrier recovery of signals employing higher-order modulation formats, such as 64QAM and 256QAM, both the number of test phases and input-resolution demands increase, which is a cause for concern and therefore explored in this work.

Using a pilot-aided carrier recovery (PAR) scheme, a set of known symbols is interleaved with the payload data symbols and used at the receiver to extract the phase information. Advantages of using PAR include increased noise tolerance and reduced risk of cycle slips, which is a problem for BPS at higher noise levels. If the pilots are to be used only for CPE, insertion of a single symbol at a rate depending on the phase noise has been suggested [3]. To reduce the impact of white noise, a short sliding-window average is used and to approximate the phase for the payload symbols between two pilots, linear interpolation is performed. The factor limiting how much phase noise PAR can handle is the CPE block length,  $C$ , i.e. one pilot symbol and the number of payload symbols up to the next pilot.

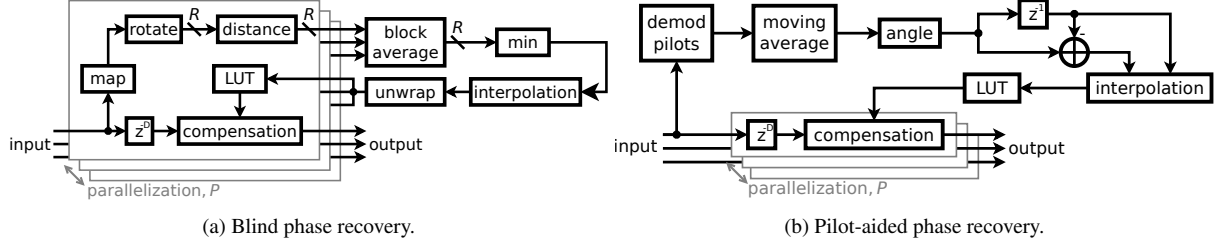


Fig. 1: Block diagrams of the two phase-recovery implementations.

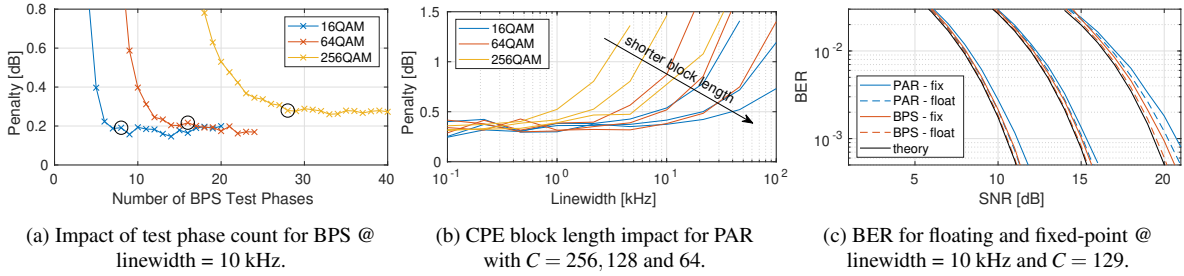


Fig. 2: Parameter and SNR dependencies for the two designs at 20 GBd.

### 3. Circuit Implementation

The two CPE algorithms were implemented for use in a DSP for 20 and 40-GBd fiber-optic systems, which we simulate as an additive white Gaussian noise (AWGN) channel with added Wiener phase noise. To fulfill the requirements on such a system, parallelization, with the factor  $P$ , pipelining and numeric approximations were used in the design. The following paragraphs describe the most important implementation aspects.

**BPS:** A block diagram of our BPS implementation is shown in Fig. 1a. At the BPS module's input, data are mapped to the first quadrant, which reduces the number of bits needed for the data representation by one. The mapping is enabled by the symmetry of the QAM symbols and can be performed using a comparator and two absolute-value operations per symbol. Not only does the mapping reduce the effective number of bits, but it also reduces the gate count of the succeeding distance block by 75%. After mapping, the symbols are rotated by  $R$  number of test phases using complex multiplication with complex constants of unit length. The symmetries of the real and imaginary part of the rotation phases are exploited to halve the number of multipliers and to further minimize the performance impact of these operations, multiplierless multiple constant multiplication (MCM) [4] is used to reduce the multiplications to a series of additions/subtractions and shifts where the intermediate results can be reused over multiple constants. The gate count of the rotate module is reduced by 25% by using MCM.

After calculating the distance from the rotated input to the closest constellation point, a sliding-window average of the distances should be computed for each test phase. Since a parallel implementation of a sliding-window average is hardware intensive, the averaging is constructed as an inner block average over the  $P$  parallel lanes, followed by an outer average used for averaging lengths exceeding the parallelization factor. The use of a block average has the added advantage of eliminating the need to parallelize the min, interpolation, and unwrap modules, with only a negligible SNR penalty.

To reduce gate count, it is imperative to keep the number of test phases to a minimum, while still achieving a good phase estimation. The relationship between the number of test phases and the penalty is shown in Fig. 2a, with the ones used for implementation circled. The number of test phases can be reduced further by the use of interpolation [5]. The hardware cost of the interpolation module is three adders and a lookup table (LUT), but the benefit is that we cut the size of the rotate, distance and average modules by 50% since we can halve the number of test phases.

**PAR:** Our pilot-aided phase-recovery implementation is shown in Fig. 1b. Here, we first demodulate the pilot symbol. This is not necessarily done every clock cycle since the block length,  $C$ , can be larger than  $P$ . For our example implementation,  $P$  is 32 and  $C$  is 128 symbols, which means that the phase estimation has four clock cycles to complete. After the demodulation, a sliding-window average is calculated and the angle of the complex result is determined using CORDIC [6]. The carrier phase error is a relatively slow-changing process, so the interpolation module is designed to update its output every  $P$ th symbol. This approach makes it possible to use one single LUT instead of one for each parallel lane, which reduces the gate count of the conversion from angles to complex vectors by 96%. The value of  $C$  determines how much phase noise the implementation can handle: For solutions using varying values of  $C$ , the impact

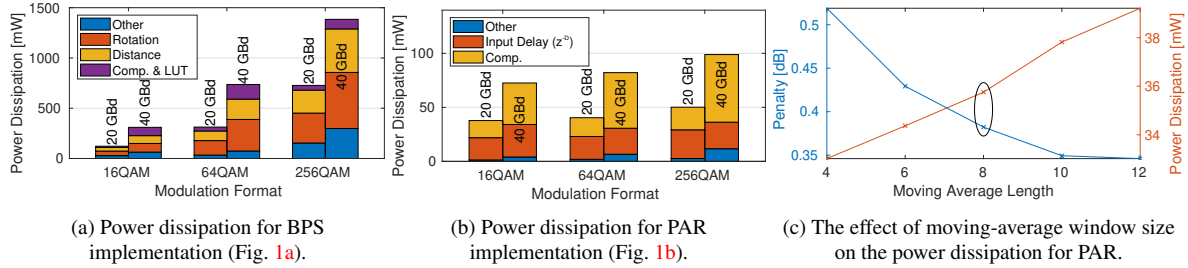


Fig. 3: Implementation results

of phase noise on the penalty is shown in Fig. 2b. A smaller  $C$  means that the delay of the input symbols,  $D$ , can be decreased, resulting in a smaller design and increased overhead.

The BER of the two designs, using either floating-point or fixed-point math, are shown in Fig. 2c. The input word length was selected based on the trade-off between word length and penalty, and is 8, 9, and 10 for 16QAM, 64QAM, and 256QAM, respectively. Compared to the theoretical best case, our PAR implementation has a larger penalty than BPS, due to the limited amount of phase information it has available.

#### 4. Power Dissipation

Synthesis of the two designs was performed using Cadence Genus and a 28-nm 0.9-V cell library. Simulation of the design was performed in Cadence Incisive, with input data generated using MATLAB, and the results back-annotated into Genus to supply switching statistics for the power dissipation analysis.

The power dissipation of the two designs is shown in Fig. 3a and 3b at two symbol rates, with similar BERs. The optimizations performed to the BPS implementation keep the increase in power dissipation with higher modulation formats to approximately a doubling with each step. For the PAR implementation the scaling to higher modulation formats has a much smaller impact on the power dissipation, as only the input word length and averaging length changes. The energy dissipated per bit, for the 20-GBd versions with 16–256QAM, is 1.8–4.5 pJ for BPS and 0.5–0.3 pJ for PAR.

When both designs are synthesized for the higher symbol rate, the parallel portion of the implementation is essentially doubled, which explains the much larger power dissipation. The input delay of the PAR implementation is, however, decreased and the effect on the delay module is therefore smaller than on the compensation module.

The two factors that have the largest impact on the power dissipation of the PAR implementation are the input word length and the length of the moving-average window. To reach the lowest possible penalty for lower-order modulation formats, a relatively long window is needed, which increases the power dissipation of the input-delay module. The trade-off between penalty and power consumption for varying averaging lengths is shown in Fig. 3c for 16QAM, with the length selected in our implementation circled.

#### 5. Conclusion

In order to demonstrate design opportunities and trade-offs, we have implemented two different algorithms for phase recovery in a coherent fiber-optic system. The implementations have been used to generate power-dissipation data which show that our optimized BPS can be a feasible option for modulation formats up to 256QAM. Unsurprisingly, the PAR implementation is more energy efficient and shows a better scaling with higher-order modulation formats, but it also has a higher SNR penalty than BPS.

*Acknowledgement: Thanks to Lars Lundberg and Mikael Mazur for interesting discussions and advice. This work was financially supported by the Knut and Alice Wallenberg Foundation and Vinnova.*

#### References

- [1] S. K. Oh and S. P. Stapleton, “Blind phase recovery using finite alphabet properties in digital communications,” *Electron. Lett.* **33**, 175–176 (1997).
- [2] T. Pfau, S. Hoffmann, and R. Noe, “Hardware-efficient coherent digital receiver concept with feedforward carrier recovery for  $M$ -QAM constellations,” *IEEE J. Lightw. Technol.* **27**, 989–999 (2009).
- [3] M. Mazur *et al.*, “Optimization of low-complexity pilot-based DSP for high spectral efficiency  $51 \times 24$  GBaud PM-64QAM transmission,” in “*Eur. Conf. Opt. Commun. (ECOC)*,” (2018), p. Mo4F.2.
- [4] Y. Voronenko and M. Püschel, “Multiplierless multiple constant multiplication,” *ACM Trans. Algorithms* **3** (2007).
- [5] H. Sun *et al.*, “Novel 16QAM carrier recovery based on blind phase search,” in “*Eur. Conf. Opt. Commun. (ECOC)*,” (2014), p. Tu.1.3.4.
- [6] J. E. Volder, “The CORDIC trigonometric computing technique,” *IRE Trans. Electron. Comput.* **EC-8**, 330–334 (1959).