
MOPS - PROJEKT

Temat: Węzeł modelowany jako kolejka FIFO o rozmiarze K wyrażonym w liczbie pakietów. Badane działanie systemu dla $n=1,2,3,4$ źródeł typu ON/OFF, gdzie długość stanów ON i OFF opisane są rozkładem wykładniczym. Stały rozmiar pakietów.

Zespół:

Kamil Chrościcki 300502

Filip Smejda 300503

Prowadzący:

dr. hab. inż. Halina Tarasiuk

Spis treści

Zespół:	1
Prowadzący:	1
1. Zagadnienia projektowe	2
1.1. Cel projektu:	2
1.2. Założenia projektowe:	2
2. Przygotowanie modelu symulacyjnego (raport częściowy)	2
3. Raport końcowy – opis wyników badań	9
4. Podsumowanie	17

1. Zagadnienia projektowe

1.1. Cel projektu:

Zamodelować jeden węzeł. Jest on opisany jako system kolejkowy. Do węzła może być podłączonych (n) źródeł ruchu, które generują pakiety. Źródła ruchu opisane są typem ON/OFF, gdzie długość stanów ON i OFF opisane są rozkładem wykładniczym. Stały rozmiar pakietów.

1.2. Założenia projektowe:

- Metryki pomiarowe (opisano niżej): średnia liczba pakietów w kolejce, średni czas oczekiwania w kolejce, średnie obciążenie serwera, poziom strat pakietów.
- Kolejka jest ograniczona maksymalną pojemnością o wartości skończonej K.
- Implementacja symulatora w języku programowania: C#.
- System obsługi pakietów w węźle – FIFO (First In First Out) – pierwszy przybył, pierwszy obsłużony. Zamodelowany jako system z pojedynczą kolejką i jednym serwerem obsługi.
- Symulator powinien pozwalać na przeprowadzenie symulacji dla węzła z założoną liczbą źródeł ruchu. W naszym modelu symulacyjnym występuje n (od 1 do 4) nadawców pakietów w postaci źródeł pakietów, które mogą przyjmować dwa stany ON i OFF.
- Czas trwania stanu źródła opisany jest w postaci rozkładu wykładniczego prawdopodobieństwa, czyli rozkładu zmiennej losowej opisującej sytuację, w której obiekt może przyjmować dwa wyżej wymienione stany.
- Pakiety w stanie ON napływają do węzła w zależności od wartości ustawionej na początku symulacji.
- Stały rozmiar pakietów określony na początku symulacji.

2. Przygotowanie modelu symulacyjnego (raport częściowy)

Model symulacyjny składa się z następujących modułów:

- **Stan systemu**

Czyli zmienne opisujące stany w poszczególnych chwilach czasowych:

- Stan serwera obsługi: wolny lub zajęty
- Liczba klientów oczekujących w kolejce na obsługę (jeżeli w ogóle ktoś oczekuje)
- Czas przybycia każdej osoby, która oczekuje na obsługę

- **Zegar symulacji**

Zmienna opisująca bieżący czas symulacji. Ustawiana na wartość 0 podczas inicjalizacji. Zwiększany jest kolejnym wystąpieniem zdarzenia w systemie.

- **Lista zdarzeń**

Lista zawierająca czas wystąpienia kolejnego zdarzenia w systemie dla każdego z typów zdarzeń. W naszym systemie: A_i – przybycie pakietu z i -tego źródła, gdzie $i=1,2,3,4$; D – zakończenie obsługi pakietu przez serwer.

- **Algorytm inicjalizujący**

Algorytm inicjalizujący system w momencie jego uruchomienia. Ustawia dane początkowe np. czas systemu = 0, status serwera = 0 (wolny), liczba pakietów w kolejce = 0, czas ostatniego zdarzenia = 0, n źródeł systemu w stanie ON, wyzeruje licznik statystyk. Algorytm ten również odpowiada za rozsynchronizowanie źródeł na początku symulacji, tak aby pierwsze pakiety z różnych źródeł nie były wysyłane w tym samym momencie.

- **Algorytm czasowy**

Określa kolejne zdarzenie z listy zdarzeń i zwiększa zegar symulacji na czas wystąpienia tego zdarzenia.

- **Algorytm zdarzeniowy**

Uaktualnia stan systemu w momencie wystąpienia kolejnego zdarzenia.

- **Algorytm bibliotek**

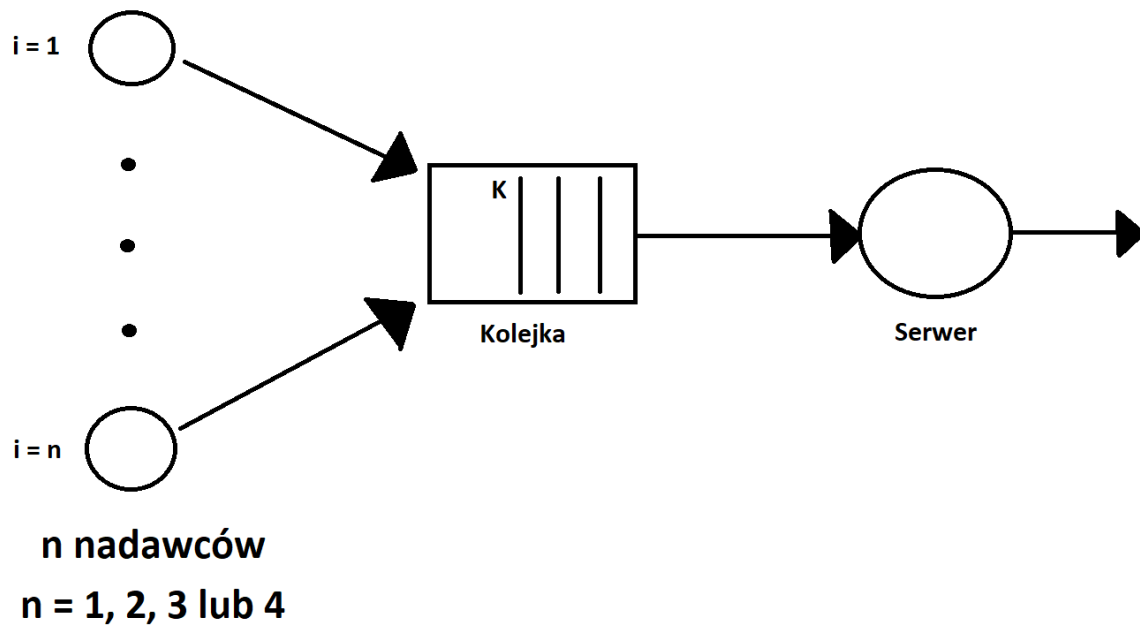
Generuje losowe obserwacje systemu w oparciu o pewien rozkład prawdopodobieństwa. W przypadku naszego systemu poddany obserwacjom będzie wpływ zmiany stanu źródła ON/OFF opisany rozkładem wykładniczym.

- **Generator raportów**

Oblicza charakterystyki na podstawie zmierzonych wartości i generuje raport po zakończeniu symulacji.

- **Program główny**

Architektura systemu



Rysunek 1. Architektura systemu

Metryki pomiarowe

- średnia liczba pakietów w kolejce

$$\hat{q}(n) = \frac{\sum_{i=0}^{\infty} iT_i}{T(n)} \quad \sum_{i=0}^{\infty} iT_i = \int_0^{T(n)} Q(t) dt \quad \hat{q}(n) = \frac{\int_0^{T(n)} Q(t) dt}{T(n)}$$

$q(n)$ – liczba pakietów w kolejce oszacowana w czasie ciągłym; zmienna losowa ciągła

$Q(t)$ – liczba pakietów w kolejce w czasie t

$T(n)$ – czas wymagany do obserwacji n opóźnień

T_i – czas podczas symulacji, kiedy kolejka ma długość i

- średni czas oczekiwania w kolejce

$$\hat{d}(n) = \frac{\sum_{i=1}^n D_i}{n}$$

$d(n)$ – średniego czasu oczekiwania w kolejce przez n klientów (zmienna losowa dyskretna)

D_i – czas oczekiwania i -tego klienta

n – liczba czasów oczekiwania (opóźnień)

- średnie obciążenie serwera

$$\hat{u}(n) = \frac{\int_0^{T(n)} B(t) dt}{T(n)}$$

$u(n)$ - wykorzystanie serwera obsługi, zmienna losowa ciągła (stosunek czasu zajętości serwera do czasu symulacji)

$B(t)$ – funkcja zajętości, $B(t)=1$, kiedy serwer jest zajęty w czasie t , $B(t)=0$, kiedy serwer jest wolny w czasie t

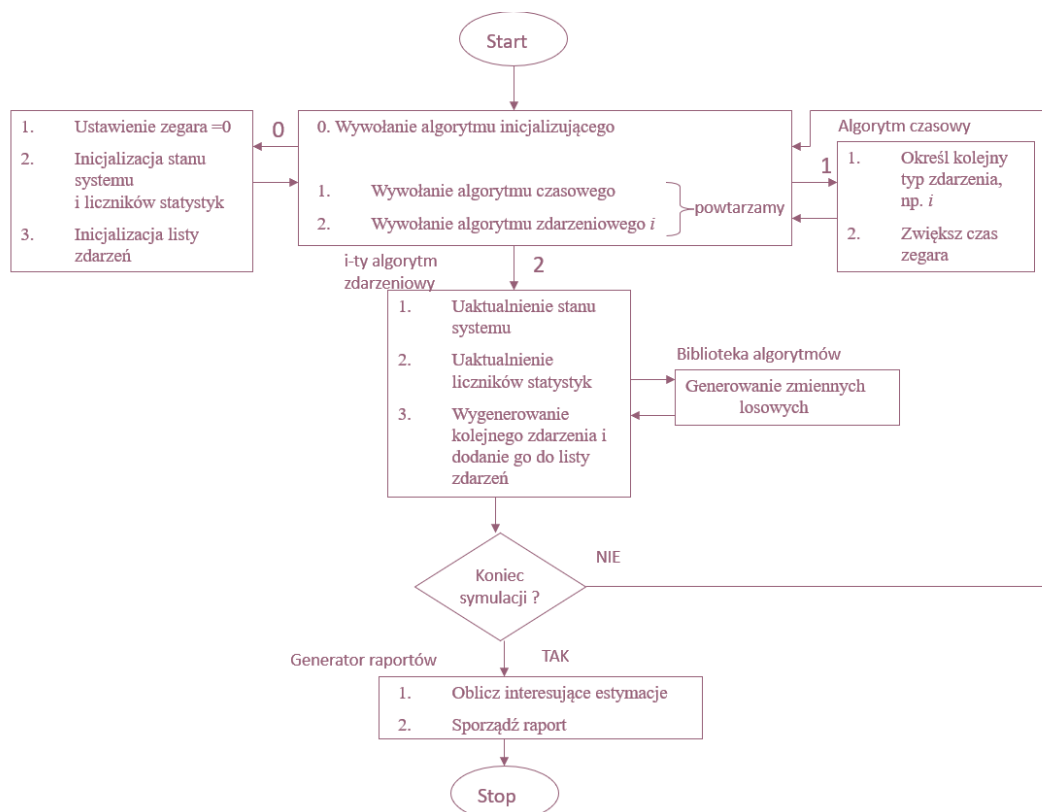
- poziom strat pakietów

$$s = \frac{k}{n} * 100\%$$

k – liczba straconych pakietów

n – liczba wysłanych pakietów

Ogólny sposób działania systemu



Rysunek 2. Sposób działania systemu - Wykorzystano slajd wykładowy autorstwa dr. hab. inż. Haliny Tarasiuk udostępniony na rzecz przedmiotu MOPS_Z20

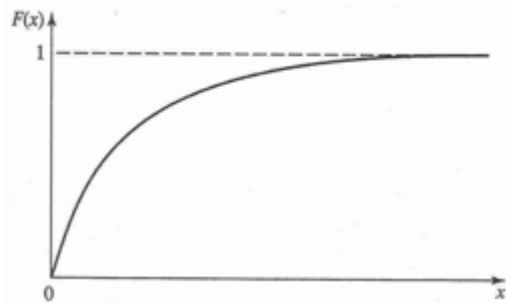
Parametry wejściowe, czyli czas trwania stanu ON/OFF:

Symulacja, która ma jakiegokolwiek losowe elementy musi angażować pewne próbki lub generować liczby losowe z rozkładów prawdopodobieństwa.

Czas stanu ON/OFF opisany jest rozkładem wykładniczym. Długości obu stanów będą zmienną losową mającą rozkład wykładniczy z wartością średnią β .

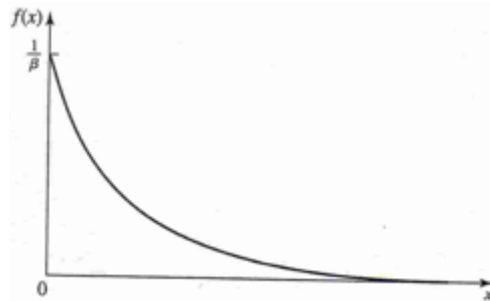
Dystrybuanta rozkładu dana jest funkcją:

$$F(x) = \int_0^x \frac{1}{\beta} e^{-t/\beta} dt = 1 - e^{-x/\beta}$$



Funkcja gęstości:

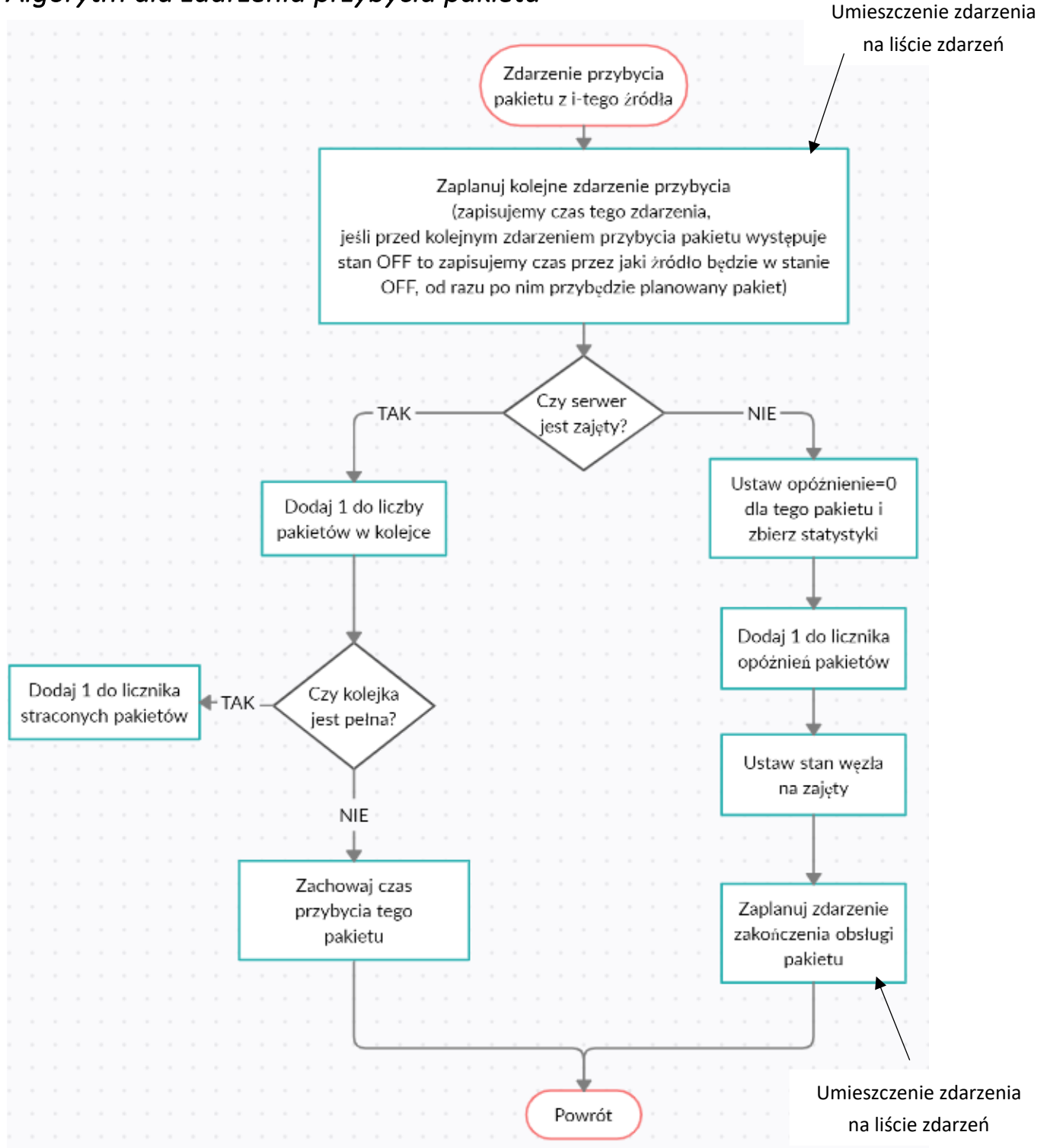
$$f(x) = \frac{1}{\beta} e^{-x/\beta} \quad \text{dla } x \geq 0$$



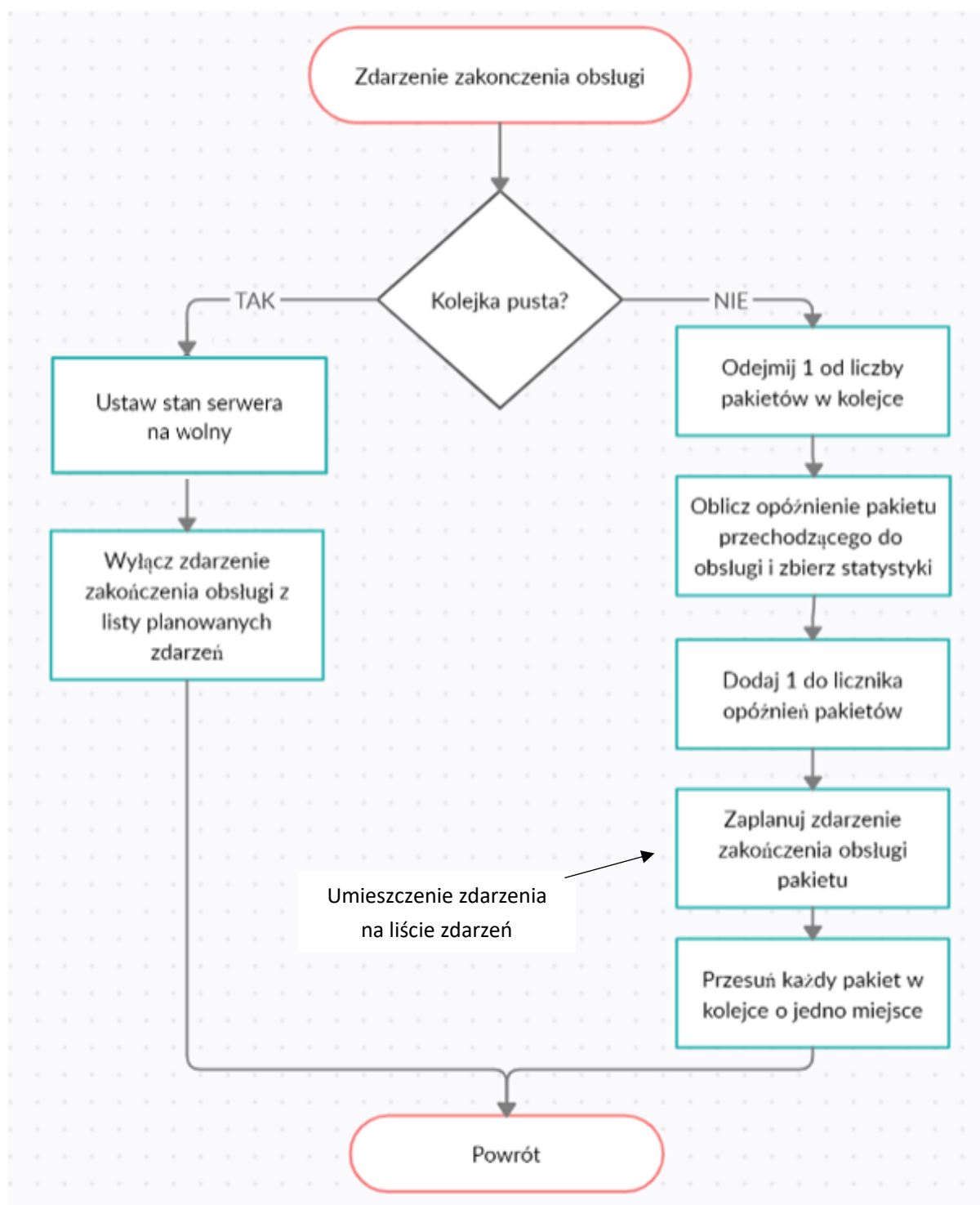
W funkcji odwrotnej przyjmujemy $u=F(x)$ i w ten sposób znajdujemy x , czyli długość stanu:

$$F^{-1}(u) = -\beta \ln(1-u)$$

Algorytm dla zdarzenia przybycia pakietu



Rysunek 3. Algorytm dla zdarzenia przybycia pakietu

Algorytm dla zdarzenia zakończenia obsługi pakietu

Rysunek 4. Algorytm dla zdarzenia zakończenia obsługi pakietu

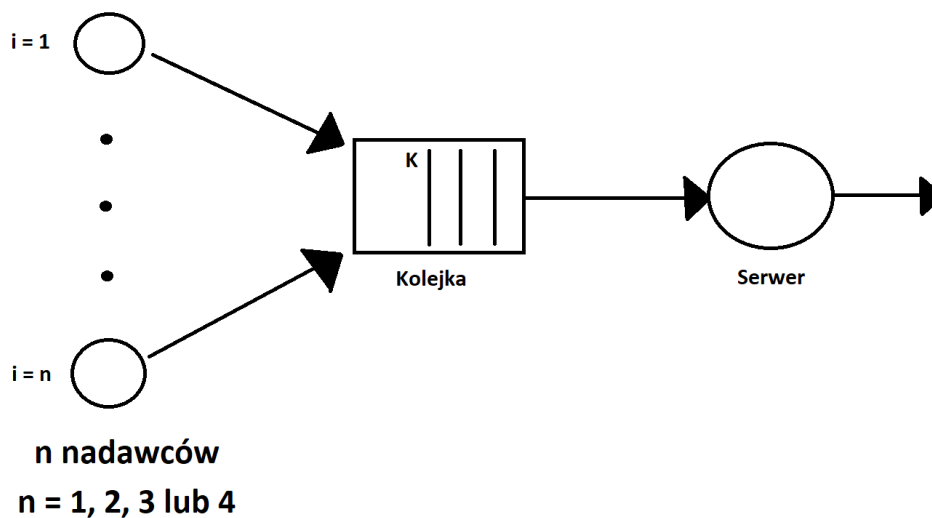
3. *Raport końcowy – opis wyników badań*

- *Cel*

1. Zbadanie wpływu liczby źródeł na zmierzone metryki pomiarowe: średnia liczba pakietów w kolejce, średni czas oczekiwania w kolejce, średnie obciążenie serwera, poziom strat pakietów.
2. Zbadanie wpływu długości kolejki na zmierzone metryki pomiarowe: średnia liczba pakietów w kolejce, średni czas oczekiwania w kolejce, średnie obciążenie serwera, poziom strat pakietów.

- *Topologia testowa*

Topologia systemu wygląda jak na rysunku 5. Symulacje przeprowadzane zostały dla poszczególnych liczby źródeł równej kolejno $n = 1, 2, 3, 4$. Następnie dla przypadku $n = 4$ zwiększana zostanie kolejka w celu dostrzeżenia zależności wpływu jej rozmiaru na wartości badanych metryk.



Rysunek 5. Topologia systemu

- *Wyniki numeryczne (zestawienie tabelaryczne/wykresy)*

Badanie wpływu ilości źródeł na metryki:

Parametry wejściowe symulacji:

Czas symulacji = 20 000s

Odstęp między pakietami = 0.0125s → 80 pakietów/s

Czas obsługi pakietu = 0.01s → 100 pakietów/s

Beta ON/OFF = 10s

Rozmiar kolejki = 4

Obliczenie maksymalnego możliwego czasu oczekiwania pakietu w kolejce:

$$\text{Rozmiar kolejki} * \text{czas obsługi pakietu} = 4 * 0.01s = \mathbf{0.04s}$$

a) Liczba źródeł $n = 1$

```
***** RESULTS *****
Total time: 20000
Number of packets sent: 812920
Mean number of packets in queue: 5,232517842159723E-07
Mean packet delay: 1,2873238E-08
Mean server load: 0,406459999938473
Packet loss level: 0%
```

b) Liczba źródeł $n = 2$

```
***** RESULTS *****
Total time: 20000
Number of packets sent: 1626190
Mean number of packets in queue: 0,9238755003533939
Mean packet delay: 0,013959280882067
Mean server load: 0,6618331079082702
Packet loss level: 18,60323824399363%
```

c) Liczba źródeł $n = 3$

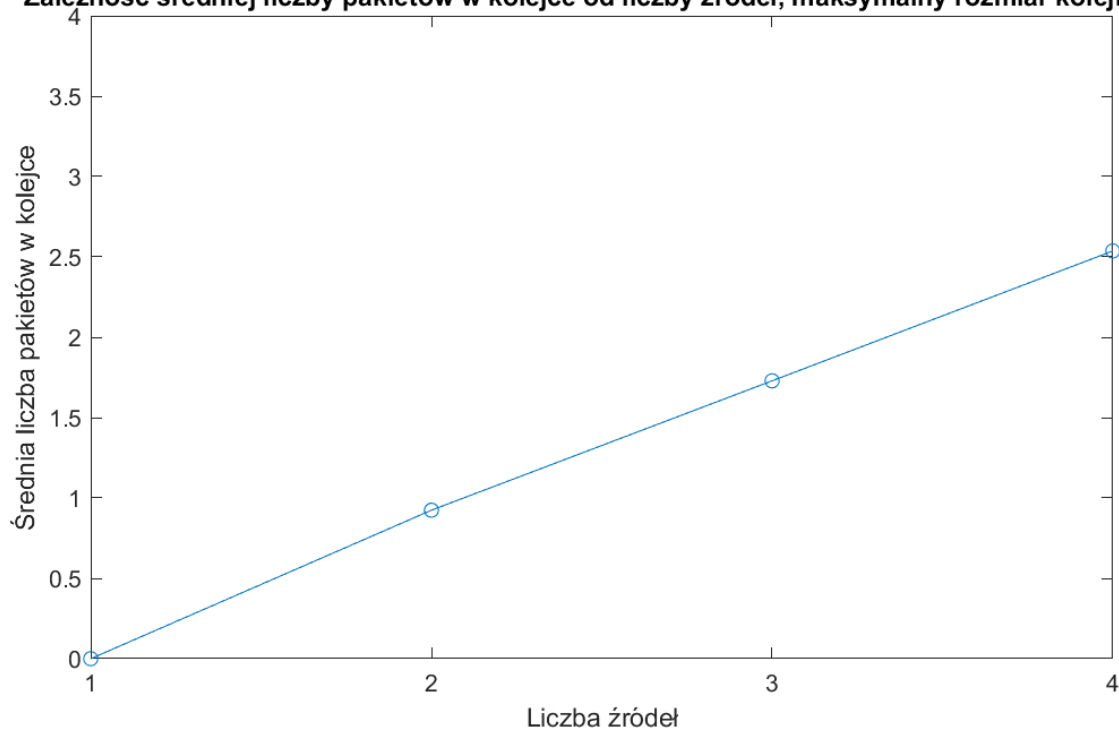
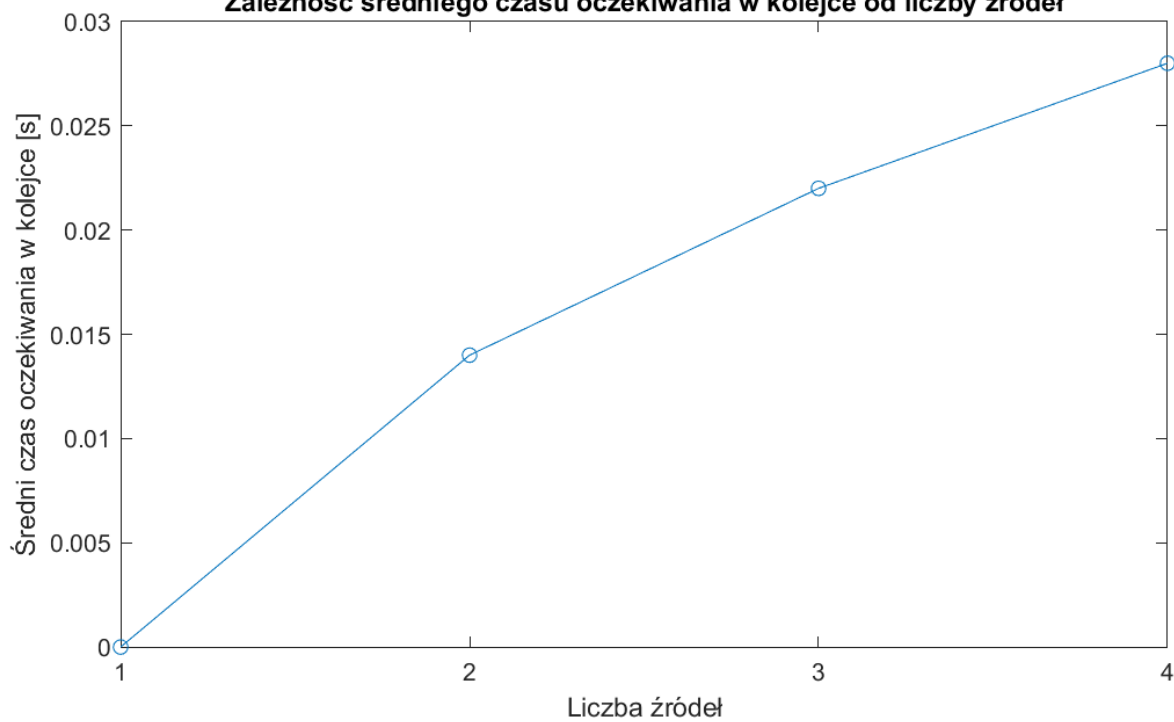
```
***** RESULTS *****
Total time: 20000
Number of packets sent: 2317266
Mean number of packets in queue: 1,7286509602203042
Mean packet delay: 0,021993251881646
Mean server load: 0,785991603705311
Packet loss level: 32,1621686936243%
```

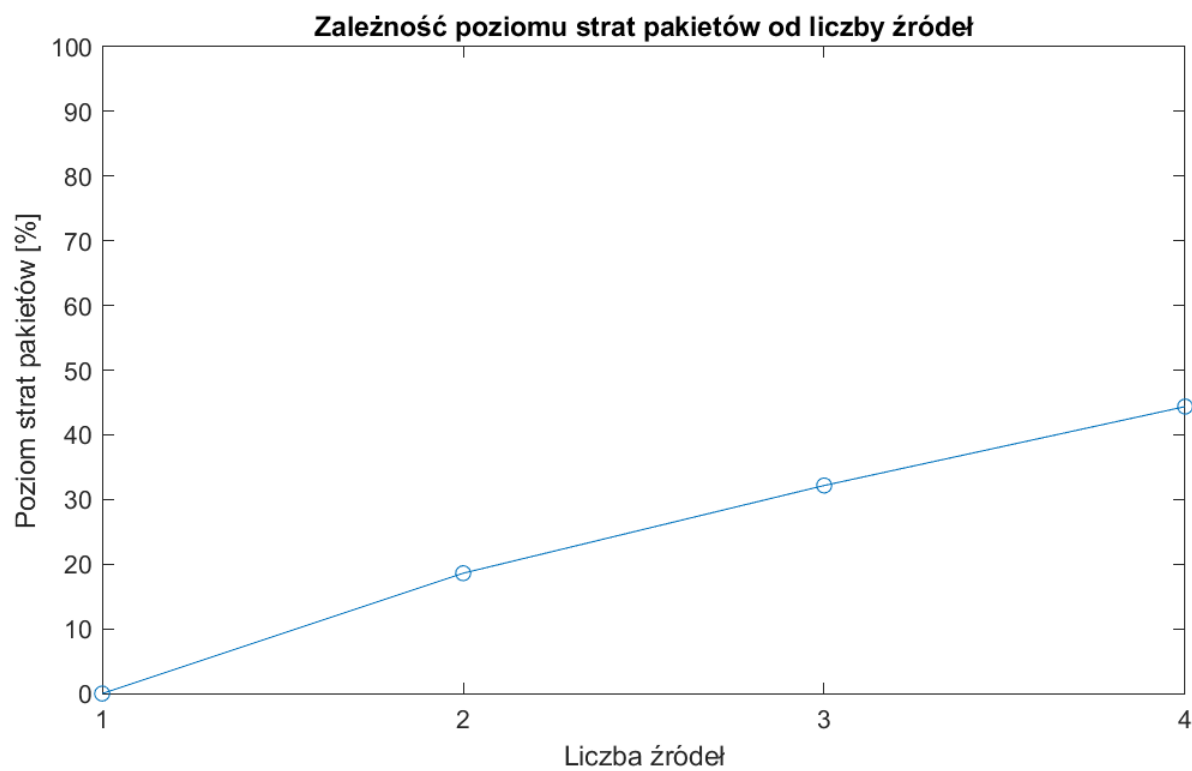
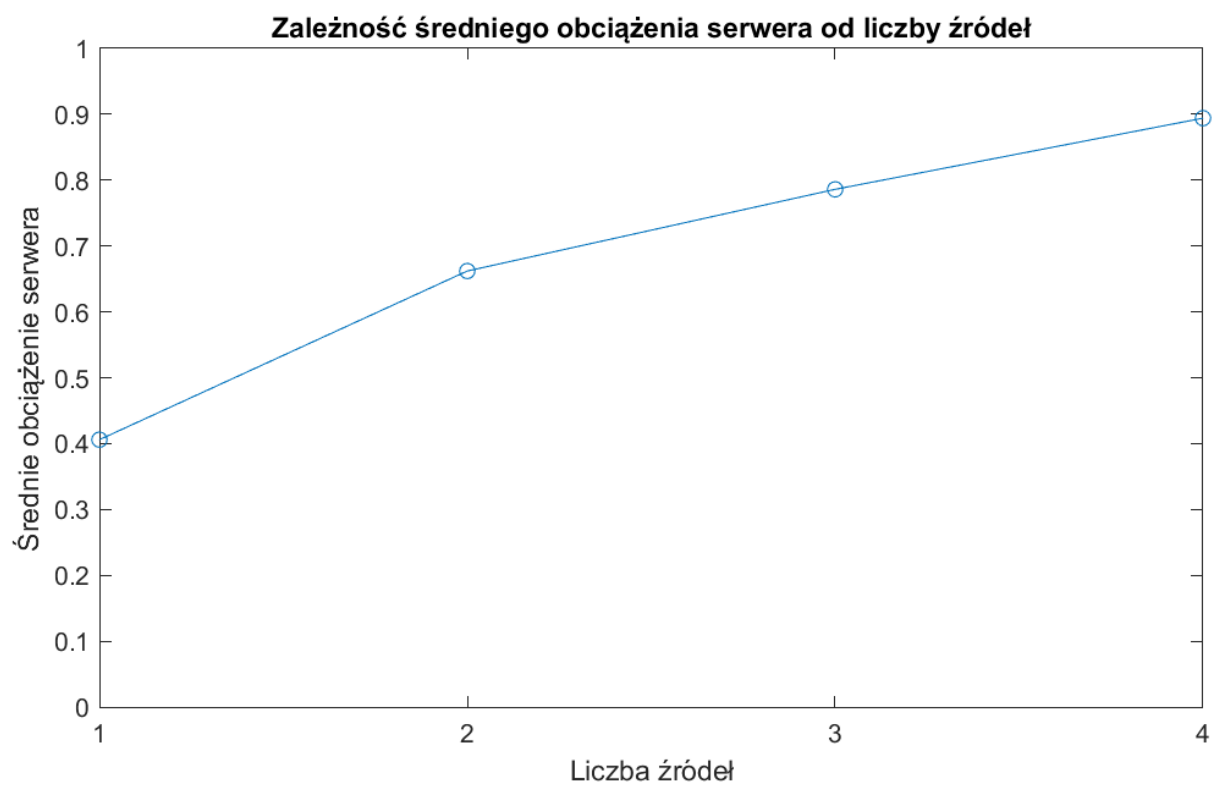
d) Liczba źródeł $n = 4$

```
***** RESULTS *****
Total time: 20000
Number of packets sent: 3212616
Mean number of packets in queue: 2,537640828362795
Mean packet delay: 0,02839456958616
Mean server load: 0,8937055561581055
Packet loss level: 44,36275608413828%
```

Tabela 1. zawierająca uzyskane wartości metryk w zależności od liczby źródeł.

Liczba źródeł	Średnia liczba pakietów w kolejce	Średni czas oczekiwania w kolejce [s]	Średnie obciążenie serwera	Poziom strat pakietów [%]
$n = 1$	0	0	0.406	0
$n = 2$	0.924	0.014	0.662	18.603
$n = 3$	1.729	0.022	0.786	32.162
$n = 4$	2.537	0.028	0.894	44.363

Zależność średniej liczby pakietów w kolejce od liczby źródeł, maksymalny rozmiar kolejki = 4**Zależność średniego czasu oczekiwania w kolejce od liczby źródeł**



- *Wnioski*

Na podstawie otrzymanych wyników oraz uzyskanych charakterystyk obserwujemy wzrost wartości poszczególnych metryk wraz ze wzrostem liczby źródeł. Parametry wejściowe zostały dobrane w taki sposób, aby zademonstrować poprawne działanie systemu w różnych przypadkach. Występowanie zaledwie jednego źródła przy szybszej obsłudze pakietu niż jego generowaniu skutkuje zerowymi wartościami metryk średniej liczby pakietów w kolejce, średniego czasu oczekiwania w kolejce oraz poziomu strat pakietów. Wynika to faktu, iż pakiet który jest generowany nie będzie oczekiwać, a więc zostanie obsłużony bez kolejki. Sytuacja ta zaczyna się zmieniać w momencie dokładania kolejnych źródeł. Wszystkie metryki zaczynają znacząco wzrastać. Opisywaną sytuację obrazują wykreślone powyżej krzywe. Pakiety zaczynają napływać do systemu zależnie od czasu trwania stanu ON/OFF źródła i mogą napotkać zajęty serwer. Wynika to z potencjalnego obsługiwania przebywającego tam innego pakietu. Ze względu na taką sytuację znacznie wzrosło wykorzystanie kolejki w węźle przez przychodzące pakiety. Średnia liczba pakietów w kolejce zaczyna dążyć do maksymalnej długości kolejki (4 w tym przypadku). Porównując maksymalny możliwy czas oczekiwania pakietu w kolejce z wynikami otrzymanymi podczas symulacji jesteśmy w stanie stwierdzić wiarygodność badanego systemu. Żadna z otrzymanych średnich wartości czasu pakietu w kolejce nie była większa od maksymalnej, którą otrzymaliśmy na podstawie obliczeń.

Badanie wpływu długości kolejki na metryki:

Parametry wejściowe symulacji:

Czas symulacji = 20 000s

Odstęp między pakietami = 0.2s → 5 pakietów/s

Czas obsługi pakietu = 0.08s → 12.5 pakietów/s

Beta ON/OFF = 10s

Liczba źródeł = 4

a) Rozmiar kolejki = 2

```
***** RESULTS *****
Total time: 20000
Number of packets sent: 205049
Mean number of packets in queue: 0,5061666605929407
Mean packet delay: 0,055424692796068
Mean server load: 0,7305987994414176
Packet loss level: 10,92421811371916%
```

b) Rozmiar kolejki = 5

```
***** RESULTS *****
Total time: 20000
Number of packets sent: 205708
Mean number of packets in queue: 1,4915937365743888
Mean packet delay: 0,15987320855469
Mean server load: 0,7463902188186693
Packet loss level: 9,290353316351332%
```

c) Rozmiar kolejki = 10

```
***** RESULTS *****
Total time: 20000
Number of packets sent: 201819
Mean number of packets in queue: 2,9073931193213665
Mean packet delay: 0,309874033500681
Mean server load: 0,7506000000023981
Packet loss level: 7,020647213592378%
```

d) Rozmiar kolejki = 50

```
***** RESULTS *****
Total time: 20000
Number of packets sent: 200909
Mean number of packets in queue: 12,415557488246538
Mean packet delay: 1,269802436653537
Mean server load: 0,7822043993641626
Packet loss level: 2,6668790347868936%
```

e) Rozmiar kolejki = 100

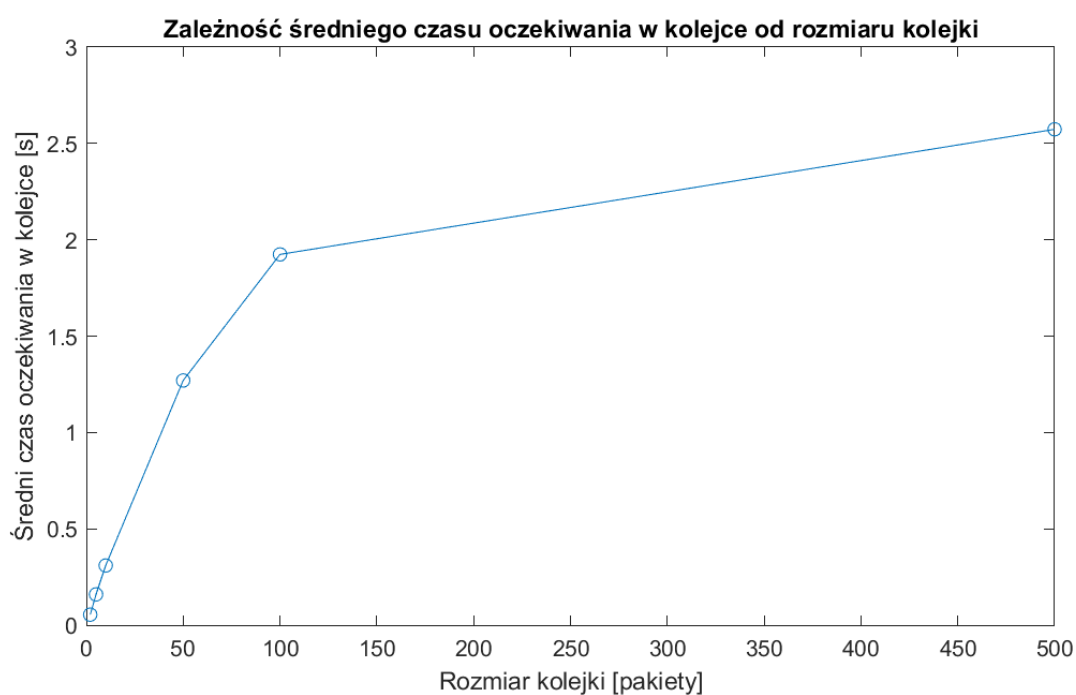
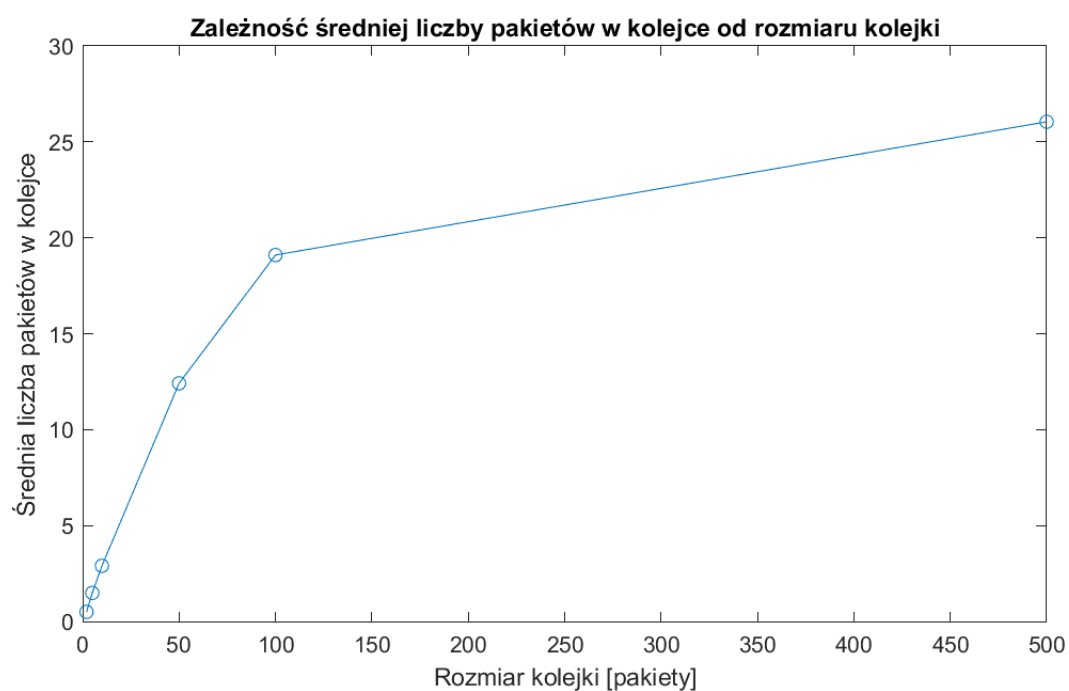
```
***** RESULTS *****
Total time: 20000
Number of packets sent: 200669
Mean number of packets in queue: 19,101934978700765
Mean packet delay: 1,923895856494706
Mean server load: 0,7942281990959101
Packet loss level: 1,0524794562189477%
```

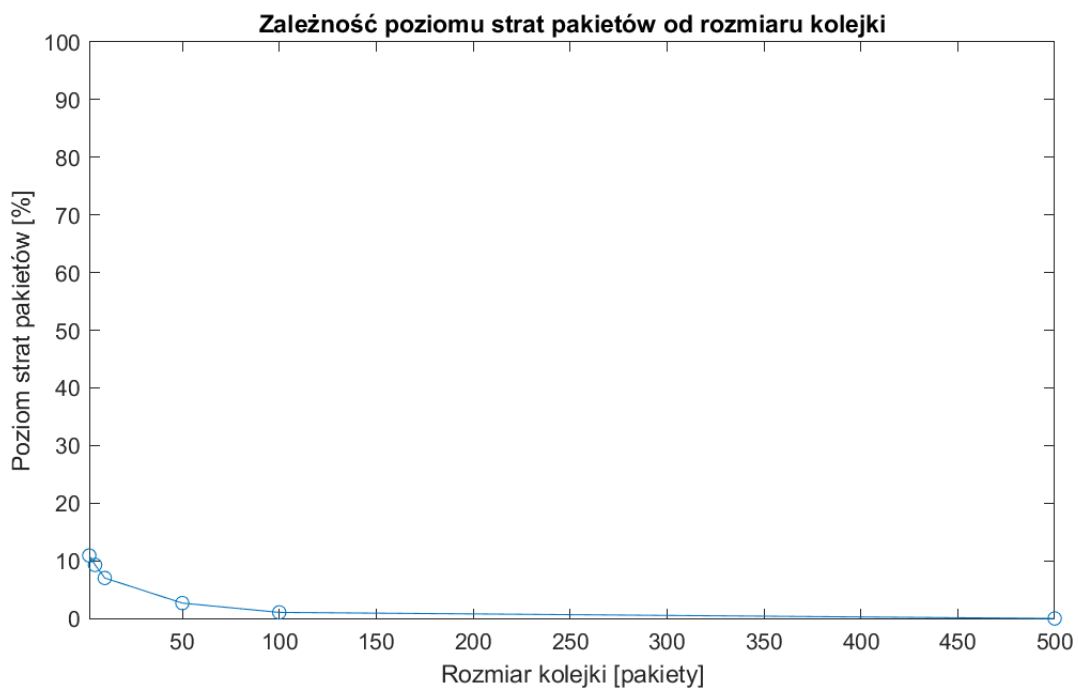
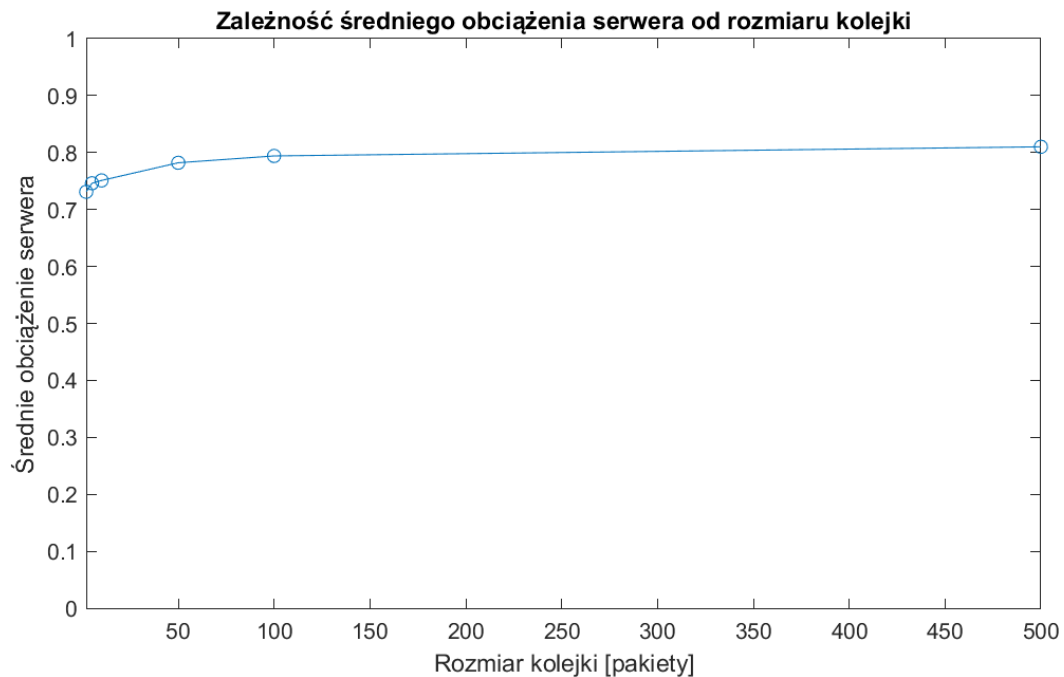
f) Rozmiar kolejki = 500

```
***** RESULTS *****
Total time: 20000
Number of packets sent: 202489
Mean number of packets in queue: 26,044872694351206
Mean packet delay: 2,572485413015894
Mean server load: 0,8099542103682388
Packet loss level: 0,0004938539871301651%
```

Tabela 2. zawierająca uzyskane wartości metryk w zależności od rozmiaru kolejki.

Rozmiar kolejki	Średnia liczba pakietów w kolejce	Średni czas oczekiwania w kolejce [s]	Średnie obciążenie serwera	Poziom strat pakietów [%]
2	0.506	0.055	0.731	10.924
5	1.492	0.160	0.746	9.290
10	2.907	0.310	0.751	7.021
50	12.416	1.270	0.782	2.667
100	19.102	1.924	0.794	1.053
500	26.045	2.573	0.810	0



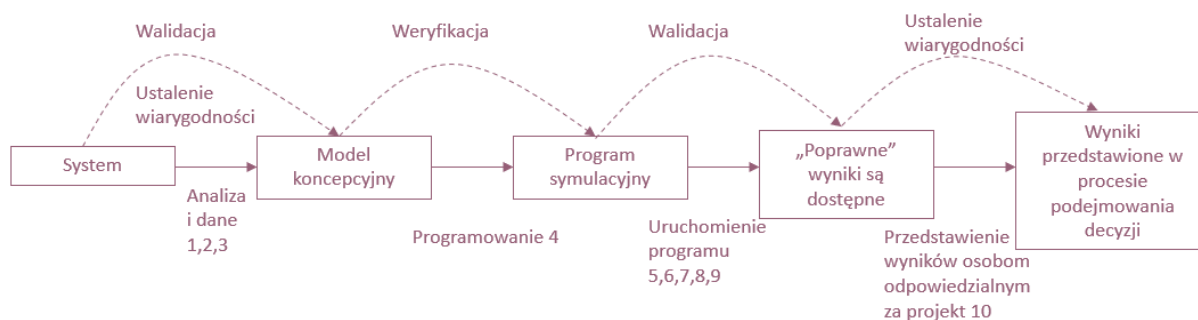


• Wnioski

Badanie zależności poszczególnych metryk od rozmiaru kolejki pozwala na dostrzeżenie kolejnych właściwości analizowanego systemu. Możemy zaobserwować, iż poziom strat pakietów znacznie maleje wraz ze wzrostem rozmiaru kolejki i dąży do 0%. Wykonanie tego doświadczenia odbyło się przy uruchomieniu czterech źródeł nadawczych. Rozmiar kolejki ma tutaj znaczącą rolę, gdyż umożliwia zachowanie większej liczby pakietów w celu ich obsłużenia, a nie konieczności ich odrzucenia. Ważne jest, aby ta metryka osiągała małe wartości. W przeciwnym przypadku tracona jest znaczna część wysyłanych pakietów prowadząca do

straty przesyłanych danych. Naturalnym jest natomiast, że zwiększanie rozmiaru kolejki sprawia, że przy tak zadanych parametrach inne metryki będą wzrastać. Poszczególnych pakietów będzie więcej i będą musiały one dłużej czekać, aby zostać obsłużone. Zajętość serwera praktycznie się nie zmienia.

4. Podsumowanie



Rysunek 6. Etapy realizacji projektu - wykorzystano slajd wykładowy autorstwa dr. hab. inż. Haliny Tarasiuk udostępniony na rzecz przedmiotu MOPS_Z20.

Powyższy schemat przedstawia poszczególne etapy realizacji wykonanego modelu symulacyjnego. Kluczowym momentem w trakcie jego realizacji było przede wszystkim stworzenie w pełni poprawnego modelu koncepcyjnego. Należało również przeanalizować i zrozumieć wszystkie założenia związane z modelowanym węzłem. Zaznajomienie się z tematem poprzez stworzenie dobrej koncepcji symulatora znacznie ułatwiło dalszą pracę. Przeprowadzona została walidacja modelu teoretycznego i ustalono jego wiarygodność.

Zaprogramowanie modelu koncepcyjnego spowodowało utworzenie faktycznego programu symulacyjnego. Została przeprowadzona jego weryfikacja.

Wykorzystując utworzony program rozpoczęto etap zbierania danych i obserwacji otrzymywanych wyników. Na ich podstawie przeprowadzono walidację. Dokładna analiza i zbieranie danych wyliczanych z poszczególnych metryk sprawiło, że uznano otrzymywane wyniki, w zakresie tego programu, za poprawne. Przedstawienie tych wartości prowadzącemu projekt oraz ich zatwierdzenie oznaczać będzie, że zostanie ustalona wiarygodność badanego modelu.

Za pomocą symulacji umożliwiało było zbadanie i analizowanie właściwości węzła FIFO z kolejką o rozmiarze K , wyrażonym w liczbie pakietów. Stan długości stanów ON i OFF źródeł generujących pakiety opisane są rozkładem wykładniczym. Wykazano zależność wielkości poszczególnych metryk od zwiększającej się ilości opisanych wyżej źródeł. Dodatkowo zwróciliśmy szczególną uwagę na zmianę poziomu strat pakietów w funkcji rozmiaru kolejki.