

Sprawozdanie

KOD BCH – etap początkowy

1. Wstęp merytoryczny

Kody BCH (Bose-Chaudhuri-Hocquenghema) wynalezione zostały przez Alexisa Hocquenghema oraz Raj Bose i D.K. Ray-Chaudhurię. Jedną z najważniejszych cech kodów BCH jest możliwość precyzyjnego sterowania ich zdolnościami korekcyjnymi. Zdolność ta umożliwia naprawę wielu źle odebranych bitów danych. Do dekodowania kodów BCH można użyć algebraicznej metody syndromów. Kody BCH są szeroko stosowane w łączności satelitarnej, sterownikach dysków, odtwarzaczach CD i DVD oraz w dwuwymiarowych kodach paskowych. Wśród kodów BCH największe znaczenie odgrywają kody binarne, dla których zachodzi własność mówiąca, że dla każdej liczby całkowitej m i $t < 2^{m-1}$ będzie istniał kod BCH, którego długość będzie równa $n = 2^m - 1$. Kod ten będzie miał zdolność do korekcji t błędnie otrzymanych bitów.

m	n	k	t	m	n	k	t	m	n	k	t	m	n	k	t
3	7	4	1	6	63	10	13	7	127	15	27	8	255	123	19
4	15	11	1			7	15			8	31			115	21
		7	2	7	127	120	1	8	255	247	1			107	22
		5	3			113	2			239	2			99	23
5	31	26	1			106	3			231	3			91	25
		21	2			99	4			223	4			87	26
		16	3			92	5			215	5			79	27
		11	5			85	6			207	6			71	29
		6	7			78	7			199	7			63	30
6	63	57	1			71	9			191	8			55	31
		51	2			64	10			187	9			47	42
		45	3			57	11			179	10			45	43
		39	4			50	13			171	11			37	45
		36	5			43	14			163	12			29	47
		30	6			36	15			155	13			21	55
		24	7			29	21			147	14			13	59
		18	10			22	23			139	15			9	63
		16	11							131	18				

Rysunek 1 Parametry kodów BCH

2. Wykonanie

Do realizacji skryptu w Matlabie przeprowadzającego symulację kodu BCH wykorzystane zostały funkcje z biblioteki Communications. Do zakodowania wiadomości wykorzystana została funkcja `bchenc(msg,n,k)`. Parametry, które ta funkcja przyjmuje oznaczają:

- * `msg` – wiadomość przeznaczona do zakodowania,
- * `n` – długość kodowa słowa, wyznaczona na podstawie tabeli z parametrami kodów BCH,
- * `k` – długość przekazanej wiadomości, wyznaczone na podstawie tabeli z parametrami kodów BCH.

Kolejnym przeprowadzonym etapem było umieszczenie w zakodowanej wiadomości losowych błędów. Do wprowadzenia zakłóceń wykorzystana została funkcja `randi([imin, imax], n, p)`.

Parametry tej funkcji oznaczają:

- * `[imin, imax]` – przedział, z którego wylosowane liczby,
- * `n` – ilość liczb do wylosowania.

Następnym przeprowadzonym etapem było zdekodowanie wiadomości. Zostało to wykonane przy pomocy funkcji `bchdec(noisycode, n, k)`. Parametry tej funkcji oznaczają:

- * `noisycode` – jest to wiadomość z wprowadzonymi zakłóceniami,
- * `n` – długość kodowa słowa,
- * `k` – długość przekazanej wiadomości.

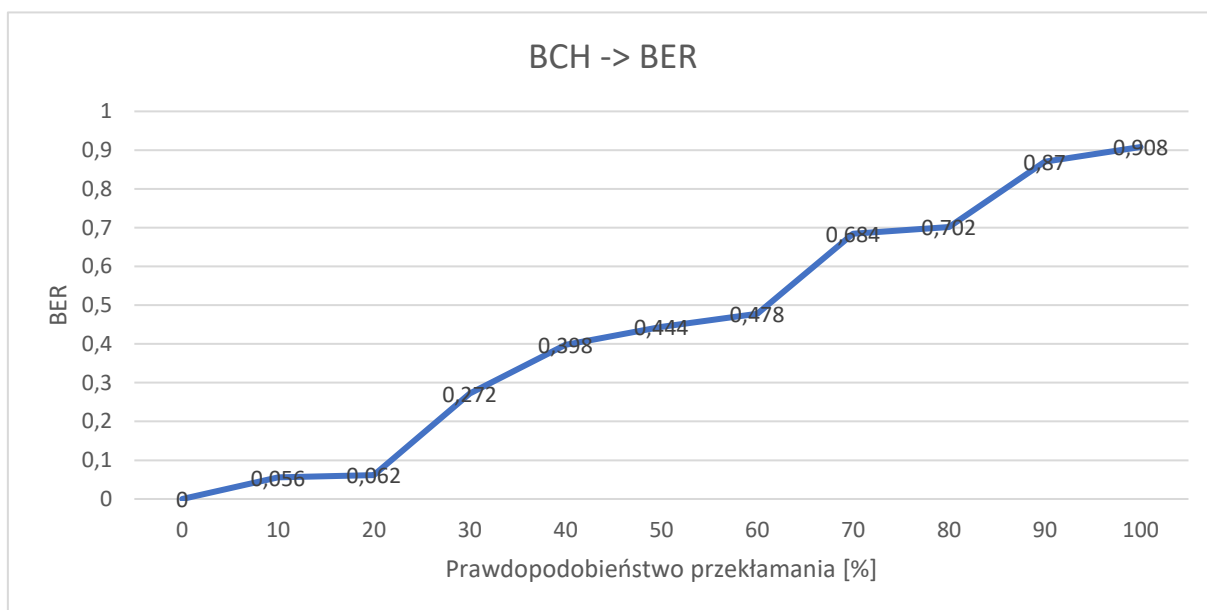
Ostatnim etapem wykonanego programu było obliczenie ilości parametru BER oznaczającego iloraz błędnie odebranych bitów w zdekodowanej wiadomości do bitów w wiadomości wysłanej. Symulacja została przeprowadzona dla różnych poziomów prawdopodobieństwa przekłamania oraz odchylenia standardowego. Dla każdego z tych poziomów zostało wykonanych 100 pomiarów. Wyniki pomiarów są wyświetlane na koniec działania symulatora.

3. Wyniki symulacji

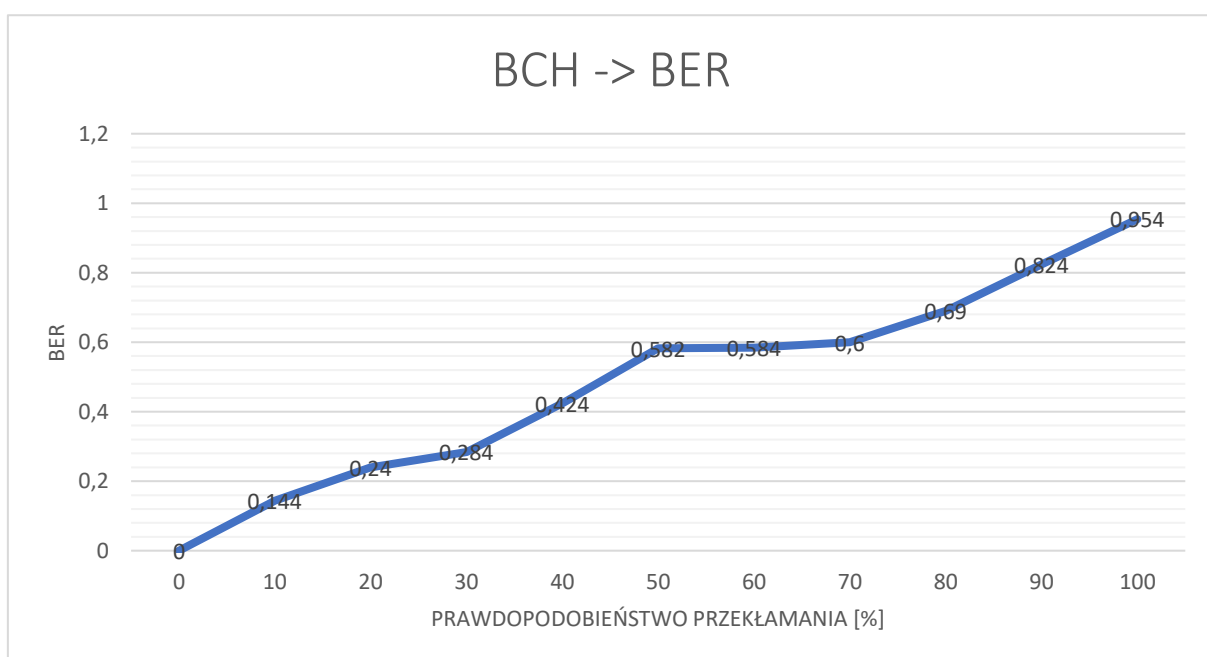
Po fazie implementacji symulatora przeprowadzone zostały jego testy. Na przedstawionych wykresach ukazane zostały wykresy zależności BER od prawdopodobieństwa przekłamania oraz od odchylenia standardowego. Testy te przeprowadzone były dla kodów BCH o różnych parametrach.

3.1 Wykresy zależności BER od prawdopodobieństwa przekłamania

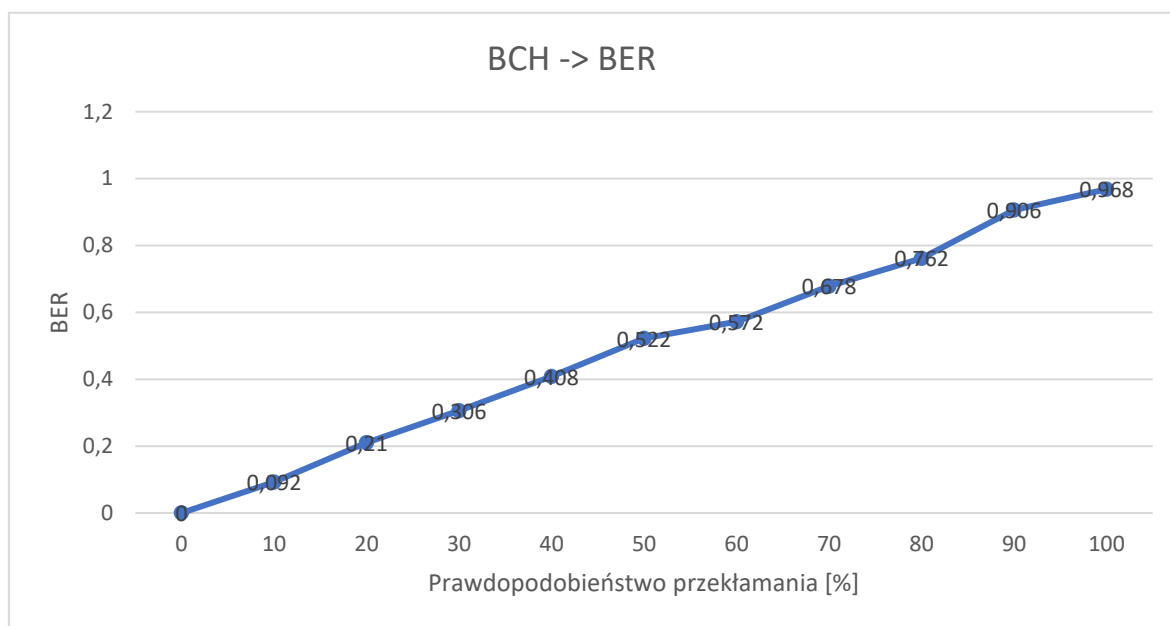
a) dla $n = 15$, $k = 11$



b) dla $n = 31$, $k = 26$

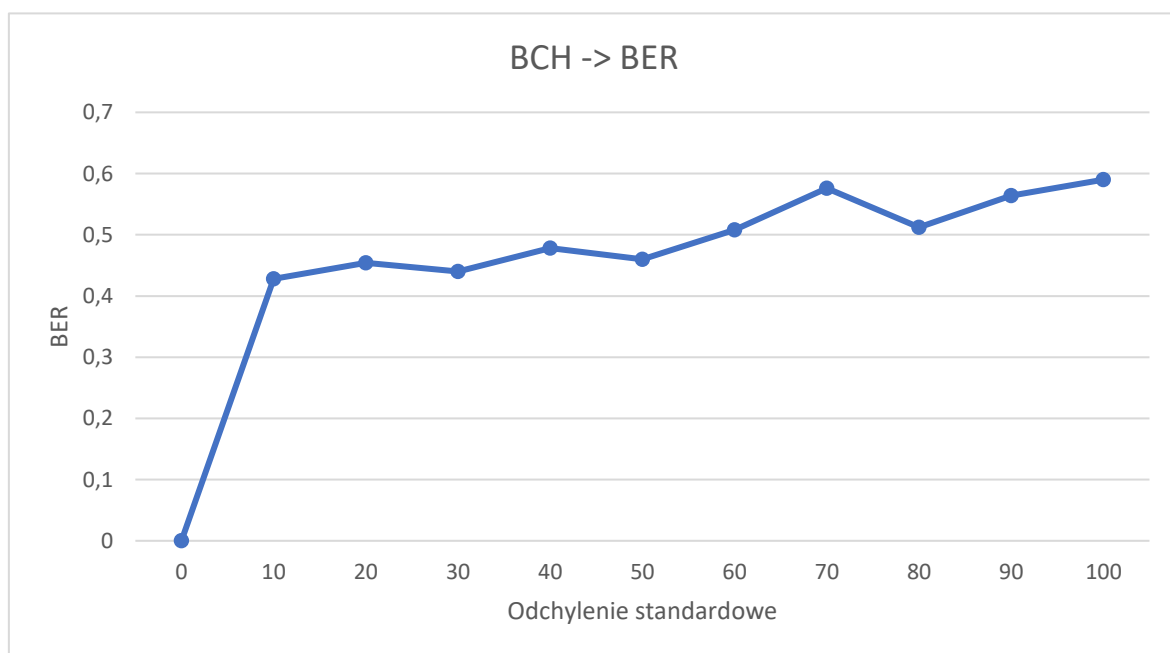


c) dla $n = 63$, $k = 57$

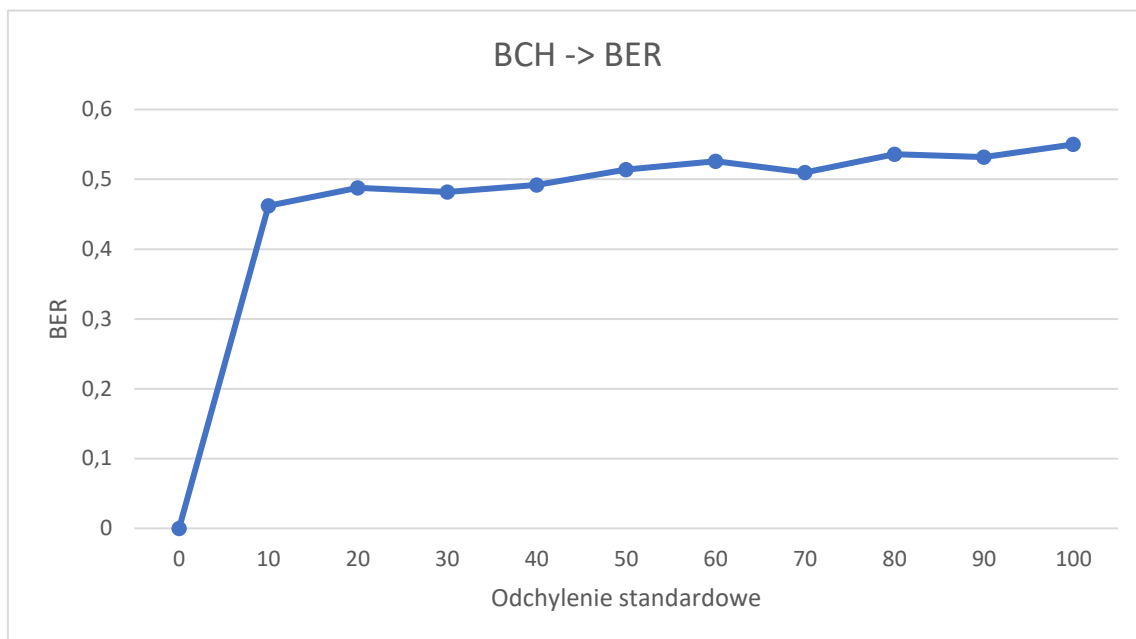


3.2 Wykresy zależności BER od odchylenia standardowego

a) dla $n = 15$, $k = 11$



b) dla $n = 31, k = 26$



c) dla $n = 63, k = 57$

