

Pilot-aided carrier phase recovery for M -QAM using superscalar parallelization based PLL

Qunbi Zhuge,^{*} Mohamed Morsy-Osman, Xian Xu, Mohammad E. Mousa-Pasandi, Mathieu Chagnon, Ziad A. El-Sahn, and David V. Plant

Department of Electrical and Computer Engineering, McGill University, Montreal, QC, H3A 2A7, Canada

^{*}qunbi.zhuge@mail.mcgill.ca

Abstract: In this paper, we present a carrier phase recovery (CPR) algorithm using a modified superscalar parallelization based phase locked loop (M-SSP-PLL) combined with a maximum-likelihood (ML) phase estimation. Compared to the original SSP-PLL, M-SSP-PLL + ML reduces the required buffer size using a novel superscalar structure. In addition, by removing the differential coding/decoding and employing ML phase recovery it also improves the performance. In simulation, we show that the laser linewidth tolerance of M-SSP-PLL + ML is comparable to blind phase search (BPS) algorithm, which is known to be one of the best CPR algorithms in terms of performance for arbitrary QAM formats. In 28 Gbaud QPSK (112 Gb/s) and 16-QAM (224 Gb/s), and 7 Gbaud 64-QAM (84 Gb/s) experiments, it is also demonstrated that M-SSP-PLL + ML can increase the transmission distance by at least 12% compared to BPS for each of them. Finally, the computational complexity is discussed and a significant reduction is shown for our algorithm with respect to BPS.

©2012 Optical Society of America

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

References and links

1. P. J. Winzer, A. H. Gnauck, C. R. Doerr, M. Magarini, and L. L. Buhl, "Spectrally efficient long-haul optical networking using 112-Gb/s polarization-multiplexed 16-QAM," *J. Lightwave Technol.* **28**(4), 547–556 (2010).
2. A. H. Gnauck, P. J. Winzer, A. Konczykowska, F. Jorge, J. Dupuy, M. Riet, G. Charlet, B. Zhu, and D. W. Peckham, "Generation and transmission of 21.4-Gbaud PDM 64-QAM using a novel high-power DAC driving a single I/Q modulator," *J. Lightwave Technol.* **30**(4), 532–536 (2012).
3. M. G. Taylor, "Phase estimation methods for optical coherent detection using digital signal processing," *J. Lightwave Technol.* **27**(7), 901–914 (2009).
4. T. Pfau, S. Hoffmann, and R. Noe, "Hardware-efficient coherent digital receiver concept with feedforward carrier recovery for M-QAM constellations," *J. Lightwave Technol.* **27**(8), 989–999 (2009).
5. E. Ip and J. M. Kahn, "Feedforward carrier recovery for coherent optical communications," *J. Lightwave Technol.* **25**(9), 2675–2692 (2007).
6. I. Fatadin, D. Ives, and S. J. Savory, "Laser linewidth tolerance for 16-QAM coherent optical systems using QPSK partitioning," *IEEE Photon. Technol. Lett.* **22**(9), 631–633 (2010).
7. Y. Gao, A. P. T. Lau, S. Yan, and C. Lu, "Low-complexity and phase noise tolerant carrier phase estimation for dual-polarization 16-QAM systems," *Opt. Express* **19**(22), 21717–21729 (2011).
8. Q. Zhuge, C. Chen, and D. V. Plant, "Low computation complexity two-stage feedforward carrier recovery algorithm for M-QAM," in *Proc. OFC'11*, Paper. OMI5.
9. X. Zhou, "An improved feed-forward carrier recovery algorithm for coherent receivers with M-QAM modulation format," *IEEE Photon. Technol. Lett.* **22**(14), 1051–1053 (2010).
10. X. Zhou and Y. Sun, "Low-complexity, blind phase recovery for coherent receivers using QAM modulation," in *Proc. OFC'11*, Paper. OMI3.
11. K. Piyawanno, M. Kuschnerov, B. Spinnler, and B. Lankl, "Low complexity carrier recovery for coherent QAM using superscalar parallelization," in *Proc. ECOC'10*, Paper. We.7.A.3.
12. Q. Zhuge, M. E. Pasandi, X. Xu, B. Châtelain, Z. Pan, M. Osman, and D. V. Plant, "Linewidth-tolerant low complexity pilot-aided carrier phase recovery for M-QAM using superscalar parallelization," in *Proc. OFC'12*, Paper. OTu2G.2.
13. I. Fatadin, D. Ives, and S. J. Savory, "Blind equalization and carrier phase recovery in a 16-QAM optical coherent system," *J. Lightwave Technol.* **27**(15), 3042–3049 (2009).
14. T. Nakagawa, M. Matsui, T. Kobayashi, K. Ishihara, R. Kudo, M. Mizoguchi, and Y. Miyamoto, "Non-data-aided wide-range frequency offset estimator for QAM optical coherent receivers," in *Proc. OFC'11*, Paper. OMI1.

1. Introduction

As the quadrature phase shift keying (QPSK) modulation based 100G products are being commercialized, research interests are moving on to the next generation optical transport systems with spectrally efficient modulation formats such as 16-quadrature amplitude modulation (QAM) [1] and 64-QAM [2]. Carrier phase recovery (CPR) is an indispensable digital signal processing (DSP) procedure in coherent transmission systems to compensate for the random phase shifts induced by both the transmitter laser and the local oscillator (LO) [3–5]. A preferred CPR algorithm should satisfy three conditions: 1) high laser linewidth tolerance; 2) low computational complexity; 3) applicability to any modulation format, which is essential for future software-defined transceivers. A conventional digital phase locked loop (DPLL) meets the second and third requirements but due to the large feedback delay caused by the parallelized and pipelined processing in high speed optical transport receivers, its laser linewidth tolerance is poor [3, 4]. The Viterbi and Viterbi algorithm is well-established for QPSK systems attributed to its high laser linewidth tolerance and reasonable computational complexity [3, 5]. However, it is not suitable for higher order QAM unless modifications are made, e.g. QPSK partitioning for 16-QAM [6, 7]. The blind phase search (BPS) proposed in [4] has demonstrated one of the best known linewidth tolerances for arbitrary QAM formats. However, the complexity of this algorithm is very high. Modified BPS algorithms have been proposed to reduce the complexity, but the complexity reduction is limited [8, 9]. More recently, a PLL combined with a maximum-likelihood (ML) algorithm based on the interleaving parallelization was proposed in [10]. Compared to the BPS algorithm, it reduces complexity by a factor of more than 15 at the expense of a lower linewidth tolerance when the parallelization degree is large. Superscalar parallelization is an alternative structure for the implementation of a PLL, which requires extra buffer units and pilot symbols but achieves very good performance [11].

In this work, we describe in more detail the modified superscalar parallelization based PLL (SSP-PLL) proposed in our previous work [12]. In addition, more experimental demonstrations and investigations of the proposed algorithm are conducted and reported. Compared to the original SSP-PLL algorithm in [11], three modifications are made in our algorithm: 1) a novel superscalar buffer structure is employed and only 200 buffer symbols (per parallelization) for QPSK and 400 buffer symbols for 16-QAM and 64-QAM are needed for 1% pilot symbol overhead, whereas > 1024 symbols buffer were used in [11]; 2) differential coding/decoding is discarded since it is unnecessary in SSP-PLL with small buffers, and by removing it the performance can be improved; 3) A ML algorithm is used after the SSP-PLL to compensate for the performance loss caused by the feedback delay in pipelined processing. We first numerically demonstrate that the proposed algorithm achieves a comparable linewidth tolerance to the BPS algorithm for QPSK, 16-QAM and 64-QAM, respectively. Then the performance is verified in 28 Gbaud QPSK, 28 Gbaud 16-QAM, and 7 Gbaud 64-QAM transmission experiments. Finally, the complexity of our algorithm is discussed, showing a significant reduction compared to BPS algorithm.

2. Superscalar parallelization based carrier phase recovery

Figure 1(a) shows the block diagram of a first-order PLL. The input symbol is decided, conjugated and multiplied by the original symbol to get the phase difference. $\text{Im}(\cdot)$ denotes the imaginary part of a complex value, and it approximates the phase of this value when the phase is small. Then the decision-directed phase error e_k multiplied by a weighting parameter g is fed into a loop filter for phase tracking. Finally, the output phase ϕ_k is applied to recover the phase of the input symbol r_k , producing the output symbol r'_k .

In high speed optical transmission systems, parallelized processing is typically needed to reduce the required clock speed [4]. Normally, the serial input symbols are interleaved to P

channels each being processed through an individual PLL at a lower clock speed as shown in Fig. 1(b). It can be seen that in this case the distance between two adjacent symbols in each channel is increased to P symbols. In addition, when considering the processing delay D due to the pipelined implementation of PLL, the total feedback delay in this case is $P \times D$. In other words, the laser linewidth tolerance of such an implementation is reduced by a factor of $P \times D$, compared to the ideal PLL with only one symbol delay.

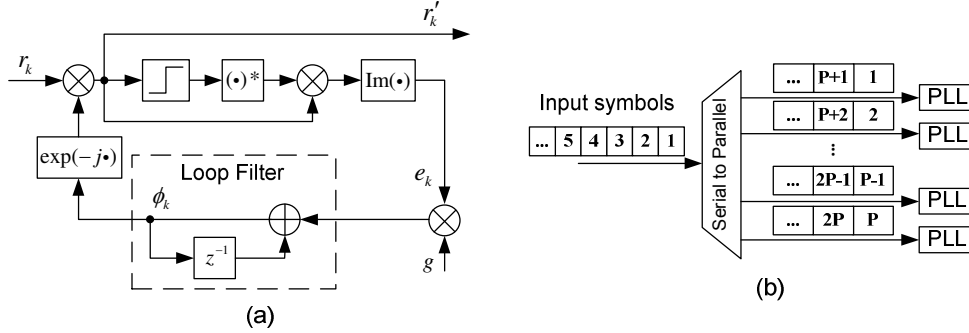


Fig. 1. (a) The block diagram of a first-order PLL. (b) The interleaving implementation of PLL in parallelized processing.

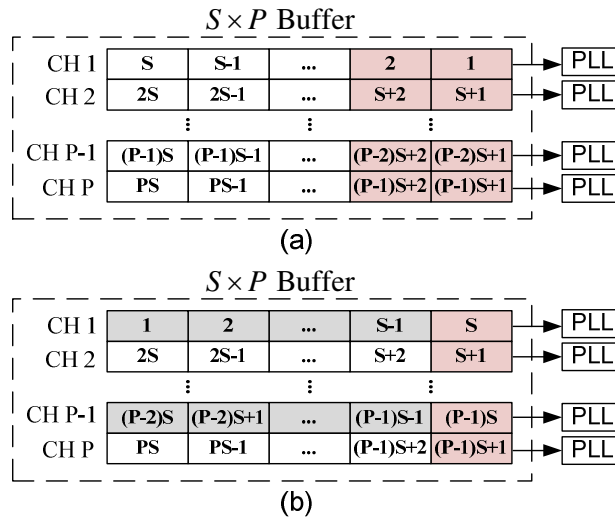


Fig. 2. (a) The original superscalar buffer structure in [11]. (b) The proposed superscalar buffer structure. CH: channel.

A PLL implemented with the superscalar parallelization (denoted as SSP-PLL) was proposed in [4] to remove the delay P caused by the interleaving parallelization and thus improve the performance. Particularly, it employs a buffer with a size of $S \times P$ symbols to store the input symbols, where S is the length of each block. The stored symbols are then rearranged to have consecutive symbols in each parallelized channel as shown in Fig. 2(a). After processing, another buffer with the same size is required to arrange back the order of symbols. Clearly, with this technique the feedback delay of PLL is reduced from $P \times D$ to only D symbols. However, two large size buffers are needed. In addition, since the PLL processing is independent between blocks in each channel, pilot symbols at the beginning of each block as highlighted in Fig. 2(a) are required for the initialization of PLL, resulting in a

pilot symbol overhead. Furthermore, the residual D symbols delay will still induce some performance loss.

We proposed a modified SSP-PLL in [12], which reduces the required buffer size and improves the laser linewidth tolerance. First, we found that the performance of SSP-PLL is only slightly affected by the block length S , which will be shown in the experimental results presented below. Therefore, the required buffer size is mainly related to the pilot symbol overhead as a tradeoff, which can be expressed as

$$\text{Overhead} = \frac{N_{ps}}{S} \quad (1)$$

where N_{ps} is the number of pilot symbols in each block. Note that for a given parallel degree P , the buffer size is proportional to S . In order to reduce the buffer size without increasing the overhead, we propose a novel superscalar structure as shown in Fig. 2(b). In particular, the order of symbols in the odd channels is inverted to have consecutive symbols at the beginning for each two channels. For example, channel 1 now starts with symbol S which is next to symbol $S + 1$ at the beginning of channel 2. By doing so, each adjacent odd and even channel can share pilot symbols because they have similar phase noise. Consequently, N_{ps} is halved and according to Eq. (1) we can either halve the buffer size for the same overhead or vice versa.

In the original algorithm, differential coding was employed [4]. However, it was noted that since the pilot symbols will initialize the PLL and removes the phase ambiguity for each block, differential coding is unnecessary. Moreover, the occurrence of cycle slip will affect only one block and thus won't cause a catastrophic failure to the system. Therefore, our algorithm removes differential coding and an appreciable performance improvement can be achieved.

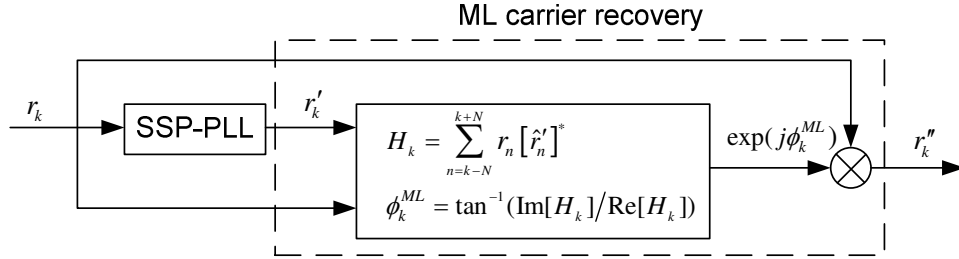


Fig. 3. The combination of the modified SSP-PLL algorithm with a ML carrier recovery.

As mentioned earlier, there is still a residual feedback delay in SSP-PLL caused by the pipelined PLL processing, which degrades performance. We propose to use a ML algorithm after the SSP-PLL algorithm to compensate for such degradation as shown in Fig. 3. Particularly, the symbol \hat{r}'_k decided based on the symbol r'_k , of which the phase is recovered by the SSP-PLL algorithm, is fed into the ML module described as below

$$H_k = \sum_{n=k-N}^{k+N} r_n [\hat{r}'_n]^* \quad (2)$$

$$\phi_k^{ML} = \tan^{-1}(\text{Im}[H_k]/\text{Re}[H_k]) \quad (3)$$

where H_k is the complex value which contains the phase difference ϕ_k^{ML} between the input symbol r_k and the decided symbol \hat{r}'_k . A filter with a length of $L = 2N + 1$ is employed in Eq. (2) to remove the phase interference from amplified spontaneous emission (ASE) noise,

fiber nonlinearities and possibly incorrect decisions. Finally, the recovered symbol r_k^n is obtained by applying the estimated phase ϕ_k^{ML} to the input symbol.

3. Performance

3.1 Simulations

Simulations were conducted to investigate the laser linewidth tolerance of the modified SSP-PLL algorithm with the ML algorithm, which is denoted as M-SSP-PLL + ML. Other algorithms including BPS in [4], interleaving parallelization based PLL + ML in [10] (denoted as ILP-PLL + ML) and the original SSP-PLL in [11] (denoted as O-SSP-PLL) were also simulated for comparison. For our algorithm, each block contained 100 symbols including 1 pilot symbol for QPSK, and 200 symbols including 2 pilot symbols for 16-QAM and 64-QAM, resulting in a 1% overhead for all systems. For O-SSP-PLL, the block length S was 512 symbols. The parallelization degree P was 16 and the PLL processing delay D was 4 symbols, leading to a feedback delay of 4 ($=D$) symbols and 64 ($=P \times D$) symbols for SSP-PLL and ILP-PLL, respectively. For BPS, the number of test phases B was 32 for QPSK and 16-QAM, and 64 for 64-QAM [4]. Differential coding was employed for O-SSP-PLL, ILP-PLL + ML and BPS to remove phase ambiguity and avoid cycle slip induced failure. The optimal parameters for the algorithms such as the error weighting parameter g for PLL and filter length L for ML and BPS were used to obtain the following results.

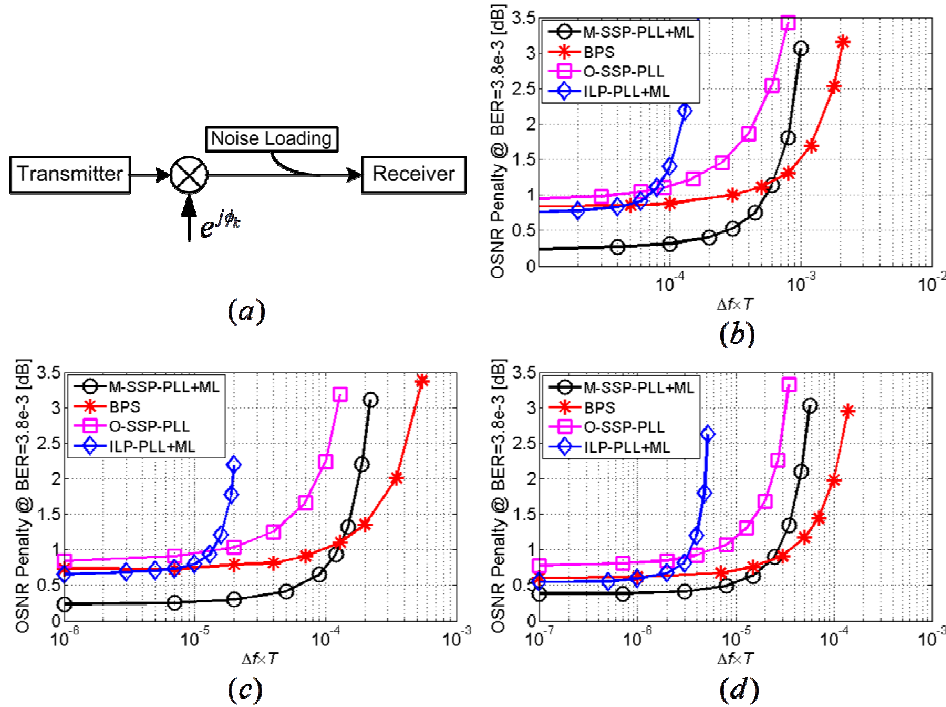


Fig. 4. (a) The simulation setup. Simulated linewidth tolerance of various algorithms for (b) QPSK, (c) 16-QAM and (d) 64-QAM.

Figure 4(a) depicts the simulation setup. Laser phase noise ϕ_k was modeled as a Wiener process with a variance of $2\pi\Delta fT$, where Δf denotes the combined linewidth of the transmitter laser and LO, and T denotes the symbol duration. ASE noise was loaded to vary the optical signal-to-noise ratio (OSNR). Figures 4(b), 4(c), and 4(d) show the OSNR penalty as a function of the linewidth and symbol duration product $\Delta f \times T$ for QPSK, 16-QAM and 64-QAM, respectively. The theoretical limit is used as a reference. The laser linewidth tolerance,

which is typically defined as the linewidth causing 1 dB OSNR penalty [4], is summarized in Table 1 for different algorithms. As can be seen, M-SSP-PLL + ML provides more than 4.5 times larger linewidth tolerance than both O-SSP-PLL and ILP-PLL + ML, while it achieves similar tolerance as BPS, which has been reported to have one of the best known performances for arbitrary QAM formats [4]. Moreover, M-SSP-PLL + ML reduces the required OSNR by 0.6, 0.5 and 0.2 dB with respect to BPS for QPSK, 16-QAM and 64-QAM, respectively, for small linewidths. This is achieved by removing differential coding in our algorithm at the expense of 1% overhead. However, with a large $\Delta f \times T$ ($>6 \times 10^{-4}$ for QPSK, $>1.5 \times 10^{-4}$ for 16-QAM and $>3 \times 10^{-5}$ for 64-QAM) the performance of M-SSP-PLL + ML degrades more rapidly than BPS as $\Delta f \times T$ increases due to the decision-directed processing for both the PLL and ML algorithm, in which case the error will propagate once it occurs.

Table 1. Summary of Laser Linewidth Tolerance.

	BPS	ILP-PLL + ML	O-SSP-PLL	M-SSP-PLL + ML
QPSK	3×10^{-4}	6.9×10^{-5}	3.5×10^{-5}	5.4×10^{-4}
16-QAM	9.3×10^{-5}	1.4×10^{-5}	1.5×10^{-5}	1.3×10^{-4}
64-QAM	3.6×10^{-5}	3.5×10^{-6}	5.8×10^{-6}	2.7×10^{-5}

3.2 Experiments

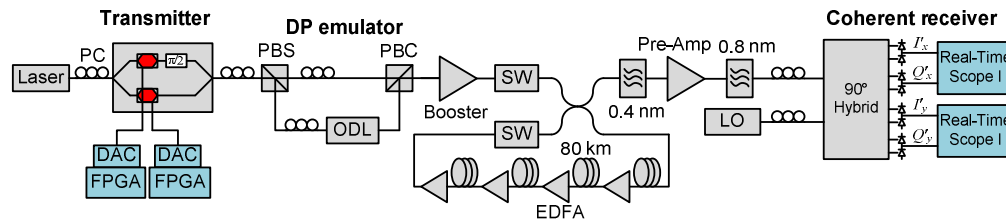


Fig. 5. Experimental setup. ODL: optical delay line, PBS/PBC: polarization beam splitter/combiner, PC: polarization controller, SW: switch.

Next, we conducted experiments to investigate the performance of our algorithm in dual-polarization (DP) 28 Gbaud QPSK (112 Gb/s) and 16-QAM (224 Gb/s) systems, and 7 Gbaud 64-QAM (84 Gb/s) system, respectively. Figure 5 shows the schematic of the experimental setup. Two digital-to-analog converters (DACs) each driven by a field-programmable gate array (FPGA) boards were used to generate non-return-to-zero (NRZ) two-level, four-level and eight-level electrical inphase and quadrature signals for QPSK, 16-QAM and 64-QAM, respectively. The IQ-modulation was employed for electrical-to-optical conversion. A DP signal was generated by splitting the signal to two orthogonal polarizations, delaying one polarization and recombining them. Then the DP signal was boosted to the optimal launch power before being fed into an optical re-circulating loop, which consisted of 4 spans of 80 km standard single mode fiber (SMF-28e +) each having an erbium doped fiber amplifier (EDFA) with 5 dB noise figure to compensate the loss. The signal out of the loop was filtered, pre-amplified and re-filtered. Then along with the LO light, the signal was fed into a 90° hybrid followed by 4 balanced photodiodes for coherent detection. The generated electrical signals were digitized by two real-time scopes operating at 80 GS/s. In the offline processing IQ imbalance was first compensated, and then CD compensation was implemented in the frequency domain. After re-sampling to two samples per symbol, an adaptive time domain filter was implemented to de-multiplex polarization and compensate for residual inter-symbol interference. Constant modulus algorithm (CMA) and radius directed equalization (RDE) was employed for the adaptation of the time domain filter for QPSK and 16-QAM [13], respectively. We also use CMA for 64-QAM because it is independent of the CPR block, but

it should be noted that by switching to least mean square (LMS) algorithm, which has to be integrated with CPR in the implementation, the performance can be improved. The integration of LMS and M-SSP-PLL + ML is left for future work. Next, frequency offset was estimated based on the periodogram of the 4th power of the symbols [14]. Finally, CPR was implemented and a decision was made on the symbols at the CPR block output. Again, along with the proposed algorithm M-SSP-PLL + ML, the performances of the other three algorithms including BPS, O-SSP-PLL and ILP-PLL + ML were also evaluated for comparison.

Different lasers including ECLs and DFB lasers were used to test the algorithms. The linewidth of the ECLs we used was ~ 100 kHz, while the linewidth of the DFB lasers was estimated from the phase noise variance measured using the coherent detection technique in [15] with the Wiener process assumption. For QPSK, the laser with a linewidth of 2.6 MHz and 0.6 MHz was used at the transmitter and receiver, respectively. For 16-QAM, the ECL was used at the transmitter, while three different lasers including the ECL, and 1.3 MHz and 2.6 MHz linewidth DFB lasers were employed for the performance evaluation. For 64-QAM, two ECLs were used.

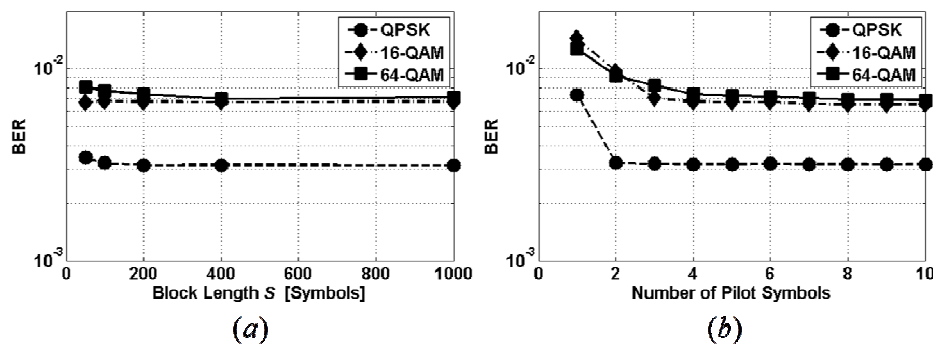


Fig. 6. (a) BER vs. block length with 2 pilot symbols for QPSK and 4 pilot symbols for both 16-QAM and 64-QAM. (b) BER vs. number of pilot symbols with $S = 100$ for QPSK and $S = 200$ for both 16-QAM and 64-QAM.

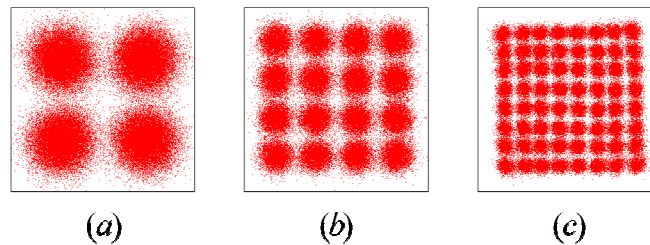


Fig. 7. Constellations of (a) QPSK (4800 km), (b) 16-QAM (640 km) and (c) 64-QAM (320 km).

First, we investigate the performance versus the block length S and the number of pilot symbols N_{PS} for M-SSP-PLL + ML for all the three modulation formats, which is shown in Figs. 6(a) and 6(b), respectively. The BER is obtained at the distance of 4800 km, 640 km and 320 km without extra noise loading for QPSK, 16-QAM (2.6 MHz laser as the LO) and 64-QAM, respectively. As per Fig. 6(a), the BER is almost constant except a very slight increase for smaller block lengths for QPSK and 64-QAM. Since S determines the required buffer size, it should be implemented as small as possible in order to save complexity. However, a smaller S results in a larger pilot symbol overhead. But as shown in Fig. 6(b), 2 pilot symbols are adequate for QPSK, while 16-QAM and 64-QAM require no more than 4 pilot symbols. Since

the pilot symbols are shared between two adjacent channels with our superscalar structure as shown in Section 2, the overhead is given by

$$Overhead = \frac{N_{ps}/2}{S} \quad (4)$$

Therefore, to limit the overhead to 1%, we set S to 100 symbols for QPSK and 200 symbols for both 16-QAM and 64-QAM. The corresponding constellations with those parameters are shown in Figs. 7(a), 7(b) and 7(c), for QPSK, 16-QAM and 64-QAM, respectively.

Next, we investigate the performance for each modulation format. Figures 8(a) and 8(b) show BER versus OSNR (at back-to-back transmission) and transmission distance for QPSK, respectively. In Fig. 8(a), M-SSP-PLL + ML reduces the required OSNR at a BER = 3.8×10^{-3} by 0.7 dB and 0.9 dB compared to BPS and O-SSP-PLL, respectively. The performance of ILP-PLL + ML depends on the parallelization degree P . With $P = 8$, it can achieve a performance as good as BPS. However, when P is increased to 16, a 1.8 dB OSNR penalty is observed with respect to M-SSP-PLL + ML. For the transmission performance in Fig. 8(b), M-SSP-PLL + ML reaches 4800 km with a BER below 3.8×10^{-3} , and increases the distance by 12.4%, 17.6%, 13.5% and 47.2% compared to BPS, O-SSP-PLL, ILP-PLL + ML with $P = 8$ and $P = 16$, respectively.

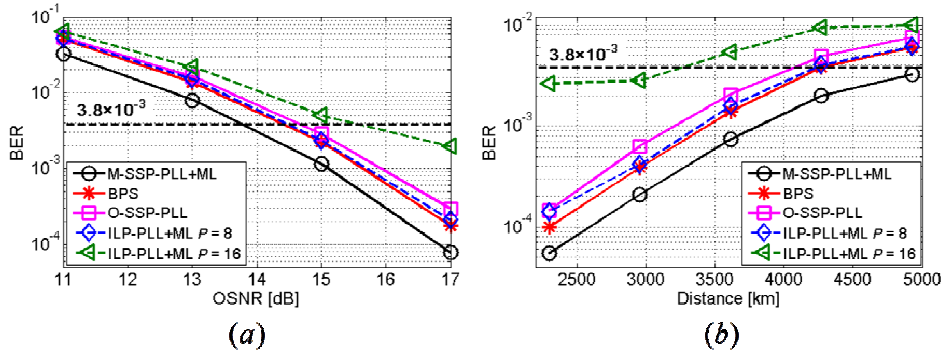


Fig. 8. (a) BER vs. OSNR at back-to-back transmission and (b) BER vs. distance for QPSK with one 2.6 MHz and one 0.6 MHz linewidth DFB laser at the two ends.

For 16-QAM, the experimental results for three different combinations of lasers are shown in Figs. 9, 10, 11, respectively. For the two ECLs scenario where the variance of laser phase noise is very small, all other algorithms perform similarly except M-SSP-PLL + ML which achieves a better performance (>1 dB and $>22\%$ in terms of OSNR in Fig. 9(a) and distance in Fig. 9(b), respectively) mainly attributed to the removal of differential coding. When the ECL is replaced by a DFB laser with a linewidth of 1.3 MHz at the receiver, the performances of O-SSP-PLL and ILP-PLL + ML become worse than BPS due to their lower tolerance to laser linewidth, especially for ILP-PLL + ML with $P = 8$ which cannot reach the BER threshold at back-to-back transmission without noise loading as shown in Fig. 10(b). However, M-SSP-PLL + ML is still able to improve the OSNR by approximately 1 dB and the distance by 25% with respect to BPS. As the linewidth of the laser further increases, a larger performance degradation of O-SSP-PLL and ILP-PLL + ML compared to BPS and M-SSP-PLL + ML is observed as plotted in Fig. 11 where an ECL and a DFB laser with a 2.6 MHz linewidth are employed. We also see that the performance difference between M-SSP-PLL + ML and BPS becomes small. This is consistent with the simulation result in Fig. 4(c) which shows that the performance of M-SSP-PLL + ML degrades faster than that of BPS for a very large linewidth. Nevertheless, M-SSP-PLL + ML still achieves an OSNR improvement of approximately 0.5 dB compared to BPS as shown in Fig. 11(a). More interestingly, it still increases the transmission distance by 25% as shown in Fig. 11(b). It is noted that the performance with one ECL and one DFB laser is worse than that with two ECLs when

comparing Fig. 9 with Figs. 10 and 11. This shouldn't be caused by the laser linewidth difference since both M-SSP-PLL + ML and BPS are expected to tolerate such a high linewidth without inducing a large penalty as shown in the simulation. Rather, this might be caused by other differences of our lasers. For example, the power of the DFB lasers is at least 2 dB less than the ECLs used in the experiments, which reduces the LO to signal power ratio (LSR).

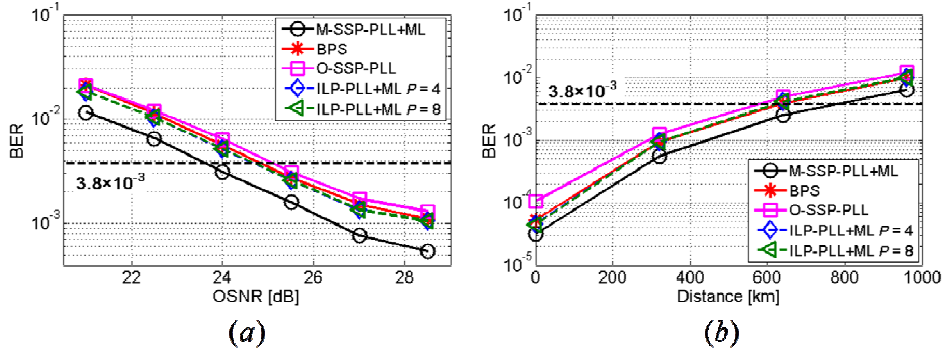


Fig. 9. (a) BER vs. OSNR at back-to-back transmission and (b) BER vs. distance for 16-QAM with two ECLs.

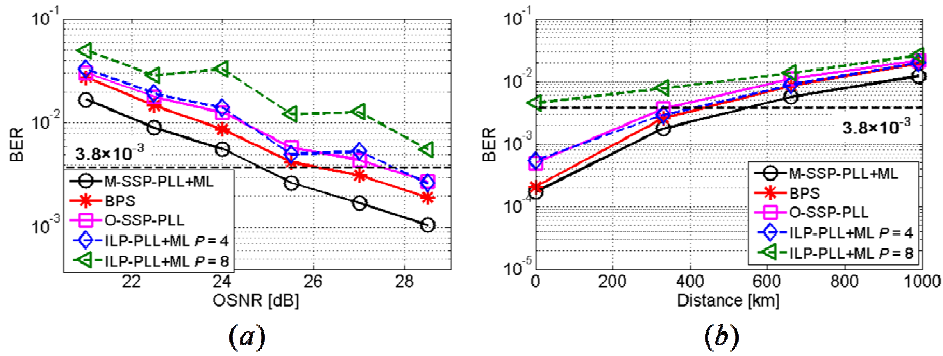


Fig. 10. (a) BER vs. OSNR at back-to-back transmission and (b) BER vs. distance for 16-QAM with one ECL and one 1.3 MHz linewidth DFB laser.

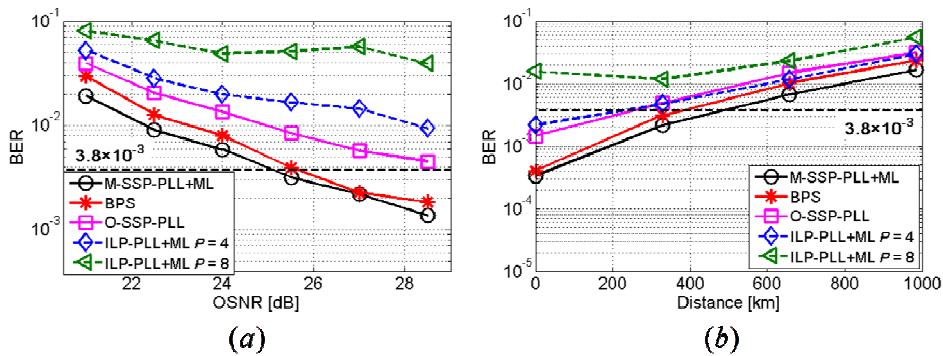


Fig. 11. (a) BER vs. OSNR at back-to-back transmission and (b) BER vs. distance for 16-QAM with one ECL and one 2.6 MHz linewidth DFB laser.

For low order QAM such as QPSK and 16-QAM, some other feedforward CPR algorithms such as the 4th power scheme provide decent performance with reasonable complexities [3, 6]. However, those algorithms are not suitable for high order QAM such as 64-QAM. In Fig. 12, we show that our algorithm with parallelized implementation also works well for 64-QAM. In particular, M-SSP-PLL + ML requires 1 dB less OSNR than other algorithms for both the 3.8×10^{-3} and 2×10^{-2} (20% forward error correction (FEC) overhead) BER thresholds at back-to-back transmission in Fig. 12(a). And it increases the transmission distance by >24% and >12% compared to other algorithms in Fig. 12(b) for 3.8×10^{-3} and 2×10^{-2} BER threshold, respectively. ILP-PLL + ML with $P = 16$ performs similarly as BPS, whereas the performance degrades when P is increased to 32. This result is consistent with that reported in the original work [10].

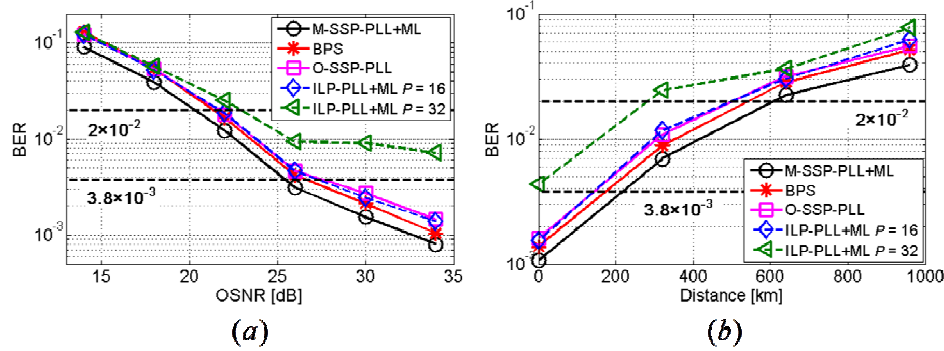


Fig. 12. (a) BER vs. OSNR at back-to-back transmission and (b) BER vs. distance for 64-QAM with two ECLs.

4. Complexity

Computational complexity of a DSP algorithm is a critical factor that determines whether an algorithm can be implemented in a real-time transceiver, where the computational resources are limited at high data rates. In this section, we compare the complexity of our algorithm with other three algorithms including BPS, O-SSP-PLL and ILP-PLL + ML. The complexities in terms of the number of operations assuming the filter length L for ML algorithm and BPS, the number of test phases B for BPS and the block length S for SSP-PLL are summarized in Table 2. The number of real multiplier, real adder and slicer is for each symbol, while the number of buffer units is for each parallelization channel. Note that each complexity multiplier is calculated as 4 real multipliers and 2 real adders. We also calculate the number of operations in Table 1 with $B = 32$ (left) and 64 (right) for BPS, $L = 21$ for all algorithms, $S_1 = 100$ (left) and 200 (right) for M-SSP-PLL + ML, and $S_2 > 512$ for O-SSP-PLL.

Table 2. Complexity comparison between CPR algorithms

	Real Multiplier	Real Adder	Slicer	Buffer Units
M-SSP-PLL + ML	15	$2L + 6$ (48)	2	$2S_1 + L$ (221/421)
BPS	$6B$ (192/384)	$(L + 6)B$ (864/1728)	B (32/64)	$L \cdot B$ (672/1344)
O-SSP-PLL	7	4	1	$2S_2$ (>1024)
ILP-SSP-PLL	15	$2L + 6$ (48)	2	L (21)

The PLL-based algorithms have similarly reasonable complexities. Compared to ILP-SSP-PLL, the SSP-PLL algorithms require extra buffer units. A block length S_2 larger than 512 symbols is suggested in the original work for O-SSP-PLL [11]. In this work, block lengths S_1 of only 100 symbols for QPSK and 200 symbols for 16QAM and 64-QAM are used with the novel superscalar structure as introduced in Section 2. Therefore, the needed buffer units are

significantly reduced compared to O-SSP-PLL. The second stage ML algorithm adds some complexities but improves the performance as demonstrated in the previous section.

M-SSP-PLL + ML not only achieves better performance than BPS as demonstrated in the previous section but also consumes much less computational resources. In particular, with the typical parameters in Table 2 it can reduce the number of real multiplier, real adder and slicer by a factor of 12~25, 18~36 and 16~32, respectively. More interestingly, the required buffer units are also reduced by a factor of 1.6~6.

5. Conclusions

In this paper, we report a modified superscalar parallelization based phase locked loop (M-SSP-PLL) followed by a maximum-likelihood (ML) carrier recovery for arbitrary QAM modulation formats. In simulation, we show that the proposed algorithm achieves a very high laser linewidth tolerance, which is similar to the blind phase search (BPS) algorithm. Its performance is also verified in 28 Gbaud QPSK (112 Gb/s) and 16-QAM (224 Gb/s), and 7 Gbaud 64-QAM (84 Gb/s) experiments, where an increase of >12% in transmission distance with respect to BPS and other PLL based algorithms is demonstrated for all of the transmission experiments conducted. Furthermore, compared to BPS, the proposed algorithm achieves a significant reduction in computation complexity.

Acknowledgments

The VEGA DACs were supplied by Micram.