



Politechnika Wrocławska

Modulacja APSK

Autorzy: Michał Skubek 248823
Michał Tkacz 248869

Prowadzący: dr hab. inż.
Henryk Maciejewski

Wrocław 2020

SPIS TREŚCI

Wstęp.....	3
Cel projektu	3
Etapy projektu	3
Realizacja projektu	3
Problematyka projektu [Etap 1].....	3
Modulacja w ujęciu ogólnym	3
Modulacja APSK.....	4
Implementacja symulatora [Etapy 2 i 3]	4
Narzędzia.....	4
Wykorzystane elementy bibliotek zewnętrznych.....	4
Pojedynczy test.....	5
Automatyzacja testów	5
Eksperymenty [Etapy 4 i 5].....	5
Metodologia.....	5
Wyniki i analiza.....	5
Uwagi i wnioski.....	9
Załączniki	9

WSTĘP

Cel projektu

Celem projektu było zbadanie wpływu doboru parametrów modulacji APSK na jakość przesyłu. W tym celu, w pierwszym kroku konieczna była implementacja programowa symulatora modulatora APSK. Następnie, przy jego użyciu wygenerowaliśmy zestaw danych, który poddaliśmy analizie. Ten dokument stanowi dokumentację realizacji projektu i omawia jej kolejne etapy.

Etapy projektu

Poniżej znajduje się ogólny schemat, według którego realizowaliśmy projekt. Bardziej szczegółowe informacje znajdują się w odpowiednich rozdziałach tego raportu.

Etap 1. Problematyka projektu oraz zebranie wymagań.

Pierwszym krokiem realizacji projektu było wybranie tematu projektu. Po zapoznaniu się z listą dostępnych tematów oraz fundamentalnymi podstawami teoretycznymi stojącymi za każdym z problemów, postanowiliśmy podjąć się realizacji zagadnienia związanego z modulacją APSK. Następnie skupiliśmy się na głębszym zapoznaniu się z tym problemem. Poszerzyliśmy naszą wiedzę na temat modulacji APSK oraz przedyskutowaliśmy wymagania projektowe.

Etap 2. Planowanie rozwiązań oraz wybór narzędzi.

Po zebraniu wymagań projektowych, kolejnym krokiem była analiza możliwych rozwiązań oraz wybór możliwie najlepszej opcji. Jako środowisko pracy wybraliśmy język *Python* wraz z wykorzystaniem dodatkowej biblioteki *Komm*. Uzyskane dane poddaliśmy analizie przy użyciu programu *Microsoft Excel 2017*.

Etap 3. Implementacja symulatora

Implementacja symulatora w języku *Python* okazała się bardzo prosta dzięki wykorzystaniu biblioteki *Komm*. Szczegółowe informacje dotyczące implementacji symulatora jak i samej biblioteki znajdują się w rozdziale *Implementacja symulatora*.

Etap 4. Przeprowadzenie eksperymentów

Eksperymenty polegały na dokonaniu modulacji sygnału przy określonych parametrach, przesłaniu przez kanał transmisyjny, a następnie demodulacji. Wynikiem eksperymentu jest współczynnik *bit-error-rate* obliczany na podstawie różnic między sygnałem wejściowym i wyjściowym.

Etap 5. Analiza wyników

Dane uzyskane w wyniku działania programu zaimportowaliśmy do programu *Microsoft Excel 2017*. Następnie utworzyliśmy odpowiednie tabele oraz wykresy, które zostały załączone do raportu.

Etap 6. Sporządzenie raportu

Ostatni etap stanowi podsumowanie całej pracy nad projektem. W kolejnych rozdziałach raportu zawarte są informacje odnoszące się bezpośrednio do każdego z etapów realizacji. Dokument zwieńczony jest wnioskami oraz uwagami autorów.

REALIZACJA PROJEKTU

Problematyka projektu [Etap 1]

Modulacja w ujęciu ogólnym

Modulacja to proces samorzutnej lub celowej zmiany jednego lub więcej parametrów okresowej fali nośnej sygnału. W przypadku fali sinusoidalnej, zmianie mogą ulegać amplituda, częstotliwość lub faza drgań, a w przypadku fali prostokątnej – szerokość, amplituda, pozycja oraz gęstość impulsów. Modulację uzyskuje się oddziałując na dany sygnał sygnałem modulującym, w wyniku czego otrzymywany jest sygnał zmodulowany. Modulacja może być procesem analogowym

(AM, FM, PM, QAM, SM, SB) jak i cyfrowym (ASK, APSK, CPM, FSK, MFSK, MSK, OOK, PPM, PSK, QAM, SC-FDE, TCM, WDM). W modulacji analogowej, fala nośna modulowana jest w sposób ciągły, a w modulacji cyfrowej w sposób dyskretny.

Modulacja APSK

Jedną z metod modulacji cyfrowej jest APSK (ang. *Amplitude and Phase-Shift Keying* lub *Asymmetric Phase-Shift Keying*), czyli amplitudowo-fazowa modulacja sygnału, będąca jedną z bardziej wydajnych dwuwymiarowych technik modulacji sygnału. Polega ona na zmianie zarówno amplitudy jak i fazy fali nośnej, innymi słowy łączy technikę kluczowania amplitudy (ASK) z kluczowaniem fazy (PSK).

APSK jest modulacją zespoloną, w której konstelacja jest sumą zbiorów konstelacji składowej modulacji PSK zwanych pierścieniami:

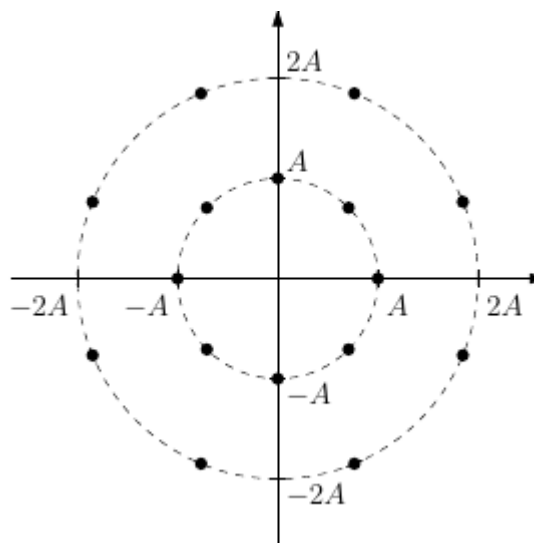
$$S = \bigcup_{k \in [0, K)} S_k,$$

gdzie K – liczba pierścieni oraz:

$$S_k = \left\{ A_k \exp\left(j \frac{2\pi i}{M_k}\right) \exp(j\phi_k) : i \in [0 : M_k) \right\},$$

gdzie M_k – rząd, A_k – amplituda, ϕ_k – przesunięcie fazowe k -tego pierścienia. Rzędy poszczególnych pierścieni nie muszą być potęgą liczby 2, ale rząd wynikowy modulacji być już musi.

Przykładowy diagram konstelacji dla $(M_0, M_1) = (8, 8)$ gdzie $(A_0, A_1) = (A, 2A)$ i $(\phi_0, \phi_1) = (0, \pi/8)$:



Implementacja symulatora [Etapy 2 i 3]

Narzędzia

Symulator modulatora APSK został napisany w języku programowania Python 3.x, przy wykorzystaniu dodatkowej biblioteki [Komm](#). W symulacji wykorzystujemy kanał AWGN.

Wykorzystane elementy bibliotek zewnętrznych

Kanał transmisji

Jako kanał transmisji wykorzystaliśmy **Komm.AWGNChannel**. W tym kanale wartość sygnału wyjściowego jest obliczana ze wzoru: $Y_n = X_n + Z_n$, gdzie X_n to sygnał wejściowy, Y_n to sygnał wyjściowy, a Z_n oznacza niezależny i równomiernie rozłożony szum o rozkładzie normalnym wokół 0. Aby otrzymać sygnał wyjściowy należy wywołać obiekt klasy **Komm.AWGNChannel** z parametrem wejściowym będącym sygnałem wejściowym.

Modulacja APSK

Do modulacji APSK sygnału wykorzystaliśmy **Komm.APSKModulation**.

Do modulacji sygnału została wykorzystana metoda *modulate(bits)*, która jako argument przyjmuje tablicę bitów do zakodowania. Długość tablicy powinna być wielokrotnością własności *bits_per_symbol*, która opisuje liczbę bitów odpowiadających jednemu symbolowi po zakodowaniu, co jest obliczane na podstawie podanych rzędów modulacji.

Do demodulacji sygnału została wykorzystana metoda *demodulate(modulated_signal[, decision_method='hard'])*, przyjmująca 2 parametry – przekodowany na symbole sygnał oraz drugi opcjonalny parametr – metodę decyzyjną.

Pojedynczy test

Polega na przesłaniu losowego sygnału poddanego modulacji przez kanał transmisji, a następnie demodulacji. Wynikiem testu jest *BER* – *bit-error-rate*, obliczany na podstawie różnicy pomiędzy sygnałem wejściowym (przed modulacją) a wyjściowym (po demodulacji), ze wzoru

$$BER = \frac{\text{liczba błędnych bitów}}{\text{liczba wszystkich bitów}}$$

Przykładowy kod pojedynczego testu został przedstawiony niżej.

channel – obiekt klasy **Komm.AWGNChannel**

modulation – obiekt klasy **Komm.APSKModulation**

sygnał_wejsciowy – tablica binarna (dozwolone wartości: 0, 1) o długości będącej wielokrotnością *modulation.bits_per_symbol*

```
def single_test2(channel, modulation, sygnał_wejsciowy):  
    sygnał_wyjsciowy = modulation.demodulate(  
        channel(modulation.modulate(sygnał_wejsciowy))  
    )  
    ber = get_ber(sygnał_wejsciowy, sygnał_wyjsciowy)  
    return round(ber/len(sygnał_wejsciowy), 3)
```

Automatyzacja testów

Aby usprawnić wykonywanie pomiarów zastosowaliśmy automatyczne testy. Lista parametrów dla modulacji znajduje się w pliku „*tests_cfg.json*”. Automatyzacja polegała na wygenerowaniu sygnału do przesłania, a następnie utworzeniu obiektów reprezentujących wykorzystaną modulację oraz kanał transmisyjny i wykonaniu pojedynczych testów określoną liczbę razy.

Eksperymenty [Etapy 4 i 5]

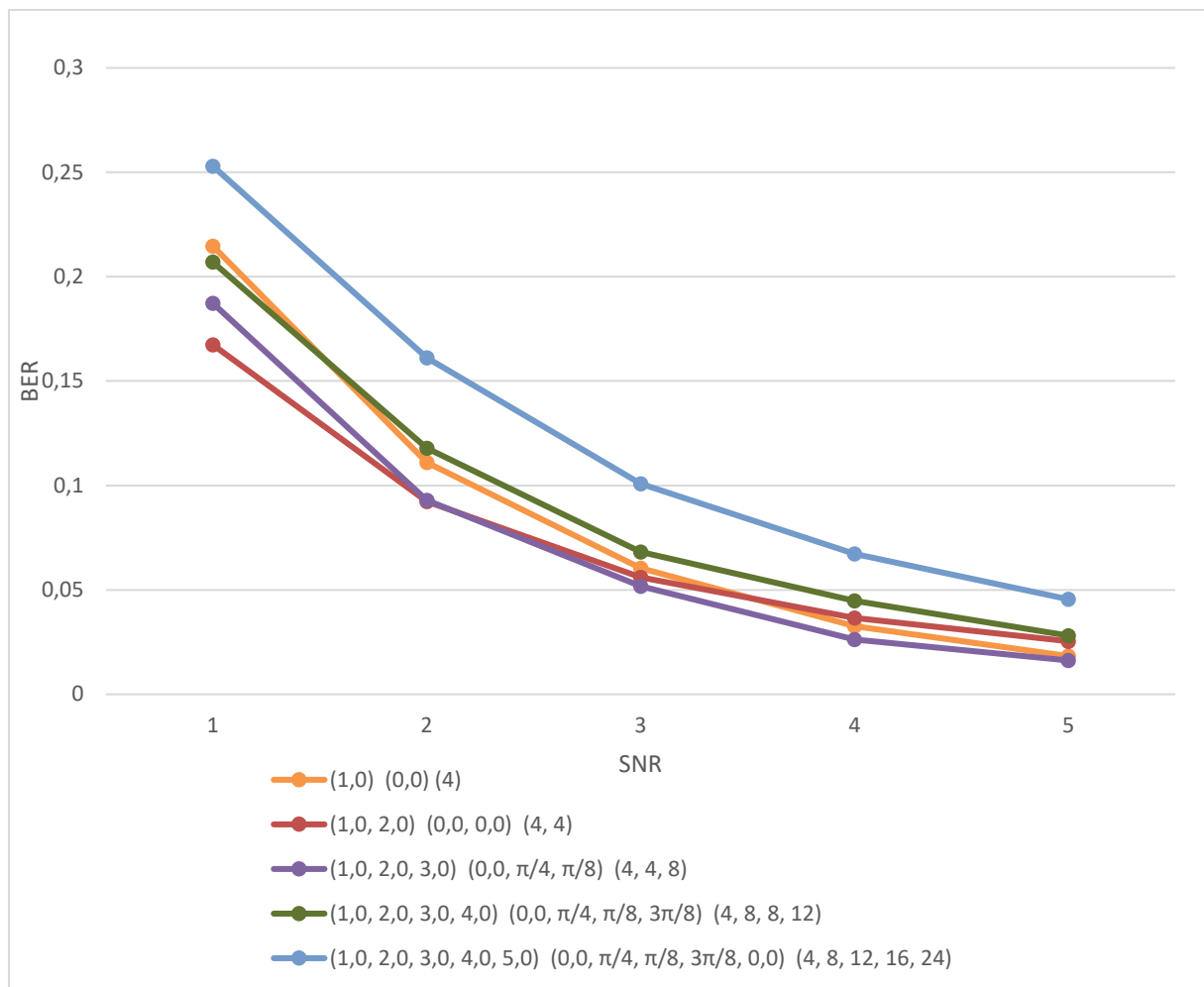
Metodologia

Do testowania różnych modulacji APSK wykorzystywaliśmy raz wygenerowany losowy sygnał. Każdy zestaw parametrów modulacji (amplitudy, przesunięcia fazowe, rzędy) był testowany dla pięciu wartości *signal-to-noise-ratio*: $SNR \in \{1, 2, 3, 4, 5\}$. Dla każdej pary: *zestaw parametrów* – *wartość SNR* test był powtarzany 10 razy w celu uśrednienia wyników. Za każdym razem notowana była wartość *BER*. Uzyskane wyniki w wyniku działania symulatora zostały poddane analizie przy użyciu Microsoft Excel 2017.

W tym dokumencie przedstawione zostały najbardziej interesujące wyniki. Pełna tabela z uzyskanymi danymi znajduje się w pliku *wyniki.xlsx* załączonym do sprawozdania.

Wyniki i analiza

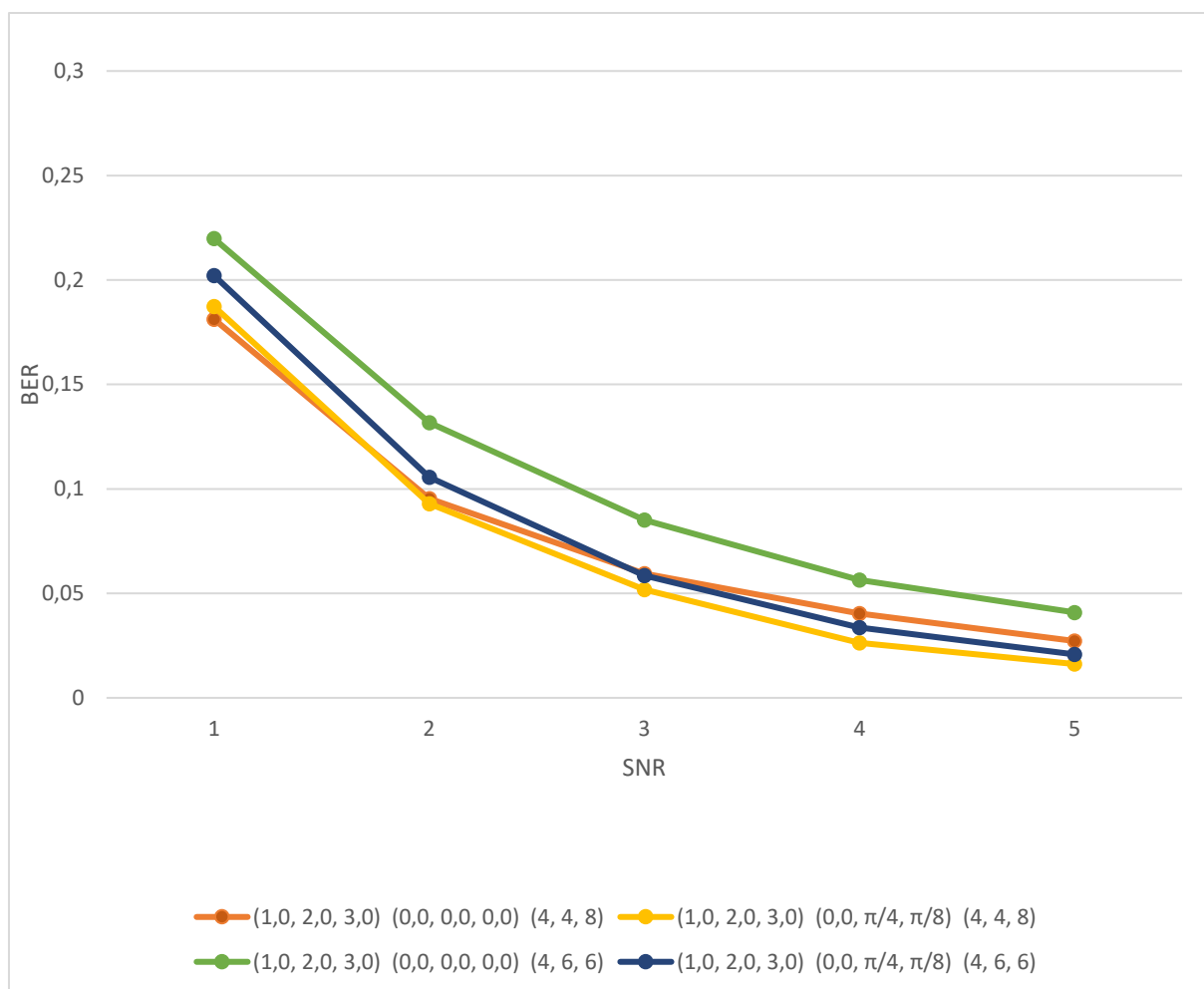
Poniższy wykres przedstawia serie najbardziej optymalne dla kolejnych klas TS, gdzie $TS \in \{2, 3, 4, 5, 6\}$.



Rysunek 1 - wykres zależności BER od SNR

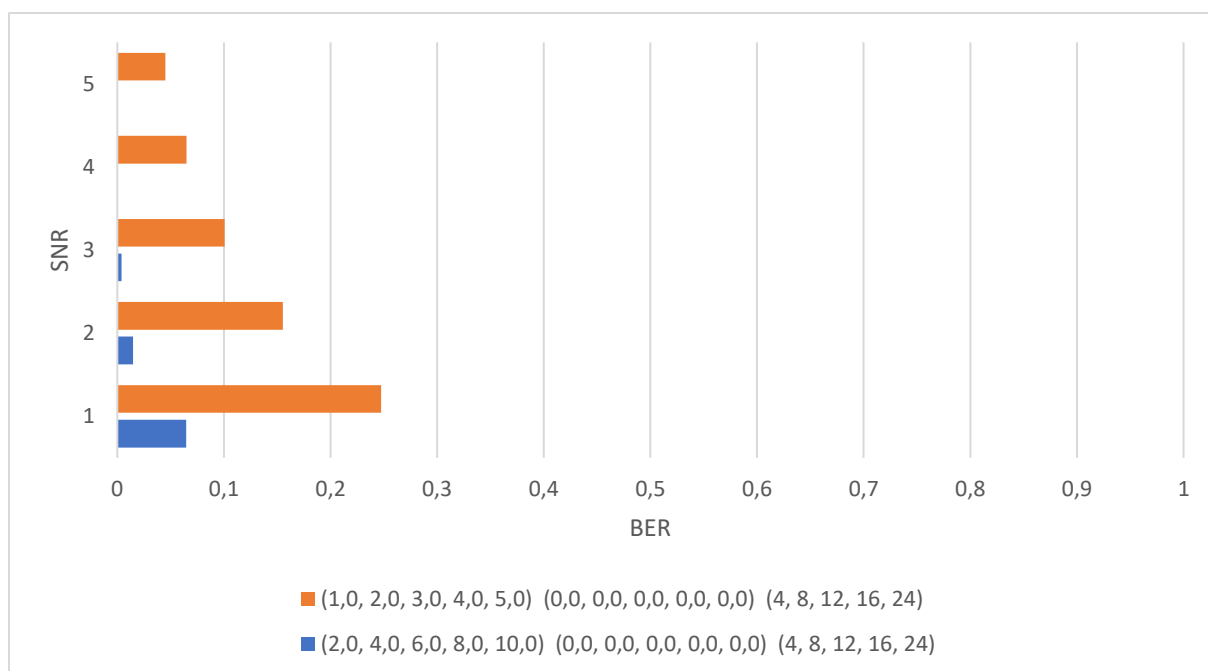
Amplitude	Offset	Orders	SNR	TS	BER[AVG]
(1,0, 2,0, 3,0)	(0,0, $\pi/4$, $\pi/8$)	(4, 4, 8)	1	4	0,1873
			2	4	0,0929
			3	4	0,0518
			4	4	0,0263
			5	4	0,0162

Tabela 1 - wyniki testów dla jednego z zestawu modulacji wykorzystanych do stworzenia rysunku 1

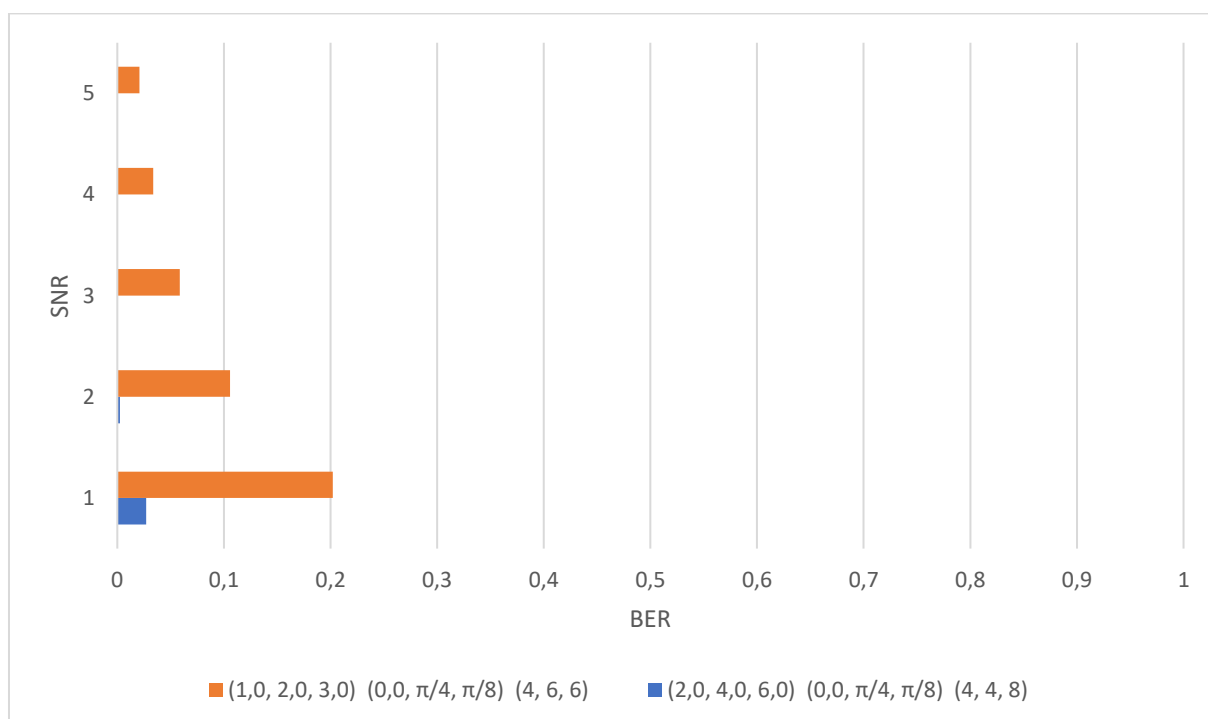


Rysunek 2 - wpływ przesunięcia fazowego oraz rozłożenia punktów konstelacji na BER

Na powyższym wykresie można zauważyć wpływ przesunięcia fazowego kolejnych konstelacji oraz rozłożenia punktów konstelacji na BER. Testy przeprowadzane dla innych modulacji wykazały podobną zależność – odpowiedni dobór tych parametrów zmniejsza prawdopodobieństwo pojawienia się błędów w transmisji. Jest to spowodowane zwiększeniem odległości pomiędzy punktami konstelacji, co pozwala na zwiększenie dokładności demodulacji zakłóconego sygnału.



Rysunek 3 - wpływ amplitudy na wartość BER



Rysunek 4 - wpływ amplitudy na wartość BER

Rysunek 3 oraz rysunek 4 pokazują, że zwiększenie amplitudy ma duży wpływ na wartość BER, ponieważ im większe amplitudy, tym większe odległości między punktami konstelacji, zatem zwiększa się odporność na zakłócenia.

Uzyskane dane nie stanowią pełnego rozwiązania problemu. Przeprowadzone testy są ograniczone zarówno w dziedzinie doboru parametrów poszczególnych prób, jak i liczby prób oraz ich dokładności. Liczba możliwych konfiguracji parametrów (amplitudy, przesunięcia fazowe, rzędy, SNR, długość sygnału) jest bardzo duża, co powoduje, że zbadanie ich metodą testowania jest trudne.

Aby uzyskać bardziej rzetelne wyniki, konieczne byłoby posłużenie się złożonym aparatem matematycznym i wykonanie odpowiednich obliczeń.

UWAGI I WNIOSKI

Celem projektu było zbadanie wpływu doboru parametrów modulacji APSK na jakość przesyłu sygnału. W tym celu zaimplementowany został programowy symulator modulacji APSK, napisany w języku Python z użyciem biblioteki Komm. Z jego użyciem przeprowadzone zostały testy modulacji dla różnych konfiguracji parametrów, a uzyskane wyniki poddaliśmy analizie.

Uzyskane dane wyraźnie pokazują, że dobór parametrów ma istotny wpływ na jakość transmisji. Szczególnie znaczące okazuje się zwiększenie amplitudy, a co za tym idzie odległości pomiędzy punktami na diagramie konstelacji oznaczającymi symbole. Widoczna jest zależność, że im większe różnice pomiędzy kolejnymi amplitudami, tym mniejszy współczynnik BER – rysunek 3 i rysunek 4.

Można jednak zauważyć, że manipulacja zarówno amplitudą jak i przesunięciem fazowym generuje jeszcze lepsze, bardziej optymalne wyniki, co widoczne jest na rysunku 2. Nie bez znaczenia pozostaje dobór liczby punktów w poszczególnych rzędach konstelacji.

Modulacje o większej szybkości transmisji są bardziej podatne na losowe zakłócenia sygnału, a co za tym idzie ich współczynnik BER jest wyższy. Dla kanałów o niskim poziomie zakłóceń osiągnane przez nie wyniki są porównywalne do ich wolniejszych odpowiedników, co pozwala na ich wykorzystanie.

Podsumowując, dobór najbardziej optymalnej konfiguracji parametrów modulacji może stanowić pewne wyzwanie. Nie mniej jednak, jest to ważne, aby uzyskać transmisję sygnału o jak najmniejszym poziomie zakłóceń.

ZAŁĄCZNIKI

- [Repozytorium na GitHub \[LINK\]](#) – zawiera wszystkie pliki źródłowe oraz raport z projektu.
- Plik *wyniki.xlsx* – zawiera pełne dane uzyskane w wyniku działania symulatora.