

Sprawozdanie z listy 3. - Technologie sieciowe

Jakub Bachanek

11 maja 2022

1 Ramkowanie

1.1 Koncepcja

W trakcie przesyłania bitów przez kanały komunikacji mogą wystąpić błędy, przez co sygnał będzie zawierał zmienione fragmenty. Ramkowanie z zastosowaniem metody rozpychania bitów polega na takim przetwarzaniu źródłowego strumienia bitów, aby odbiorca wiadomości mógł zidentyfikować i odrzucić zniekształcone części. Źródło jest dzielone na ustalonej długości elementy, dla których obliczana jest wartość pola kontrolnego CRC, po czym całość jest "rozpychana" w celu uniknięcia sekwencji o specjalnym znaczeniu. Wynik tej operacji jest umieszczany w ramce, która posiada unikalne znaczniki początku i końca. Program dekodujący rozpoznaje ramki, wykonuje procedurę odwrotną do "rozpychania", a następnie stwierdza poprawność na podstawie porównania odebranego kodu CRC z nowo obliczonym na podstawie danych w ramce.

1.2 Znaczniki

Jako znaczniki początku i końca ramki zastosujemy sekwencję bitów "01111110". Ich użycie pozwala, aby ramki mogły być różnych długości.

1.3 Pola kontrolne CRC

Do obliczenia CRC użyjemy biblioteki *zlib* w języku *Python*. Zwróci nam ona 32-bitową sumę kontrolną, którą użyjemy do testowania integralności danych.

1.4 Rozpychanie bitów

Rozpychanie bitów polega na transformacji ciągu bitów na taki, który nie zawiera zarezerwowanych sekwencji. W tym przypadku chcemy, żeby wiadomość połączona z CRC nie zawierała znaczników początku i końca. Realizujemy to poprzez dodanie bitu "0" po każdym pięciu kolejnych bitach "1". Po tej operacji

mamy pewność, że "01111110" będzie unikalne.

1.5 Programy

Stworzone programy pozwalają na zakodowanie strumienia bitów z pliku wejściowego opisaną metodą oraz jego dekodowanie. Pomocniczo jest możliwość utworzenia losowego pliku składającego się z "0" i "1" oraz dostępna jest opcja losowego zniekształcenia zakodowanego pliku.

2 CSMA/CD

2.1 Opis działania

CSMA/CD (Carrier Sense Multiple Access with Collision Detection) jest protokołem określającym sposób przekazywania pakietów w sieciach komputerowych typu LAN. Polega na nasłuchiwanie łącza przez wiele podłączonych urządzeń w celu badania jego stanu aktywności. Jeżeli stacja chce nadać sygnał i kanał nie jest zajęty, to rozpoczyna transmisję. W przeciwnym przypadku oczekuje do momentu kiedy linia się zwolni. Jeśli w trakcie wysyłania węzeł wykryje kolizję, to wtedy przerywa nadawanie i przystępuje do próby rozwiązania konfliktu.

2.2 Kolizja

Kolizja jest sytuacją, gdy dochodzi do nałożenia się dwóch lub więcej sygnałów, przez co informacje zostają utracone i węzły nie są w stanie ich poprawnie odczytać.

Warunkiem wykrycia kolizji jest wymóg, aby czas transmisji był co najmniej dwa razy dłuższy od czasu propagacji ($T_T > 2 \cdot T_P$). Wtedy mamy pewność, że zdarzenie zostanie zauważone dla wszystkich możliwych przypadków.

Po detekcji kolizji urządzenie zaczyna nadawać sygnał zakłócający (JAM), który poinformuje pozostałych o zaistnieniu takiej sytuacji. Następnie wszyscy, którzy transmitowali w momencie otrzymania sygnału JAM przystępują do losowania czasu oczekiwania przed ponowną próbą nadawania. Przedział czasowy z jakiego jest wybierana ta wartość jest uzależniony od liczby kolejnych nieudanych transmisji.

Poniżej przykładowy moment wystąpienia kolizji:

```
( _ )( _ )( _ )( _ )( _ )( _ )( _ )( b )( _ )( _ )( _ )( _ )( _ )( _ )( _ )( c )( c )( c )
( _ )( _ )( _ )( _ )( _ )( _ )( b )( b )( b )( _ )( _ )( _ )( _ )( _ )( _ )( c )( c )( c )( c )
( _ )( _ )( _ )( _ )( _ )( b )( b )( b )( b )( b )( b )( _ )( _ )( _ )( _ )( c )( c )( c )( c )( c )
( _ )( _ )( _ )( _ )( b )( b )( b )( b )( b )( b )( b )( b )( _ )( _ )( c )( c )( c )( c )( c )( c )
( _ )( _ )( _ )( b )( b )( b )( b )( b )( b )( b )( b )( b )( b )( c )( c )( c )( c )( c )( c )( c )
( _ )( _ )( b )( b )( b )( b )( b )( b )( b )( b )( b )( b )( bc )( bc )( c )( c )( c )( c )( c )( c )
( _ )( b )( b )( b )( b )( b )( b )( b )( b )( b )( b )( bc )( bc )( bc )( bc )( c )( c )( c )( c )( c )
( b )( b )( b )( b )( b )( b )( b )( b )( b )( b )( bc )( bc )( bc )( bc )( bc )( bc )( c )( c )( c )( c )
( b )( b )( b )( b )( b )( b )( b )( b )( bc )( bc )( bc )( bc )( bc )( bc )( bc )( bc )( c )( c )( c )
( b )( b )( b )( b )( b )( b )( bc )( bc )( bc )( bc )( bc )( bc )( bc )( bc )( bc )( bc )( c )( c )
( b )( b )( b )( b )( b )( bc )( J )( bc )( bc )( bc )( bc )( bc )( bc )( bc )( bc )( bc )( bc )( c )
( b )( b )( b )( b )( bc )( J )( _ )( _ )( _ )( J )( bc )( bc )( bc )( bc )( bc )( bc )( bc )( bc )( bc )
( b )( b )( b )( bc )( J )( _ )( _ )( _ )( _ )( _ )( J )( bc )( bc )( bc )( bc )( bc )( bc )( bc )( bc )
( b )( b )( bc )( J )( _ )( _ )( _ )( _ )( _ )( _ )( J )( bc )( bc )( bc )( bc )( bc )( bc )( bc )( bc )
( b )( bc )( J )( _ )( _ )( _ )( _ )( _ )( _ )( _ )( _ )( J )( bc )( bc )( bc )( bc )( bc )( bc )( bc )
```

2.3 Program

Zadanie zostało zrealizowane w języku *Python* w paradygmacie obiektowym. Ręcznie dobieramy parametry uruchamiania takie jak długość kanału, liczba urządzeń oraz prawdopodobieństwo nadawania. Rozmieszczenie przypisywane jest losowo, a emitowane sygnały przemieszczają się w określonym kierunku w tablicy, która odzwierciedla łącze. Węzły i ich wiadomości oznaczane są kolejnymi literami alfabetu.

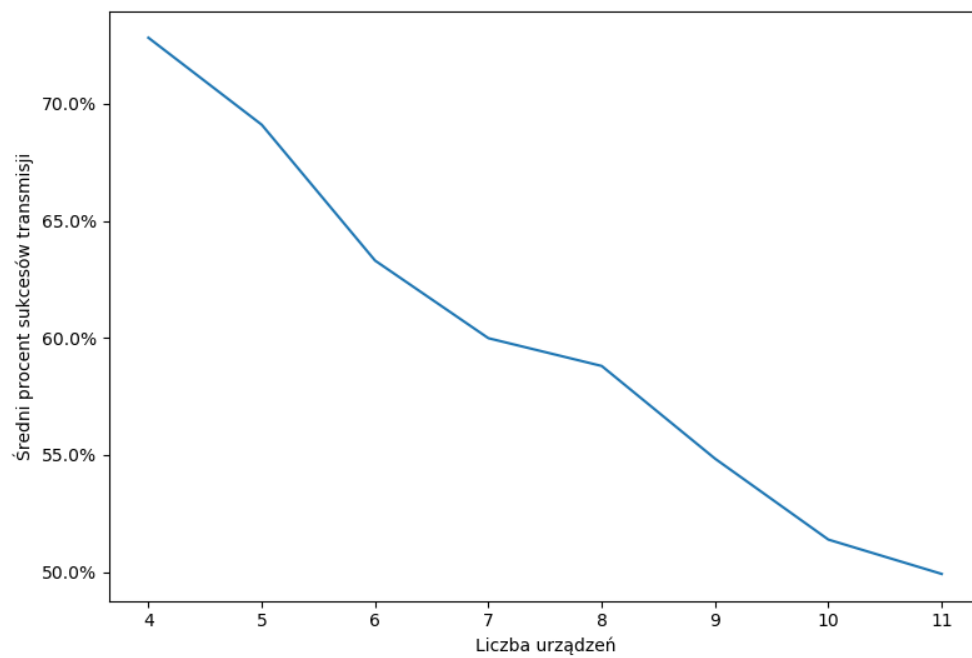
2.4 Symulacja

Do zadania przeprowadzona zostaje symulacja. W każdej iteracji, z określonym prawdopodobieństwem, urządzenia próbują nadać wiadomość do losowo wybranej stacji. Liczba prób i sukcesów jest zliczana i przedstawiana w podsumowaniu. Wizualizacja przebiegu działania programu jest wyświetlana w terminalu.

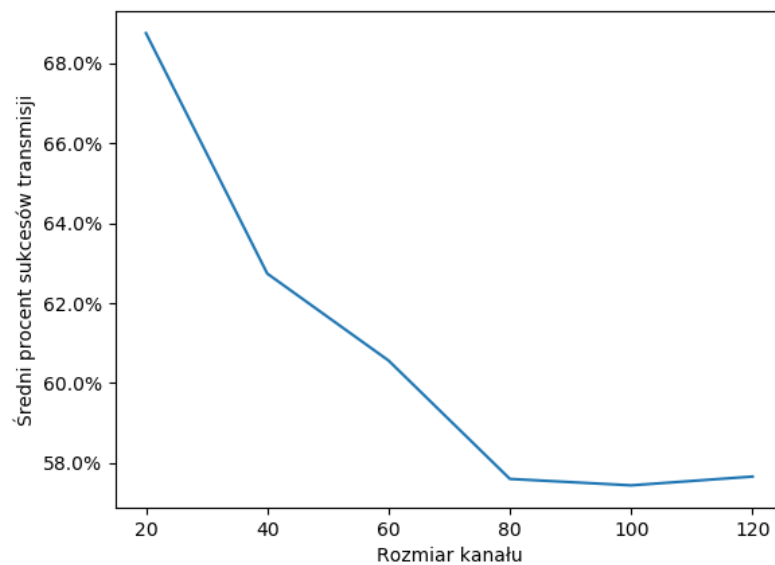
2.5 Testy

Poniższe wykresy przedstawiają otrzymane wyniki przy zmianach wskazanych parametrów. Sukcesem nazywamy zdarzenie, gdy urządzenie nada całą długość wiadomości i w tym czasie nie wykryje obcego sygnału.

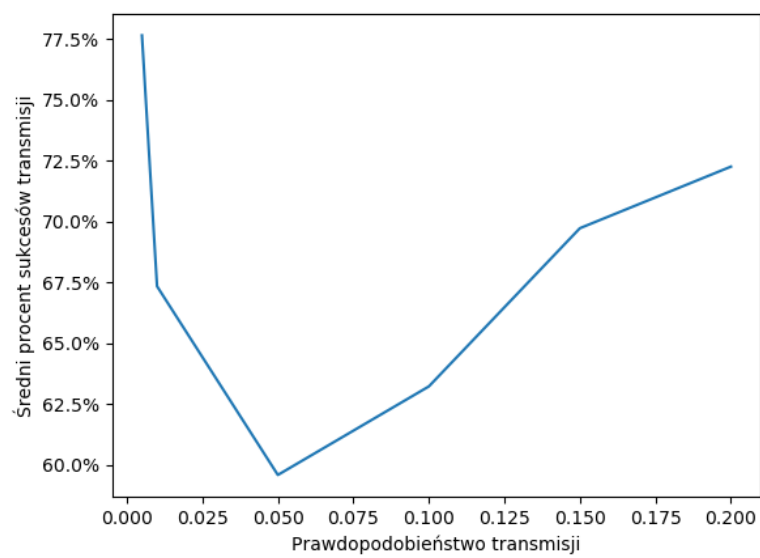
2.5.1 Wpływ liczby urządzeń na liczbę sukcesów transmisji



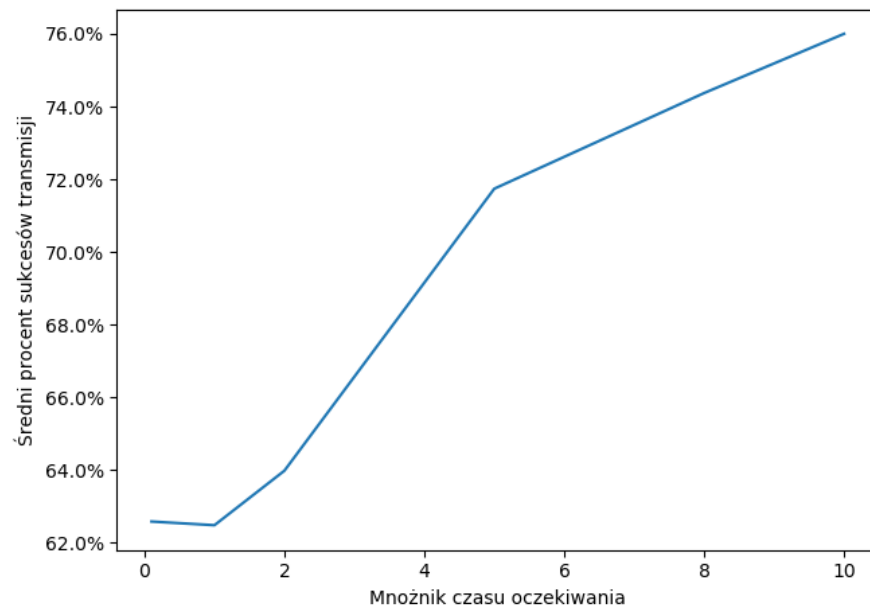
2.5.2 Wpływ długości łącza na liczbę sukcesów transmisji



2.5.3 Wpływ prawdopodobieństwa transmisji na liczbę sukcesów transmisji



2.5.4 Wpływ długości czasu wstrzymania po kolizji na liczbę sukcesów transmisji



2.6 Wnioski

Na wyniki eksperymentów ma wpływ dużo różnych parametrów. Zmiana każdego z nich skutkuje widoczną różnicą w działaniu symulacji. Jeżeli wartości są nieodpowiednio dobrane, korzystanie z sieci staje się bardzo nieefektywne. Ponadto istotne jest samo rozmieszczenie urządzeń względem siebie. Różne konfiguracje odległości między nimi mogą dawać znacznie odmienne rezultaty. Ze względu na zapotrzebowanie na lepszą wydajność, połączenia *half-duplex* i co za tym idzie *CSMA/CD* nie są już obecnie powszechnie wykorzystywane w sieciach Ethernet.