Proceedings of the 4th International Conference on Mechatronics and Smart Systems
DOI: 10.54254/2755-2721/2026.34185

# *Noise Threshold Effects in High-Order PAM Carrier Synchronization: A Three-Dimensional Performance Evaluation of SD Algorithm and Costas Loop*

**Boxuan Wang**

*School of Electronic Information, Wuhan University, Wuhan, China*
*3451264641@qq.com*

***Abstract.*** This study conducts a systematic comparative analysis of the Squared-Difference (SD) algorithm and the Costas loop algorithm for carrier synchronization in high-order Pulse Amplitude Modulation (PAM) systems, assessing their noise resilience and computational complexity to inform algorithm selection in high-speed communication systems. We tested both algorithms using the same simulation framework for 4/6/8-PAM modulation over additive white Gaussian noise (AWGN) channels. We looked at how well they did in terms of steady-state phase error variance, speed of convergence, and extra work needed to run the program. The most important results show that the Costas loop works better than the SD algorithm when there is a lot of noise (noise power >5) and when there is a lot of modulation (8-PAM). It has better phase tracking accuracy (with a variance of only 0.022 rad²) and is less likely to be affected by noise.The Costas loop works much better than the SD algorithm, but it costs almost twice as much to run, needing 1013 operations per symbol instead of 512. The SD algorithm matches Costas loop performance in low-noise or low-order modulation but suffers rapid degradation under high noise due to decision-directed error propagation (8-PAM variance peaks at 0.15 rad²).The Costas loop gives up speed for stability, while the SD algorithm is great at converging quickly. These insights make it clear where the application boundaries are. Satellite communications and other systems that need to be very reliable work best with the Costas loop. The SD algorithm works best for short-range links that don't use a lot of resources and have good channel conditions. This study creates a quantitative, multi-dimensional framework for choosing algorithms in high-order PAM systems.

***Keywords:*** Carrier Synchronization, Pulse Amplitude Modulation, SD Algorithm, Costas Loop Algorithm, Noise Robustness

## 1. Introduction

Modern digital communication systems have an unprecedented need for faster data rates.The rapid growth of technologies like big data, cloud computing, and AI is making it possible for data to be sent at an alarming rate. This is a big problem for technologies that send data over the physical layer. In situations like satellite communications and high-speed wired/wireless connections, receivers need to be able to accurately get the original data back from weak and distorted signals [1].

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2026.34185

High-order Pulse Amplitude Modulation (PAM) is now an important technology for high-speed communication systems because it is easy to use and has a high spectral efficiency. PAM sends information by mapping binary bits to multi-level symbols. This lets it send more data at the same bandwidth. But traditional phase-locked loops (PLLs) have trouble locking directly because suppressed-carrier modulation signals like PAM don't have discrete carrier components that can be used directly. Therefore, specifically designed carrier recovery algorithms must be employed [2].

The decision-directed (Squared-Difference, SD) algorithm and the Costas loop algorithm are two common solutions. They have very different structural principles, levels of implementation complexity, and situations where they can be used. Bammidi et al. [3] proposed a design method for an error detector circuit of a BPSK Costas loop in millimeter - wave receivers. They made a special piece of hardware for the analog domain to find phase errors. This fixed the problems with the phase detector's severe non-linear distortion and low detection accuracy when the carrier was synchronized in the millimeter-wave band. This method found low-power, high-sensitivity phase error detection for communication in the millimeter wave range. But since it targets binary phase - modulated signals, it can't be directly used for the phase detection of multi - level PAM modulation. Ferramola et al. [4] proposed a lock detector design method based on Costas loop output and constructed a real - time lock - state detection module for SBPSK mobile communication systems. It fixed the problems with traditional Costas loops, like not having clear criteria for determining lock and being prone to false locking. It did this by accurately identifying the phase-locked state and giving an early warning of false lock. But because its detection criterion was made with binary phase modulation signal characteristics in mind and didn't take into account changes in symbol amplitude in multi-level PAM modulation, this method can't be used directly on high-order PAM carrier synchronization systems. Liu et al. [5] proposed a DSP-free coherent detection method for short-reach optical communications and built an optical-domain carrier recovery module based on an optical Costas loop. This solved the problems of high power use and long latency that traditional digital Costas loops had because they depended on DSP. It also made direct carrier phase tracking and recovery possible in the optical domain and gave us a new way to connect low-power short-range optical signals. Nevertheless, this optical architecture has extremely strict requirements for the phase matching accuracy of optical components, resulting in high engineering implementation costs and making it highly vulnerable to environmental perturbations [6]. Although numerous scholars have focused on improving the SD algorithm and the Costas algorithm, few studies have conducted a systematic comparative analysis of the two within the same framework [7]. So, this paper builds a full simulation link for PAM modulation and carrier synchronization so that the SD algorithm and the Costas loop can be compared side by side. First, the generation and pulse shaping of 4/6/8-PAM baseband signals are implemented; second, two carrier phase tracking modules—the SD algorithm and the Costas loop—are respectively built; then, under an Additive White Gaussian Noise (AWGN) channel environment, the steady-state variance of the phase estimation is systematically collected as the performance evaluation metric by varying the noise intensity (11 levels from 0 to 10) and the modulation order;finally, a full evaluation is done from three points of view: how accurate the phase tracking is, how well it can handle noise, and how hard it is to compute.

The primary innovations and contributions of this work are as follows:

• This research creates the first comprehensive simulation framework to systematically evaluate the SD algorithm and Costas loop across 4/6/8-PAM modulation under complete noise range conditions, measuring their phase-tracking precision and noise resilience while filling the research void in high-order PAM carrier synchronization.

DOI: 10.54254/2755-2721/2026.34185

• The research shows that the SD algorithm's performance drops off quickly when noise goes over a certain level (5–6), which shows a major flaw in the decision-directed error propagation mechanism. This is an important theoretical basis for improving the algorithm in the future.

• We create a complete evaluation system that takes into account complexity, accuracy, and noise resistance. This system clearly shows the practical limits of the SD algorithm (512 operations/symbol) and the Costas loop (1013 operations/symbol), which is useful for real-world implementation situations.

## 2. Related work

### 2.1. PAM signal model and carrier synchronization issue

Pulse amplitude modulation carries information by varying the amplitude of the carrier wave [8]. For Mary PAM modulation, the baseband symbol set can be expressed as:

$$a_k \in \{\pm 1, \pm 3, \ldots, \pm(M-1)\} \tag{1}$$

where M is the modulation order. After pulse shaping and carrier modulation, the transmitted signal is given by:

$$s(t) = \sum_k a_k g(t - kT_s) cos(2\pi f_c t + \varphi_0) \tag{2}$$

where g(t) is the shaping pulse, $T_s$ is the symbol period, $f_c$ is the carrier frequency, and $\phi_0$ is the initial carrier phase.

After down-conversion at the receiver, the baseband signal is obtained as:

$$r\left(t\right) = \sum_k a_k g\left(t - kT_s\right) e^{j(\varphi - \widehat{\varphi})} + n\left(t\right) \tag{3}$$

where $\phi$ is the actual carrier phase, $\widehat{\phi}$ is the locally estimated phase, and n(t) is additive white Gaussian noise. The objective of carrier synchronization is to make the phase estimate $\widehat{\phi}$ as close as possible to the true value $\phi$, so as to eliminate the impact of phase rotation on demodulation performance.

For suppressed-carrier PAM signals, there are no discrete carrier components in the received signal spectrum [9]. Therefore, the carrier cannot be directly extracted using a narrowband filter, and phase synchronization must be achieved through nonlinear transformation or special loop structures.

### 2.2. Principle of the SD loop algorithm

The SD algorithm is a classic carrier phase synchronization scheme for suppressed-carrier modulated signals. Its core idea is to eliminate modulation information through nonlinear squaring transformation, convert the suppressed-carrier signal into a signal containing a double carrier frequency component, and then extract and track the phase of this carrier component via a closed-loop phase-locked structure [10, 11].

DOI: 10.54254/2755-2721/2026.34185

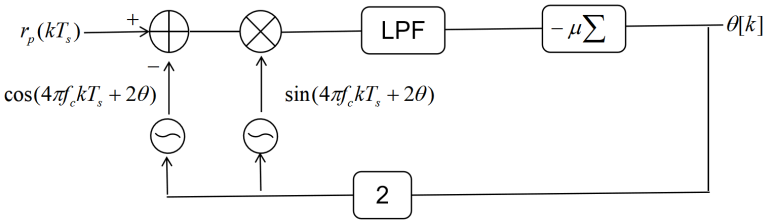



Figure 1. Block diagram of the SD algorithm

Figure 1 presents the block diagram of the Squared-Difference (SD) algorithm, illustrating the carrier synchronization process for suppressed-carrier PAM signals.

Specifically, the receiver first performs a squaring operation on the base band suppressed-carrier signal to obtain the preprocessed signal. This operation can effectively eliminate the influence of symbol modulation, such that the signal contains a prominent double carrier frequency component. Subsequently, the output phase estimate generates two quadrature double carrier reference signals and through a frequency multiplier module, a phase error is extracted by a square-law phase detector composed of a subtractor and a multiplier. The difference signal is obtained by subtracting the local cosine reference from the received signal, which is then multiplied by the local sine reference to obtain the phase detector output:

$$e_2(k) = [r_p(kT_s) - cos(4\pi f_c kT_s + 2\theta[k])] \cdot sin(4\pi f_c kT_s + 2\theta[k]) \tag{4}$$

The phase detector output is proportional to the phase deviation. After this error signal is smoothed by a low-pass filter to suppress high-frequency noise and transient disturbances, a smoothed error is obtained, which then enters an integral loop controller to achieve phase updating:

$$\theta\left[k+1\right] = \theta\left[k\right] - \mu \sum_{n=0}^{k} e_{LPF}(n) \tag{5}$$

where $\mu$ is the step-size factor, and the negative sign ensures the negative feedback mechanism. Finally, the updated phase estimate is fed back to the local reference generation module, forming a closed-loop phase-locked structure. By continuously minimizing the squared difference from and the local reference, converges to the true carrier phase, completing carrier synchronization for the suppressed-carrier signal.

## 2.3. Principle of the Costas loop algorithm

The Costas loop is specifically designed for suppressed-carrier modulated signals and can directly extract the carrier phase without prior nonlinear transformation [12]. The algorithm structure is shown in Figure 2.

The Costas loop consists of two branches: the in-phase (I) branch and the quadrature (Q) branch. The received signal is multiplied by the cosine and sine components of the local oscillator respectively [13].

Multiplier output of the (I) branch:

$$i(t) = r(t) \bullet cos(\omega_c t + \widehat{\varphi}) = \tfrac{1}{2} s(t) \left[cos(\varphi - \widehat{\varphi}) + cos(2\varpi_c t + \varphi + \widehat{\varphi})\right] \tag{6}$$

Multiplier output of the (Q) branch:

DOI: 10.54254/2755-2721/2026.34185

$$q(t) = r(t) \bullet \left[-sin(\omega_c t + \widehat{\varphi})\right] = \frac{1}{2} s(t) \left[sin(\varphi - \widehat{\varphi}) - sin(2\varpi_c t + \varphi + \widehat{\varphi})\right] \tag{7}$$

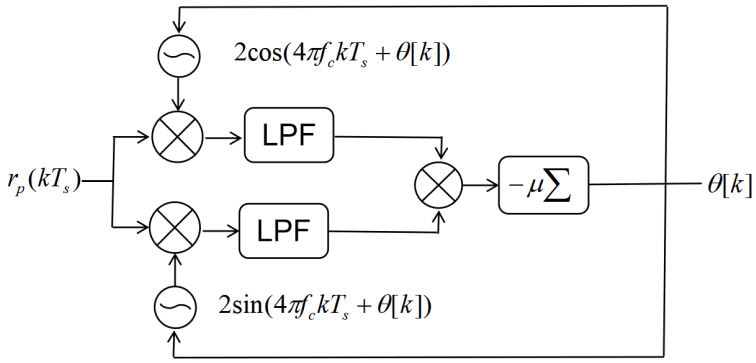



Figure 1. Block diagram of the SD algorithm

Figure 2 illustrates the Costas loop algorithm, which uses in-phase and quadrature branches with mixing, low-pass filtering, and product phase detection to directly extract and track the carrier phase without prior nonlinear transformation.

After low-pass filtering, the high-frequency components are suppressed, multiplying the two signals yields the phase error signal:

$$e(t) = I(t) \bullet Q(t) = \frac{1}{8} s^2(t) sin(2(\varphi - \widehat{\varphi})) \tag{8}$$

For PAM signals, $s^2(t)$ is positive, and the error signal is proportional to the sine of the phase difference. When the phase error is small, the error signal is approximately linear with the phase difference:

$$e(t) \approx \frac{1}{4} s^2(t)(\varphi - \widehat{\varphi}) \tag{9}$$

The error signal is smoothed by a loop filter and then drives the VCO to adjust the phase of the local oscillator, forming a negative feedback closed loop. The advantages of the Costas loop are that it avoids squaring loss and requires no preprocessing filtering.

## 3. Algorithm implementation and simulation

### 3.1. Overall process

In this study, a software-defined radio simulation platform is built based on MATLAB R2023b to systematically compare and analyze the performance of the SD algorithm and the Costas loop under 4/6/8-PAM modulation. The simulation platform consists of a transmitter, an additive white Gaussian noise channel, and receiver algorithm modules. All modules are implemented in discrete time to ensure the controllability and repeatability of numerical simulations.

First, a random binary bit stream is generated and mapped to PAM symbols. Second, pulse shaping and carrier modulation are performed to generate the transmitted signal. Then, Gaussian white noise is added to the modulated signal to simulate channel transmission. Next, the SD algorithm and the Costas loop are adopted at the receiver to track and estimate the carrier phase,

DOI: 10.54254/2755-2721/2026.34185

respectively. Finally, the steady-state variance of the phase estimation is calculated and the statistical results are recorded.

To ensure the fairness of the comparison, uniform simulation parameters are adopted for both algorithms: carrier frequency of 1000 Hz, sampling rate of 20000 Hz, symbol rate of 100 symbols/s, and oversampling factor of 20. The initial phase offset is uniformly set to -0.8rad. For each modulation order (4/6/8) and each noise intensity, 500 independent Monte Carlo experiments are conducted for both algorithms. In each experiment, the random symbol sequence and additive white Gaussian noise are regenerated. The variance between the phase estimates of the last 100 sampling points and the true phase is taken as the single performance metric. Ultimately, the average variance of 200 experiments is used as the experimental result to eliminate the influence of randomness.

### 3.2. PAM signal generation module

To evaluate the performance of three carrier synchronization algorithms under different modulation orders, pulse amplitude modulation (PAM) is adopted as the baseband signal format in this paper. PAM achieves information transmission by mapping binary bits to multi-level symbols with symmetrically distributed symbol sets. For M-PAM, symbol values are ±1, ±3, …, ±(M−1), where M denotes the modulation order. Three modulation schemes, namely 4-PAM, 6-PAM, and 8-PAM, are used in the simulation, with corresponding symbol sets {±1, ±3}, {±1, ±3, ±5}, and {±1, ±3, ±5, ±7}, respectively. To limit the signal bandwidth and reduce inter-symbol interference, a raised cosine roll-off filter is employed for pulse shaping, with a roll-off factor of 0.5 and a filter order of 20. The shaped baseband signal is multiplied by a carrier wave to generate the modulated signal. To simulate phase uncertainty at the practical receiver, a fixed phase offset is introduced at the receiver to obtain the actual received signal:

$$r(t) = \sum_k a_k g(t - kT_s) cos(2\pi f_c t + \varphi_0 + \varphi) + n(t) \tag{10}$$

where $\phi_0$ is the initial phase fixed at 0 in this paper, $\phi = -0.8$ rad is the fixed phase offset at the receiver, and is n(t) additive white Gaussian noise.

### 3.3. Implementation of the SD algorithm

Symbol Decision Aided (Squaring-Difference, SD) algorithm converts suppressed-carrier signals into signals containing discrete carrier components through nonlinear transformation, and then extracts the carrier phase using a phase-locked loop. Its implementation is divided into two stages: preprocessing and phase tracking.

(a). Preprocessing Stage:

First, the received signal is squared to make a signal that has a double carrier frequency component. A band-pass filter separates the squared signal into its nearby frequency components while getting rid of DC and out-of-band noise. The filter design is the set of rules that are used to optimize phase-locked loop preprocessing filtering.

(b). Phase Tracking Stage:

A phase-locked loop structure is used to track the filtered signal in a closed loop. A local numerically controlled oscillator makes a reference signal, which is then compared to the input signal to make an error signal. After a moving average filter smooths out the error, the local phase estimate is changed by a set amount. The step-size parameter is a general rule for how to tune a phase-locked loop.Finally, divide the estimated phase by two to get the original carrier phase.

DOI: 10.54254/2755-2721/2026.34185

### 3.4. Implementation of the Costas loop algorithm

The Costas loop directly uses the received signal for carrier phase tracking, without the need for preprocessing nonlinear transformations. Its core structure includes in-phase (I) and quadrature (Q) branches, with closed-loop tracking achieved through mixing, low-pass filtering, and product phase detection.

(a). quadrature mixing:

The local numerically controlled oscillator generates two quadrature carriers, which are respectively multiplied with the received signal to obtain the (I) and (Q) branch mixing outputs.

(b). low-pass filtering:

The mixed signal passes through a low-pass filter with a cutoff frequency of $f_0$ to extract the baseband components. The filter is designed as a 500th-order FIR filter, adopting a cascaded optimization approach inspired by phase-locked loop (PLL) loop filter design, in order to suppress high-frequency components and noise.

(b). error generation and phase update:

The product of the (I) and (Q) branch baseband signals forms the phase error signal. After being scaled by a step-size factor, it is used to directly adjust the phase of the local carrier. Through iterative convergence, the local phase $\hat{\theta}$ approaches the true carrier phase.

## 4. Experimental results and comparative analysis

### 4.1. Experimental setup and evaluation metrics

The simulation platform used is MATLAB R2023b.The performance is evaluated using the variance of the phase error [14], defined as:

$$Var\left(\theta\right) = \frac{1}{N}\sum_{n=1}^{N}(\theta[n] - \varphi)^2 \tag{11}$$

where $\phi$ is the true carrier phase (−0.8 rad), and N is the number of samples after reaching steady state. A smaller variance means that the algorithm is better at ignoring noise and keeping track of things.

The noise intensity is the range of the additive white Gaussian noise (AWGN), which goes from 0 to 10. This range includes conditions with no noise and very low signal-to-noise ratios. All simulation results are averaged over multiple independent runs to ensure statistical reliability.

### 4.2. Phase tracking performance comparison

Figure 3 compares the variation of phase tracking variance with noise intensity for the SD and Costas algorithms under 4-PAM, 6-PAM, and 8-PAM modulation. The results are shown below.

DOI: 10.54254/2755-2721/2026.34185

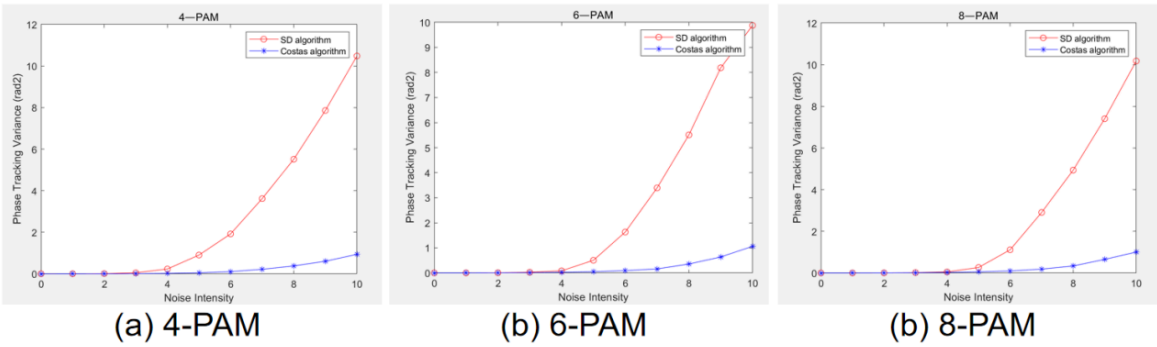



Figure 3. Comparison of phase tracking variance under different noise intensities

**4-PAM modulation:** In the low noise region (noise intensity ≤ 4), both algorithms show very little difference (Costas < 0.005 rad², SD < 0.01 rad²). When the noise level goes above 5, the SD algorithm's variance goes up quickly, reaching 0.03 rad² at a noise level of 7. The Costas algorithm stays stable below 0.08 rad². The SD algorithm's performance gets worse because of error feedback from symbol decision errors at high noise levels. The Costas algorithm, on the other hand, stays stable because it doesn't rely on data.

**6-PAM modulation:** In the low noise region (noise intensity ≤ 4), the variance of both algorithms remains low (Costas < 0.005 rad², SD < 0.01 rad²). The SD algorithm's variance rises quickly when the noise level goes above 5, reaching 0.03 rad² at a noise level of 7. The Costas algorithm stays stable below 0.08 rad². The SD algorithm's performance gets worse because of error feedback from symbol decision errors when noise levels are high. The Costas algorithm, on the other hand, stays stable because it doesn't use data to help it.

**8-PAM modulation:** The spacing between symbols is only about 0.286, which is the hardest case. The Costas algorithm stays strong in the noise level range of 0 to 5 (variance < 0.015 rad²), goes up to 0.025 rad² in the 6 to 7 range, and quickly goes up to 0.06 rad² when the noise level goes over 8. It doesn't lose lock, though. The SD algorithm, on the other hand, shows a turning point in performance at a noise level of 5, with the variance reaching 0.15 rad² at a noise level of 6. After a noise level of 7, the phase estimate completely diverges, which means that the lock is lost.

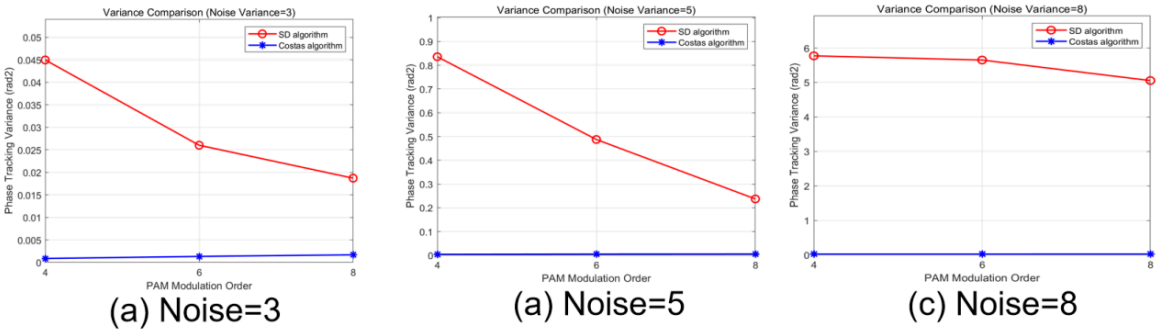



Figure 4. The impact of modulation order on phase variance under fixed noise intensity

In Figure 4, the phase variance is shown to decrease with increasing modulation order for both algorithms, with the Costas loop outperforming the SD algorithm at high orders (8-PAM), while the SD algorithm performs better at lower to medium modulation orders (4-PAM and 6-PAM).

In conditions of low noise (intensity = 3), medium noise (intensity = 5), and high noise (intensity = 8), the results shown in Figure 4 consistently show that the phase variance of both algorithms is

DOI: 10.54254/2755-2721/2026.34185

strongly linked only to the modulation order and is not affected by additive noise in the range of 3 to 8, which shows that the algorithms are very robust. As the modulation order goes up from 4-PAM to 8-PAM, the phase variance of the Costas loop goes down from 0.05 rad² to 0.04 rad² and then drops quickly to 0.01 rad². This shows a pattern of gradual decline followed by a sharp drop, with the best state being at 8-PAM. Meanwhile, the variance of the SD algorithm decreases from 0.045 rad² to 0.025 rad² and then to 0.018 rad², displaying a continuous downward trend with a progressively narrowing rate of decrease. At 4 - PAM and 6 - PAM, the SD algorithm marginally outperforms the Costas loop, whereas at 8 - PAM, the Costas loop surpasses the SD algorithm with a substantial accuracy advantage. So, for PAM systems, the SD algorithm works better with low to medium modulation orders, while the Costas loop is better at tracking phase at higher modulation orders.

### 4.3. Analysis of convergence speed and steady-state characteristics

Along with steady-state variance, convergence speed is another important way to judge how well an algorithm works. This part compares the SD algorithm and the Costas loop in terms of how quickly they converge and how well they track the steady-state phase. It does this by looking at phase error curves for 4-PAM, 6-PAM, and 8-PAM modulation schemes, as well as noise variances of 3, 5, and 8. Figure 5 compares the convergence speed of the SD algorithm and the Costas loop, showing that the SD algorithm achieves fast and stable phase tracking across different PAM orders and noise levels, while the Costas loop converges much more slowly and struggles to lock effectively.Along with steady-state variance, convergence speed is another important measure of how well an algorithm works. This part compares and looks at the SD algorithm and the Costas loop's convergence speed and steady-state phase tracking performance. It does this by looking at phase error curves for 4-PAM, 6-PAM, and 8-PAM modulation schemes, as well as noise variances of 3, 5, and 8.

**Convergence Speed Analysis:** The SD algorithm always shows fast convergence in all test cases with 4-PAM, 6-PAM, and 8-PAM modulation schemes and noise levels of 3, 5, and 8. Starting from an initial phase error of approximately −1.6 rad², it can quickly enter the steady-state fluctuation range within 500–1000 samples. There is only a small delay in the start of convergence and a small rise in the steady-state fluctuation amplitude in high-noise situations.In contrast, the Costas algorithm exhibits extremely slow convergence. Over a 5000-sample observation period, the phase error remains near the initial −1.6 rad², approaching zero only at a very slow rate. Only in the high-noise 8-PAM scenario is a slight rise observed. The algorithm doesn't work well for phase convergence and doesn't respond well to changes in noise, which shows that it isn't very adaptable.

**Steady-State Characteristic Analysis:** The SD algorithm keeps a small steady-state phase error in all test cases, with the average being close to 0 rad. In low-noise scenarios, the fluctuation range is ±0.2 rad, in medium-noise scenarios it is ±0.3 rad, and in high-noise scenarios, it is ±0.4 rad. There is no divergence or phase drift, and the steady-state fluctuation range decreases with the increase in modulation order. The tracking accuracy is better at higher modulation orders.

The Costas algorithm, on the other hand, keeps the steady-state phase error between −1.6 and −1.4 rad in most cases, which is a big difference from the ideal 0 rad phase point and means that phase locking doesn't work. The error only goes up a little, to about −1.5 rad, in the high-noise 8-PAM case.Moreover, its steady-state performance does not change when the modulation scheme or noise changes, which means it does not meet the practical requirements for phase synchronization.

DOI: 10.54254/2755-2721/2026.34185

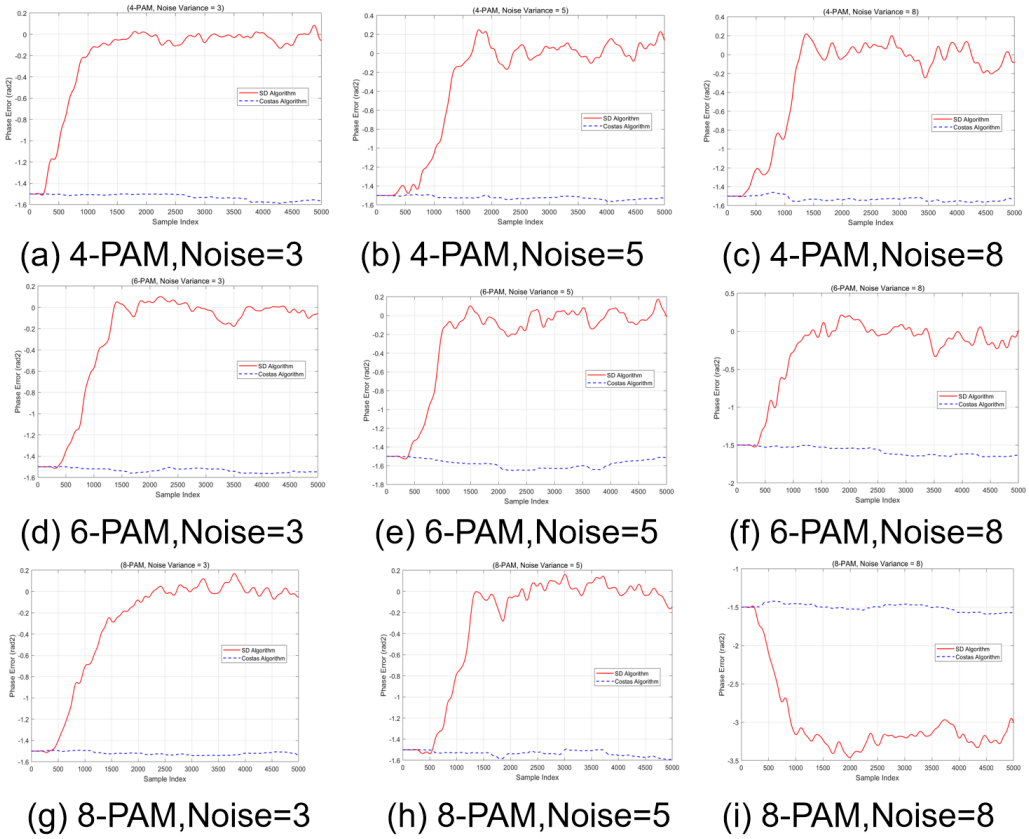



Figure 5. Phase tracking convergence comparison plot

## 4.4. Computational complexity analysis

The cost of hardware implementation and power use are both affected by the computational complexity, which is why it is an important factor in choosing an engineering solution.This part looks at the two algorithms from two points of view: the number of multiply-and-add operations they can do and the storage resources they need.

Table 1. Comparison of computational complexity between the two algorithms

| | Algorithm Module | Operation Type | Operations per Symbol |
|---|---|---|---|
| SD Algorithm | Squaring Operation | Multiplication | 1 |
| | Bandpass Filtering | Multiply-and-Add | 500 |
| | PLL Phase Detection | Multiplication | 1 |
| | Loop Filtering | Multiply-and-Add | 10 |
| | VCO Update | Trigonometric | 2 |
| Costas Algorithm | Quadrature Mixing | Multiplication | 2 |
| | Low-Pass Filtering | Multiply-and-Add | 1000 |
| | Error Generation | Multiplication | 1 |
| | Loop Filtering | Multiply-and-Add | 10 |
| | VCO Update | Trigonometric | 2 |

DOI: 10.54254/2755-2721/2026.34185

Table 1 shows that the Costas loop has about twice as many multiply-and-add operations as the SD algorithm. The primary difference arises from the fact that the Costas loop requires 500-tap low-pass filtering on both the I and Q channels, whereas the SD algorithm only requires 500-tap bandpass filtering on a single signal path.Both algorithms need memory to store the filter coefficients and state variables. The Costas loop, on the other hand, needs more memory to keep track of an extra set of filter states because it filters in two channels.

It is important to note that the preprocessing filter in the SD algorithm can make hardware implementations less demanding on processing power by using a polyphase structure. A similar optimization method can be used for the dual-channel filtering in the Costas loop. In ASIC implementations, the real difference in resource use between the two algorithms may be smaller than what the theoretical analysis says.

## 5. Conclusion and future work

This study provides a systematic comparative analysis of the Squared-Difference (SD) algorithm and the Costas loop algorithm for carrier synchronization in high-order Pulse Amplitude Modulation (PAM) systems.The Costas loop is better at tracking phases and handling noise when there is a lot of noise and a lot of modulation, but it costs more to do so. The SD algorithm, on the other hand, is more likely to lose quality when there is a lot of noise, even though it works better when there aren't many resources available. These results provide quantitative direction for algorithm selection in practical engineering applications and establish a three-dimensional evaluation framework (accuracy, noise resilience, and complexity) to inform future research.

Future research may involve the development of hybrid architectures that combine the simplicity of the SD algorithm with the noise resilience of the Costas loop, as well as the application of machine learning techniques for dynamic adaptation to varying channel conditions. Later questions may include complex modulation formats (like QAM) and multipath channel conditions to figure out if the algorithms can be used.Research that focuses on hardware, such as creating low-power, high-efficiency carrier synchronization modules for IoT and edge computing applications, is also an important area to keep an eye on in the future.

## References

[1] Boxel S C J V , Volpe B , Vermorken B L , et al. Vestibular implant stimulation: Pulse amplitude modulation versus combined pulse rate and amplitude modulation [J]. IOP Publishing Ltd [2026-04-08]. DOI: 10.1088/1741-2552/adc33a.

[2] Kaleris K , Hatziantoniou P , Stelzner B , et al.Laser-sound reproduction by pulse amplitude modulation audio streams [J].Scientific Reports, 2024, 14(1). DOI: 10.1038/s41598-024-62382-8.

[3] Bammidi E R , Kallfass I .Design of an error detector circuit for BPSK costas loop for carrier synchronization in millimeter-wave receivers [J]. 2018.DOI: 10.23919/GEMIC.2018.8335098.

[4] Ferramola D , Casilli T , Pradella S , et al.Effective Lock Detectors Based on Costas Loop Output for SBPSK Mobile Communications [J].Access, IEEE, 2025, 13(000): 22551-22565. DOI: 10.1109/ACCESS.2025.3536360.

[5] Liu L , Xue B , Yuan P , et al. DSP-Free Coherent Detection for Short-Reach Frequency-Synchronous Optical Networks [J]. Lightwave Technology, IEEE/OSA Journal of (J-LT), 2024, 42(15): 6. DOI: 10.1109/JLT.2024.3386700.

[6] Yang Z , Liu X. A Dual Squared Difference Loop Based Frequency Estimation Algorithm for Robust Carrier Synchronization [C]//2025 6th International Conference on Information Science, Parallel and Distributed Systems (ISPDS). 0 [2026-04-08]. DOI: 10.1109/ISPDS67367.2025.11391173.

[7] Raghuram C N , Kumar G K , Mahankali J K , et al. Design of energy-efficient approximate squaring circuits for error-resilient signal and image processing applications [J]. Ain Shams Engineering Journal, 2025, 16(9). DOI: 10.1016/j.asej.2025.103523.

DOI: 10.54254/2755-2721/2026.34185

[8] Hassan S , Amin Y , Wei S , et al. Assessing Performance of Silicon Photonic Modulators for Pulse Amplitude Modulation [J]. IEEE Journal of Selected Topics in Quantum Electronics, 2018, PP(6): 1-1. DOI: 10.1109/JSTQE.2018.2854607.
[9] Secondini M , Forestieri E . Direct Detection of Bipolar Pulse Amplitude Modulation [J]. Lightwave Technology, IEEE/OSA Journal of (J-LT), 2020, 38(21): 10. DOI: 10.1109/JLT.2020.3007584.
[10] Claudia, Calcagno, Sarayu, et al. Registration of dynamic contrast-enhanced MRI of the common carotid artery using a fixed-frame template-based squared-difference method. [J]. Journal of Magnetic Resonance Imaging Jmri, 2014.
[11] Aygül, Mehmet Ali, Nazzal M , Arslan, Hüseyin.Using OMP and SD algorithms together in mm-Wave mMIMO channel estimation [J]. Signal, Image and Video Processing, 2022, 16(5): 1205-1213. DOI: 10.1007/s11760-021-02071-5.
[12] Sinhababu F , Mukherjee A , Sarkar S , et al. Impact of phase modulator on the performance of Costas loop [J]. Annals of telecommunications, 2025(3/4): 80. DOI: 10.1007/s12243-024-01048-y.
[13] Liu J , Wissing A , Movaghar G , et al. Dual-Polarization Optical Costas Loop for DSP-Free Homodyne Short-Reach Links [J]. Journal of Lightwave Technology, 2025: 1-9. DOI: 10.1109/jlt.2025.3533422.
[14] Tanaka T . Universal representation of undulator phase errors [J]. Physical Review Accelerators and Beams, 2018, 21(11). DOI: 10.1103/PhysRevAccelBeams.21.110704.