

wt. 9¹⁵-11⁰⁰
29.03.2020

Technologie sieciowe

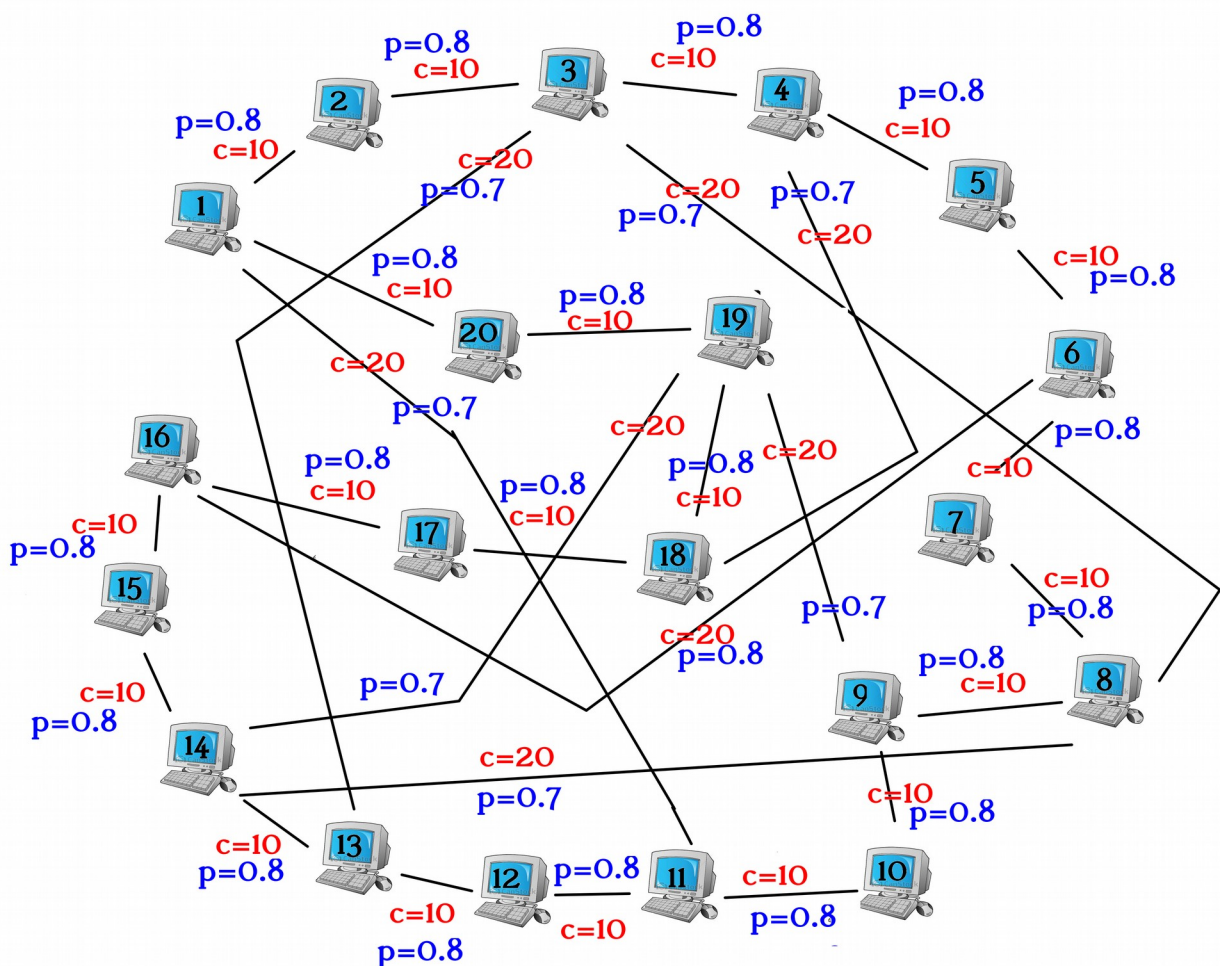
Lista II

Michał Kalina
250088

Na początku stworzyłem graf. Ścieżki są indeksowane poprzez indeksy dwóch punktów które łączą. Mniejszy indeks jest pierwszą cyfrą przy znakowaniu e
Tabela natężeń:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1		4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
2	2		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	3	2		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	1	3	2		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
5	1	1	3	2		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
6	5	1	1	3	2		1	1	1	1	1	1	1	1	1	1	1	1	1	1
7	1	5	1	1	3	2		1	1	1	1	1	1	1	1	1	1	1	1	1
8	1	1	5	1	1	3	2		1	1	1	1	1	1	1	1	1	1	1	1
9	1	1	1	5	1	1	3	2		1	1	1	1	1	1	1	1	1	1	1
10	1	1	1	1	5	1	1	3	2		1	1	1	1	1	1	1	1	1	1
11	1	1	1	1	1	5	1	1	3	2		1	1	1	1	1	1	1	1	1
12	1	1	1	1	1	1	5	1	1	3	2		1	1	1	1	1	1	1	1
13	1	1	1	1	1	1	1	5	1	1	3	2		1	1	1	1	1	1	1
14	1	1	1	1	1	1	1	1	5	1	1	3	2		1	1	1	1	1	1
15	1	1	1	1	1	1	1	1	1	5	1	1	3	2		1	1	1	1	1
16	1	1	1	1	1	1	1	1	1	1	5	1	1	3	2		1	1	1	1
17	1	1	1	1	1	1	1	1	1	1	1	5	1	1	3	2		1	1	1
18	1	1	1	1	1	1	1	1	1	1	1	1	5	1	1	3	2		1	1
19	1	1	1	1	1	1	1	1	1	1	1	1	1	5	1	1	3	2		1
20	1	1	1	1	1	1	1	1	1	1	1	1	1	1	5	1	1	3	2	

Topologia grafu:



Do przetestowania sprawności napisałem program w Pythonie
W programie zaimplementowałem funkcję znajdującą najkrótszą ścieżkę

```
def find_shortest_path(graph, start, end, path=[]):
    path = path + [start]
    if start == end:
        return path
    if not start in graph:
        return None
    shortest = None
    for node in graph[start]:
        if node not in path:
            newpath = find_shortest_path(graph, node, end, path)
            if newpath:
                if not shortest or len(newpath) < len(shortest):
                    shortest = newpath
    return shortest
```

Dla przedstawionych wyżej danych otrzymałem rezultaty:

- Średnie opóźnienie = 0.0034719035669292606
- Szacowana niezawodność = 0.6

Przy szacowaniu niezawodności użyłem metody monte carlo dla 1000 testów

Podpunkt 3

Przemnażam stopniowo macierz natężeń strumienia pakietów przez n:

n	niezawodność
1	0.60
1,1	0.551
1,2	0.42
1,3	0.11
1,4	0.07

Jak widać niezawodność maleje. Nie jest to zaskoczeniem. Zgodnie ze wzorem, gdy zwiększam macierz, rośnie $a(e)$. Wtedy T maleje, przez co, $\Pr(T < T_{\max})$ rośnie.

Podpunkt 4

Przemnażam stopniowo macierz natężeń strumienia pakietów przez c:

c	niezawodność
0.999	0.46
1	0,6
4	0.58
10	0.59
1000	0.58

Przy zwiększaniu przepustowości rośnie $c(e)$. Wtedy T maleje, przez co, $\Pr(T < T_{\max})$ rośnie.

Przy $C \rightarrow \infty$ powinna dążyć do niezawodności z pustą tabelą natężeń. W moim przykładzie dąży do około 0.6

Podpunkt 5

Po dodaniu ścieżki $\{1, 5\}$ o wielkości 14 pakietów, oraz $p=0.7$ wynik się poprawił do 0.63.

Po dodaniu ścieżki $\{16, 20\}$ wynik urósł do 0.66

Po dodaniu ścieżki $\{8, 20\}$ wynik urósł do 0.68

Z każdą następną dodawaną krawędzią wynik się poprawiał, szacuję że w granicy osiągnie on 1.

Problemem może być jedynie dodanie krawędzi o zbyt niskiej przepustowości