

# PROJEKTOWANIE EFEKTYWNYCH ALGORYTMÓW

## Projekt 2

Prowadzący:  
Dr inż. Zbigniew Buchalski

## Spis treści

1.	Wstęp .....	3
2.	Analiza Problemu.....	3
3.	Opis Algorytmów .....	3
3.1.	Tabu Search .....	3
3.2.	Symulowane Wyżarzanie.....	3
4.	Wyniki.....	4
4.1.	Tabu Search .....	4
4.1.1.	Dla 47 miast.....	4
4.1.2.	Dla 170 miast.....	6
4.1.3.	Dla 403 miast.....	8
4.2.	Symulowane wyżarzanie .....	10
4.2.1.	Dla 47 miast.....	10
4.2.2.	Dla 170 miast.....	12
4.2.3.	Dla 403 miast.....	15
4.3.	Porównanie algorytmów .....	17
5.	Wnioski.....	18

## 1. Wstęp

Celem projektu było zaimplementowanie oraz dokonanie analizy efektywności działania algorytmu Tabu Search oraz Symulowanego Wyżarzania dla problemu komiwojażera.

## 2. Analiza Problemu

Problem komiwojażera należy do klasy problemów NP-trudnych. Polega on na znalezieniu minimalnego cyklu Hamiltona w pełnym grafie. Każdy wierzchołek grafu reprezentuje miasto, które musi odwiedzić komiwojażer. Grupę  $n$  miast reprezentuje zbiór  $N = \{1, \dots, n\}$ . Miasta są ze sobą połączone krawędziami  $d$ . Długość tych krawędzi zawiera macierz  $D = \{d_{ij}, i \in N, j \in N, i \neq j\}$ . Gdzie  $d_{ij} \geq 0$  oznacza odległość między miastem  $i$  oraz  $j$ . W wersji asymetrycznej, odległość pomiędzy miastem  $i$  oraz  $j$  może być inna niż odległość miasta  $j$  od  $i$ :  $d_{ij} \neq d_{ji}$ . Główną trudnością w rozwiązaniu problemu jest znacząca liczba możliwych kombinacji.

## 3. Opis Algorytmów

### 3.1. Tabu Search

Algorytm metaheurestyczny korzystający z przeszukiwania lokalnego stosowany do rozwiązywania problemów optymalizacyjnych. Metoda oparta na iteracyjnym przeszukiwaniu przestrzeni rozwiązań, wykorzystując sąsiedztwo pewnych elementów tej przestrzeni oraz zapamiętując przy tym przeszukiwaniu ostatnie ruchy (transformacje rozwiązań) i częstość ich występowania, w celu unikania minimów lokalnych i poszukiwania rozwiązań globalnie optymalnych w rozsądnym czasie.

Lista Tabu jest reprezentowana za pomocą tablicy, do której wpisuje się ruchy zmieniające poprzednie rozwiązanie zwane kadencją. Kadencja mówi przez jaki czas rozwiązania w sąsiedztwie nie będą mogły korzystać z wierzchołków posiadających kadencje. Jest to wykorzystywane po to, aby algorytm unikał wpadania w cykle. Podczas działania algorytmu może nastąpić jednak wyjątek od listy tabu. Odpowiada za to funkcja aspiracji, której działanie jest regulowane poprzez parametr. Następuje to w przypadku, gdy zabroniony ruch ma być ruchem bardzo dobrym (lepszym od aktualnego optimum).

### 3.2. Symulowane Wyżarzanie

Algorytm heurestyczny którego ideę obrazuje stosowana od wieków metoda wyżarzania metalu w procesie hartowania. Metoda działa iteracyjnie, krok po kroku zbliżając się do rozwiązania optymalnego. Jako kolejne przybliżenie rozwiązania może być wybrany dowolny element przestrzeni potencjalnych rozwiązań. Jednak jego wybór jest uzależniony od dwóch podstawowych czynników. Po pierwsze - od różnicy wartości między starym rozwiązaniem a propozycją nowego rozwiązania. Jeżeli propozycja nowego rozwiązania jest lepsza od swego poprzednika, to zostaje ona zatwierdzona jako nowe rozwiązanie (ponieważ lepiej przybliża rozwiązanie optymalne). Natomiast, jeśli nowa propozycja jest gorsza od dotychczasowego rozwiązania, wybiera się je drogą losowania. Wówczas prawdopodobieństwo wyboru nowej propozycji rozwiązania jest tym mniejsze, im różnica między rozwiązaniami jest większa. Ma to na celu ograniczenie zbytńego oddalenia się od wcześniej znalezionego rozwiązania.

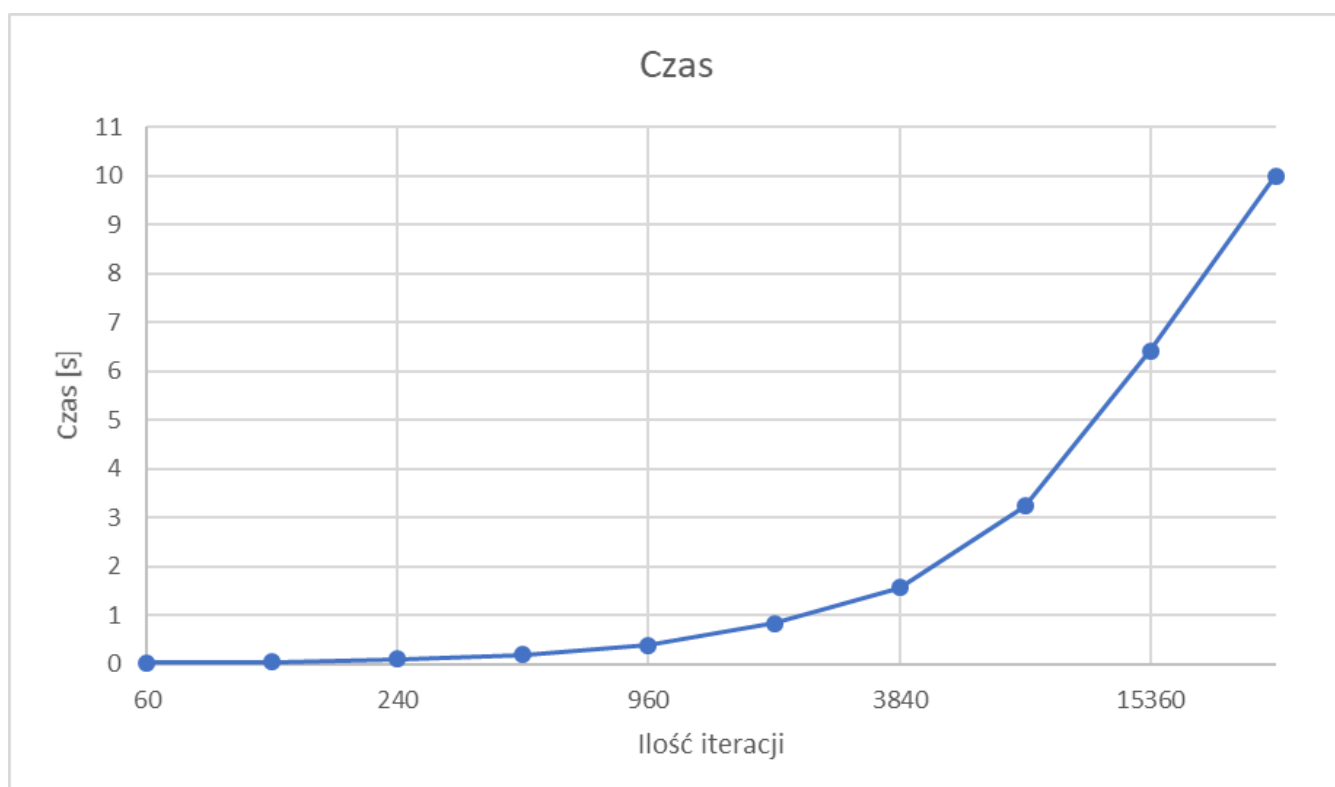
Parametr nazywany temperaturą reguluje proces wyboru kolejnych przybliżeń w przypadku, gdy propozycja nowego rozwiązania jest gorsza. Jeżeli wartość temperatury jest duża, to prawdopodobieństwo wyboru jest duże. Jeżeli wartość temperatury jest mała, to prawdopodobieństwo wyboru nowego rozwiązania, gorszego od poprzednika, jest bardzo małe. Temperatura jest w miarę postępowania algorytmu stale obniżana. Oznacza to, że w kolejnych krokach szansa przejścia do nowego, gorszego położenia maleje. Z upływem czasu rozwiązanie stabilizuje się, aż wreszcie żadne zmiany nie są akceptowane. Algorytmy tego typu są wykorzystywane do poszukiwania rozwiązań bliskich rozwiązaniom optymalnym w dużych zadaniach optymalizacyjnych.

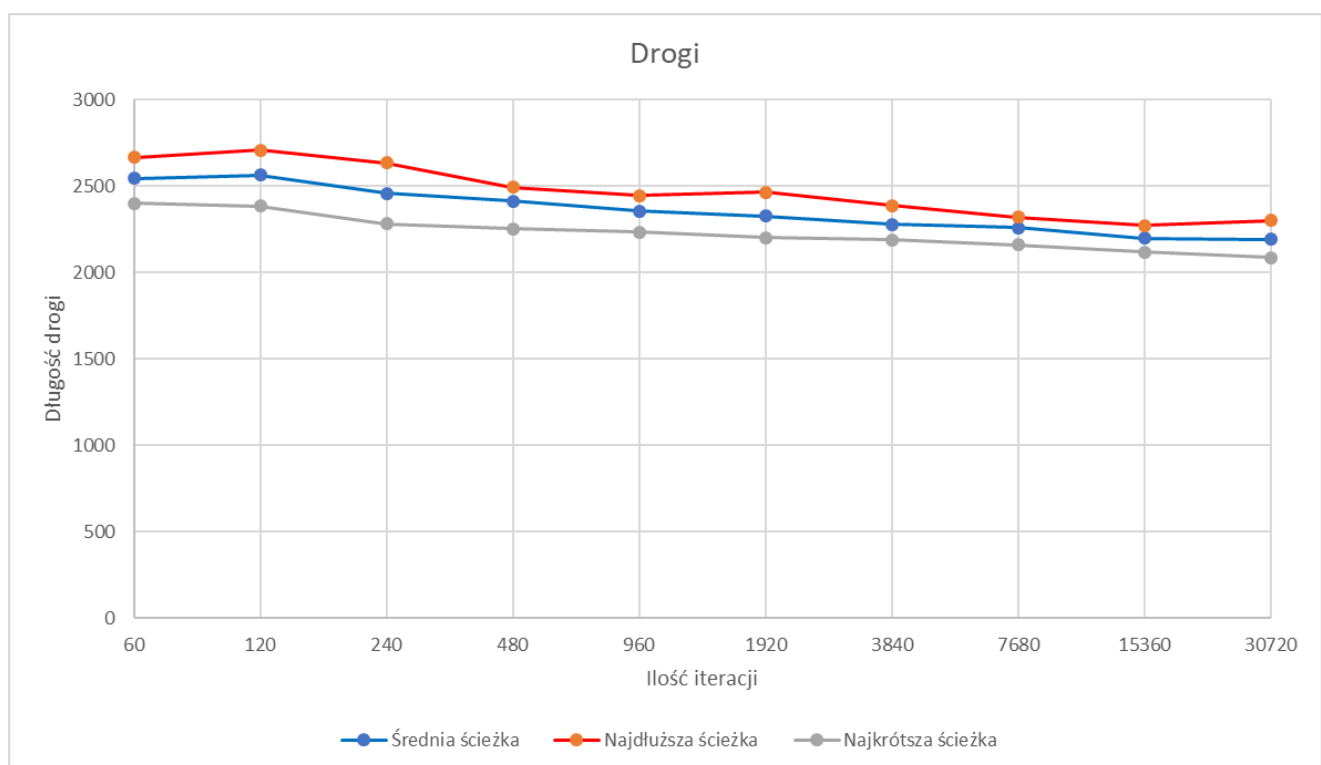
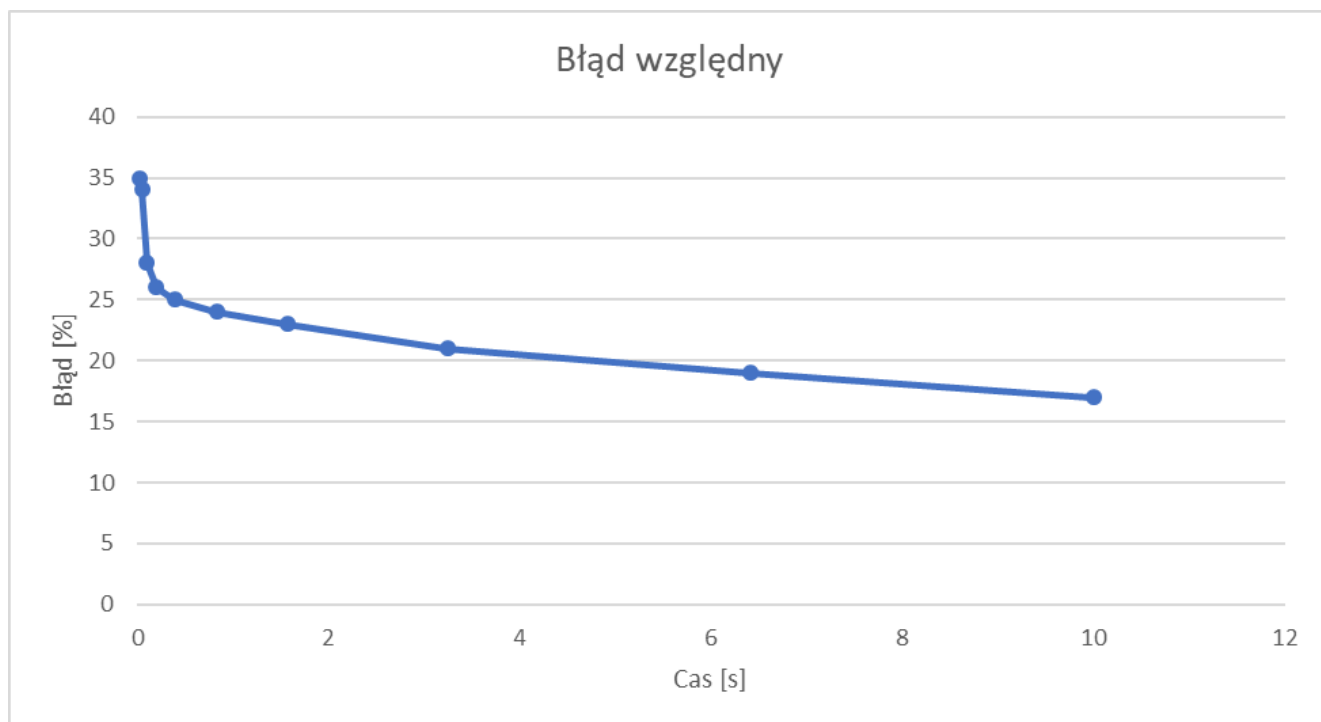
## 4. Wyniki

### 4.1. Tabu Search

#### 4.1.1. Dla 47 miast

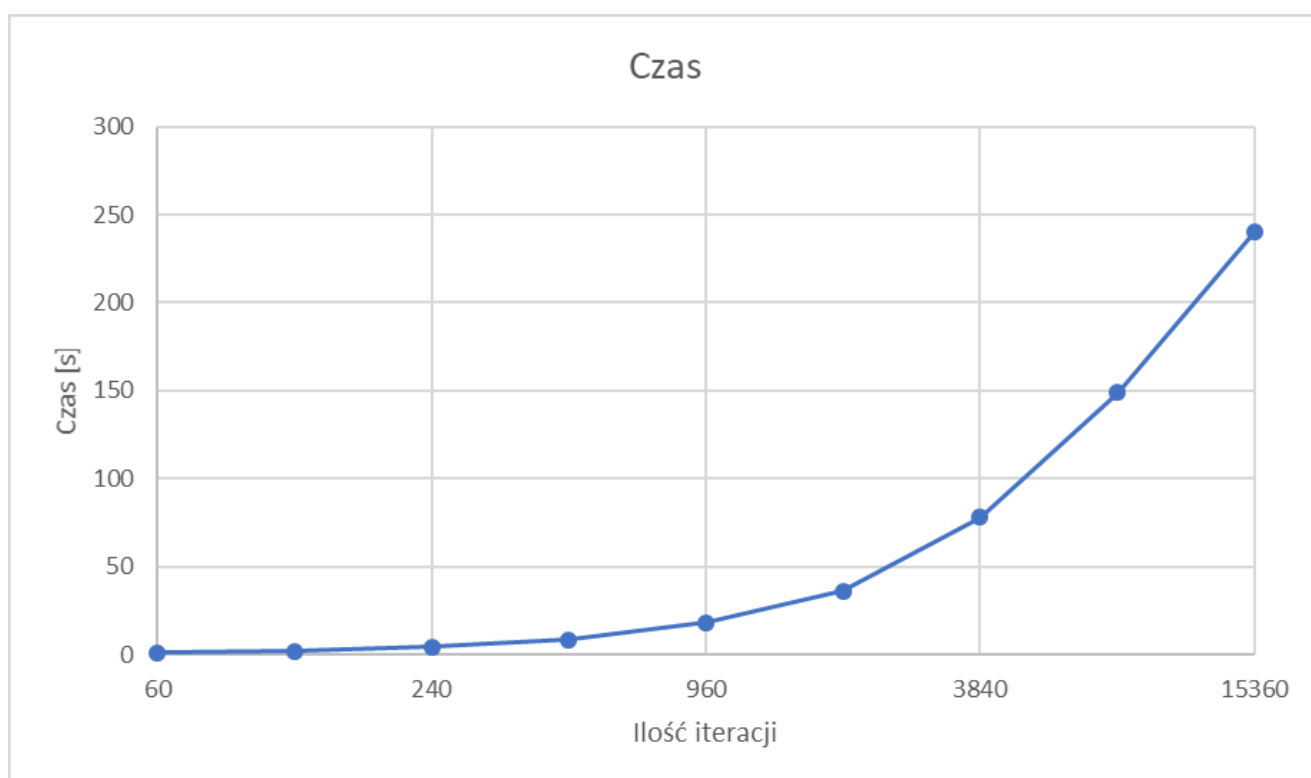
Liczba iteracji	Średni czas [s]	Najkrótsza ścieżka	Średnia ścieżka	Najdłuższa ścieżka	Wartość optymalna	Błąd [%]
60	0,0244950	2402	2545	2667	1776	35
120	0,0479596	2386	2566	2709	1776	34
240	0,0985954	2282	2461	2637	1776	28
480	0,1958520	2254	2415	2494	1776	26
960	0,3888060	2235	2357	2447	1776	25
1920	0,8310030	2203	2328	2467	1776	24
3840	1,5702000	2191	2279	2388	1776	23
7680	3,2418800	2160	2261	2322	1776	21
15360	6,4161900	2119	2200	2272	1776	19
30720	10,0003000	2086	2192	2303	1776	17

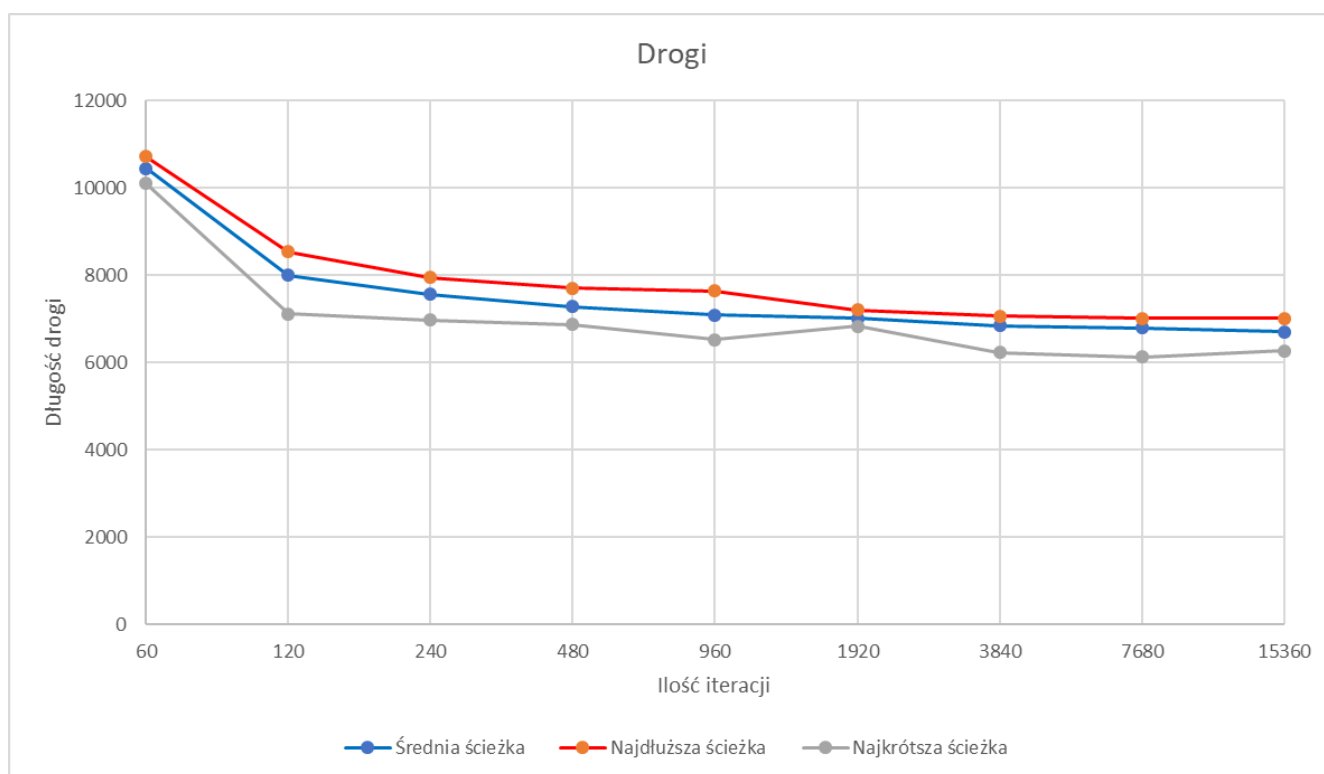
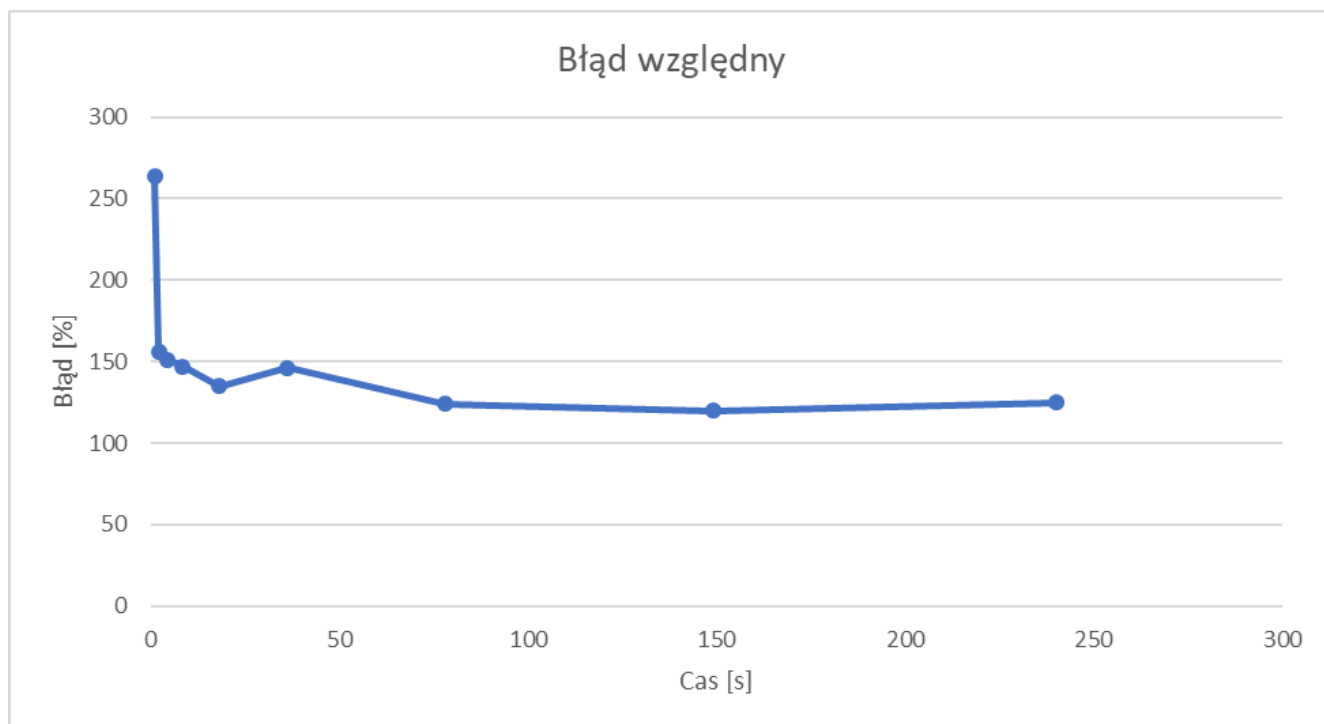




#### 4.1.2. Dla 170 miast

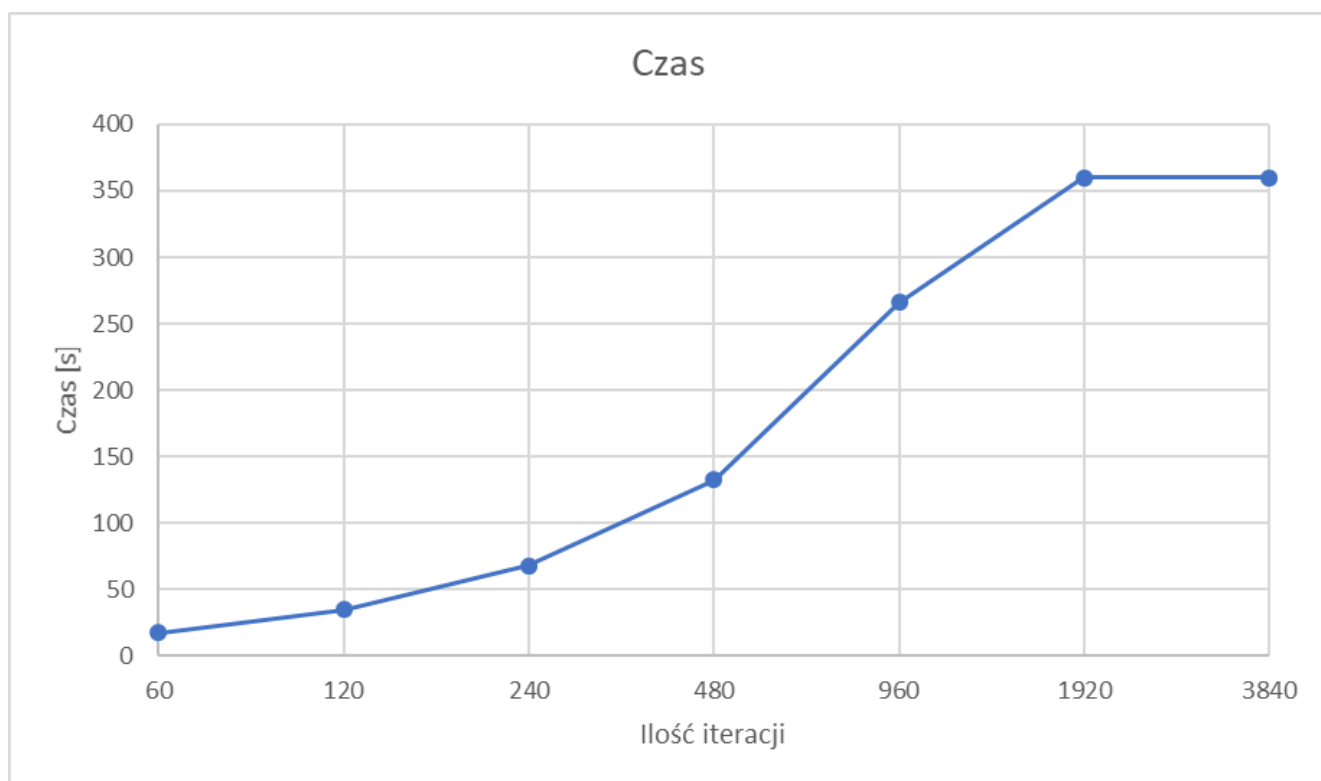
Liczba iteracji	Średni czas [s]	Najkrótsza ścieżka	Średnia ścieżka	Najdłuższa ścieżka	Wartość optymalna	Błąd [%]
60	1,0438100	10110	10450	10724	2775	264
120	2,1203400	7116	8004	8545	2775	156
240	4,5276500	6974	7562	7954	2775	151
480	8,5255700	6877	7286	7711	2775	147
960	18,1850000	6528	7090	7640	2775	135
1920	36,3112000	6836	7020	7205	2775	146
3840	78,2089000	6237	6844	7064	2775	124
7680	149,0010000	6128	6796	7016	2775	120
15360	240,0100000	6267	6710	7011	2775	125



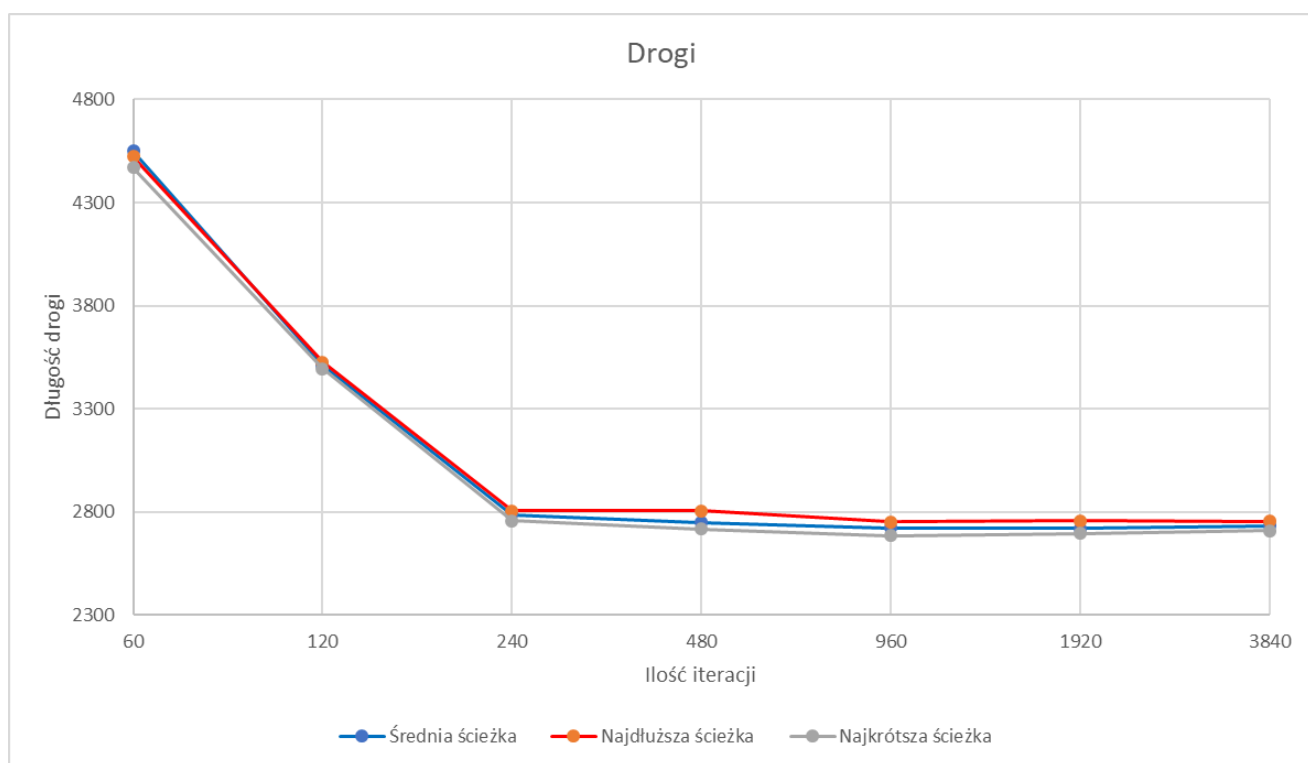
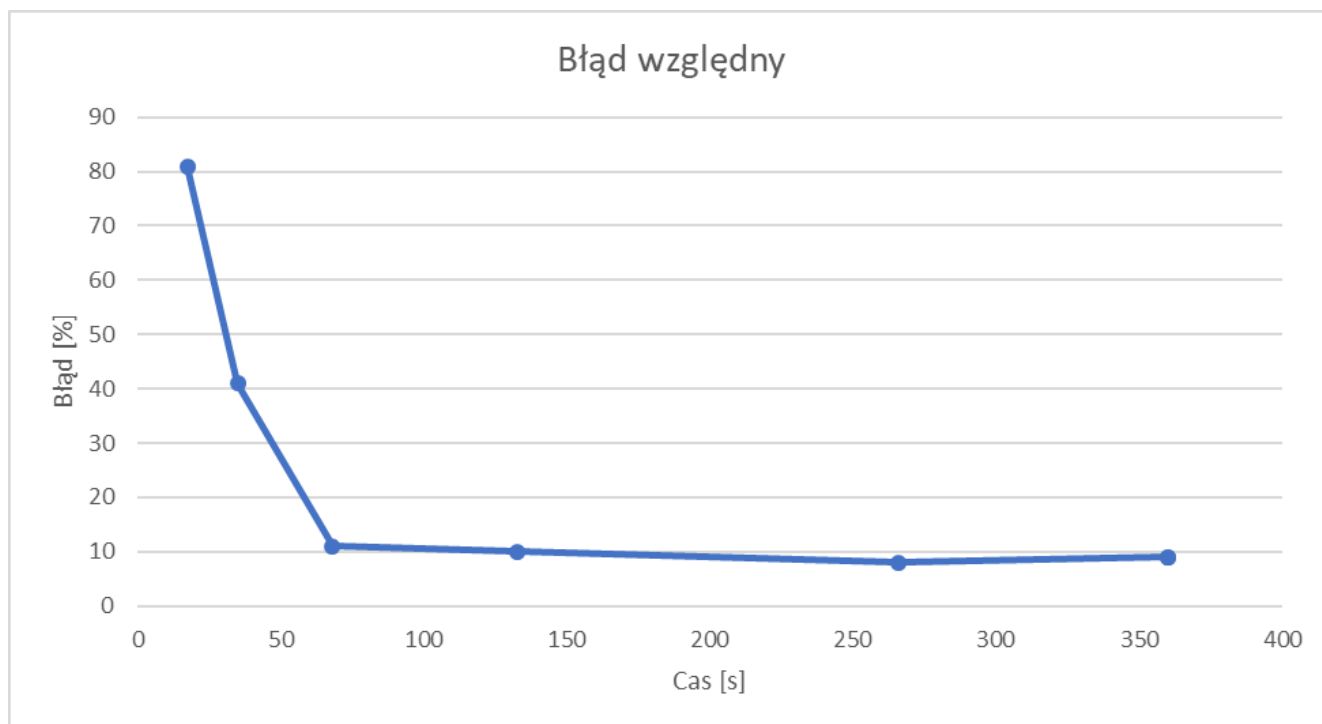


#### 4.1.3. Dla 403 miast

Stała	Średni czas [s]	Najkrótsza ścieżka	Średnia ścieżka	Najdłuższa ścieżka	Wartość optymalna	Błąd [%]	Temperatura końcowa
0,75	0,0348614	7181	7497	7801	2465	191	0,0115852
0,8	0,0417768	7211	7442	7639	2465	192	0,0103933
0,85	0,0400786	7114	7310	7423	2465	188	0,0103414
0,9	0,0357721	6970	7202	7405	2465	182	0,0103987
0,95	0,0383383	6480	6776	7212	2465	162	0,0101477
0,975	0,0363355	6152	6327	6610	2465	149	0,0101283
0,999	0,0593811	3923	4037	4123	2465	59	0,0100037
0,9999	0,2788750	2989	3042	3087	2465	21	0,0100010
0,99999	2,4015100	2569	2602	2632	2465	4	0,0100001
0,999999	24,0783000	2498	2522	2541	2465	1	0,0100000



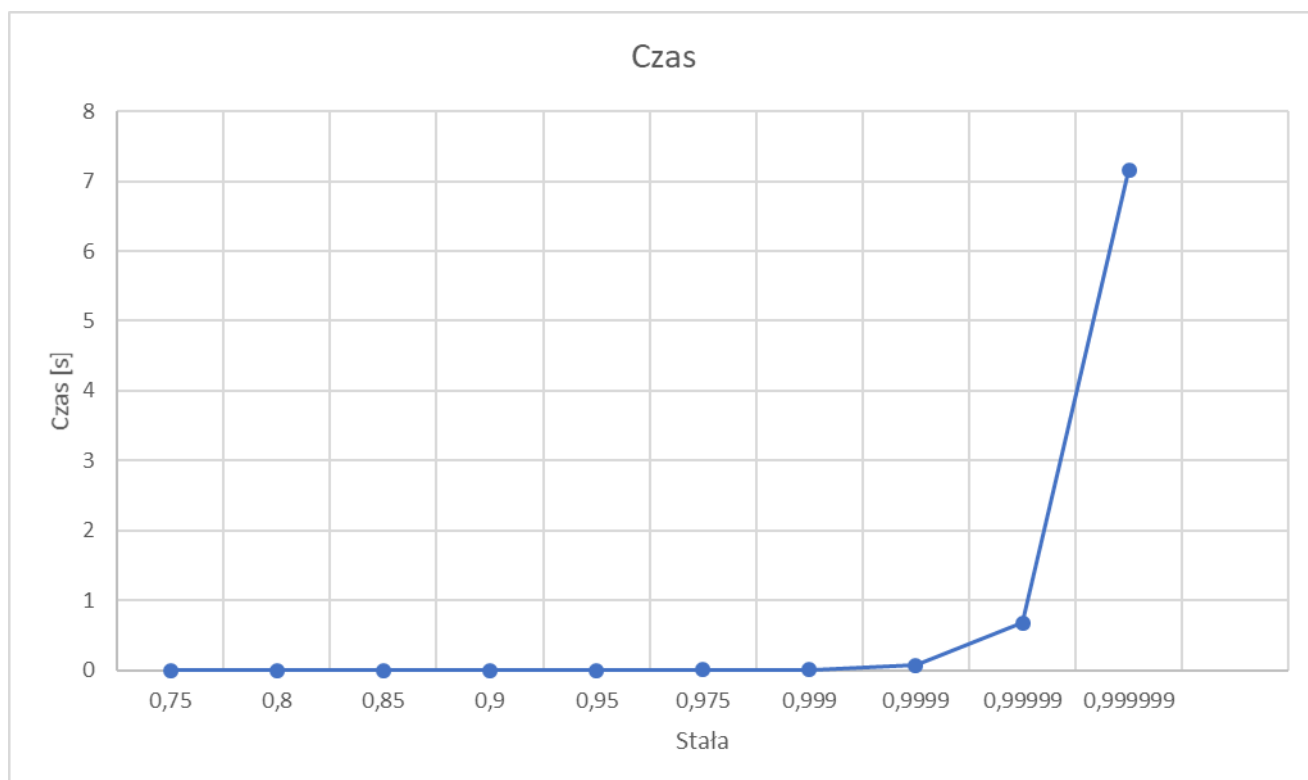


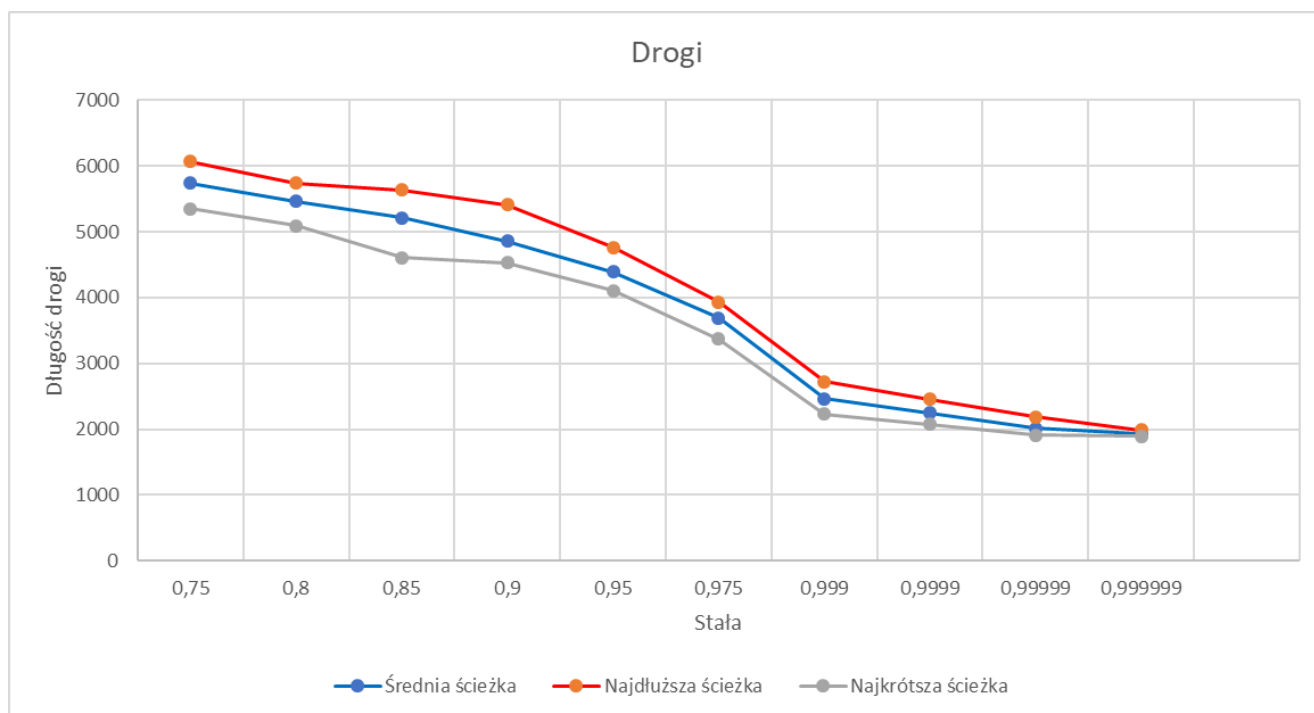
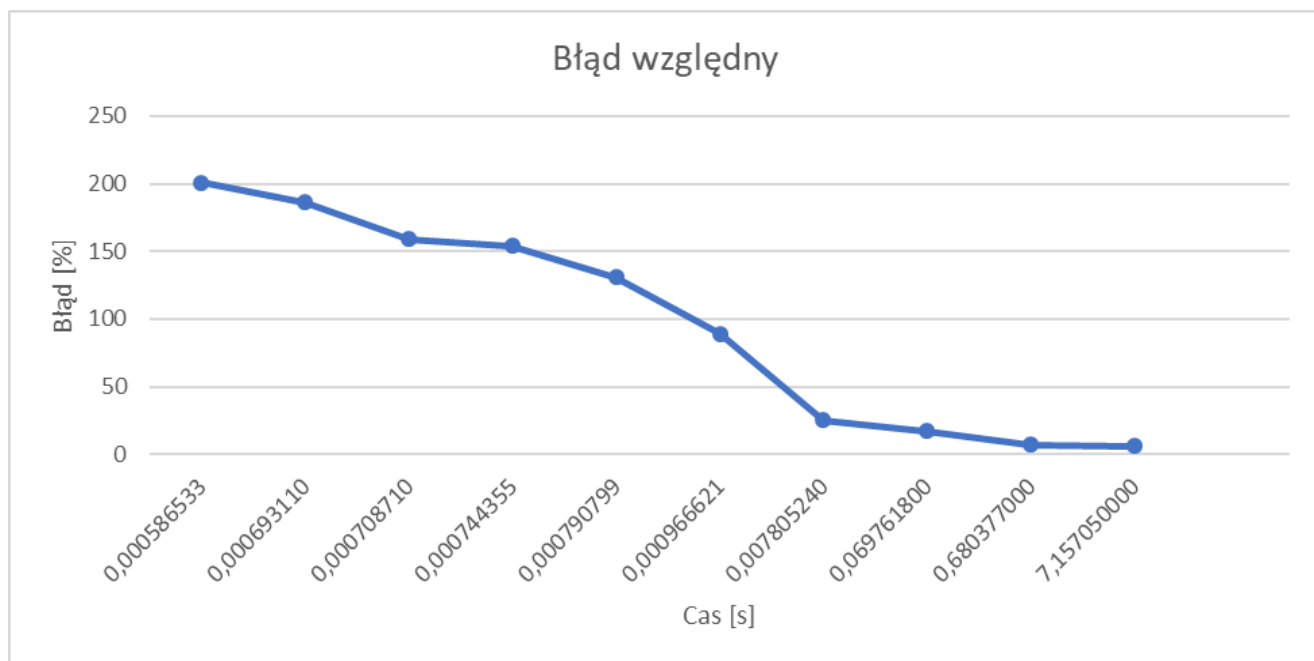


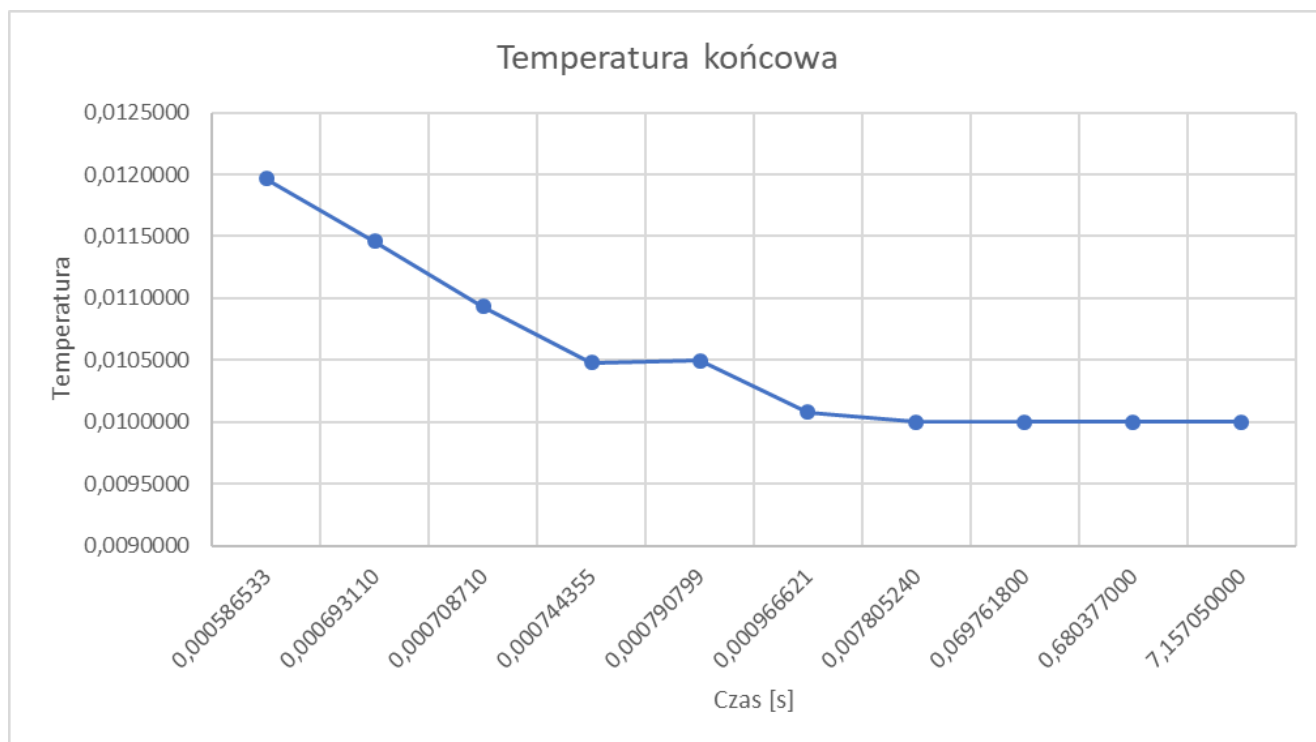
## 4.2. Symulowane wyżarzanie

### 4.2.1. Dla 47 miast

Stała	Średni czas [s]	Najkrótsza ścieżka	Średnia ścieżka	Najdłuższa ścieżka	Wartość optymalna	Błąd [%]	Temperatura końcowa
0,75	0,000586533	5346	5743	6067	1776	201	0,0119645
0,8	0,000693110	5093	5459	5739	1776	186	0,0114624
0,85	0,000708710	4609	5208	5638	1776	159	0,0109320
0,9	0,000744355	4528	4856	5411	1776	154	0,0104799
0,95	0,000790799	4109	4390	4764	1776	131	0,0104935
0,975	0,000966621	3371	3690	3936	1776	89	0,0100781
0,999	0,007805240	2227	2465	2724	1776	25	0,0100004
0,9999	0,069761800	2079	2249	2459	1776	17	0,0100005
0,99999	0,680377000	1915	2014	2191	1776	7	0,0100000
0,999999	7,157050000	1891	1932	1989	1776	6	0,0100000

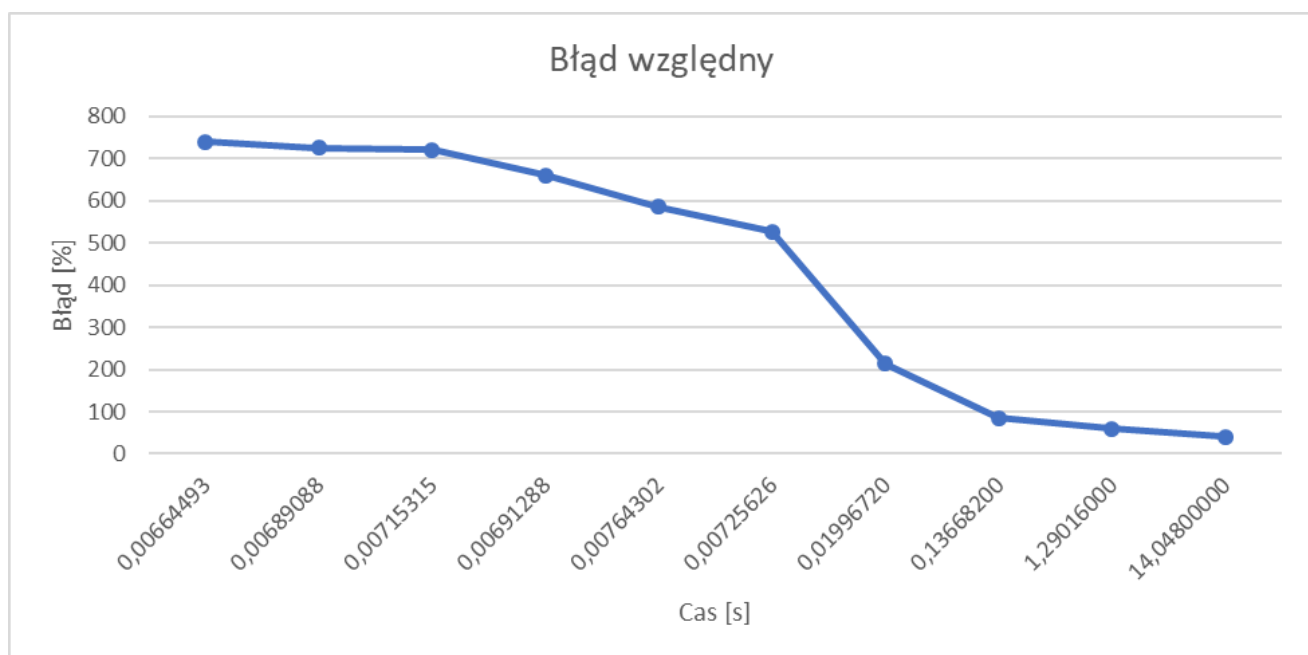
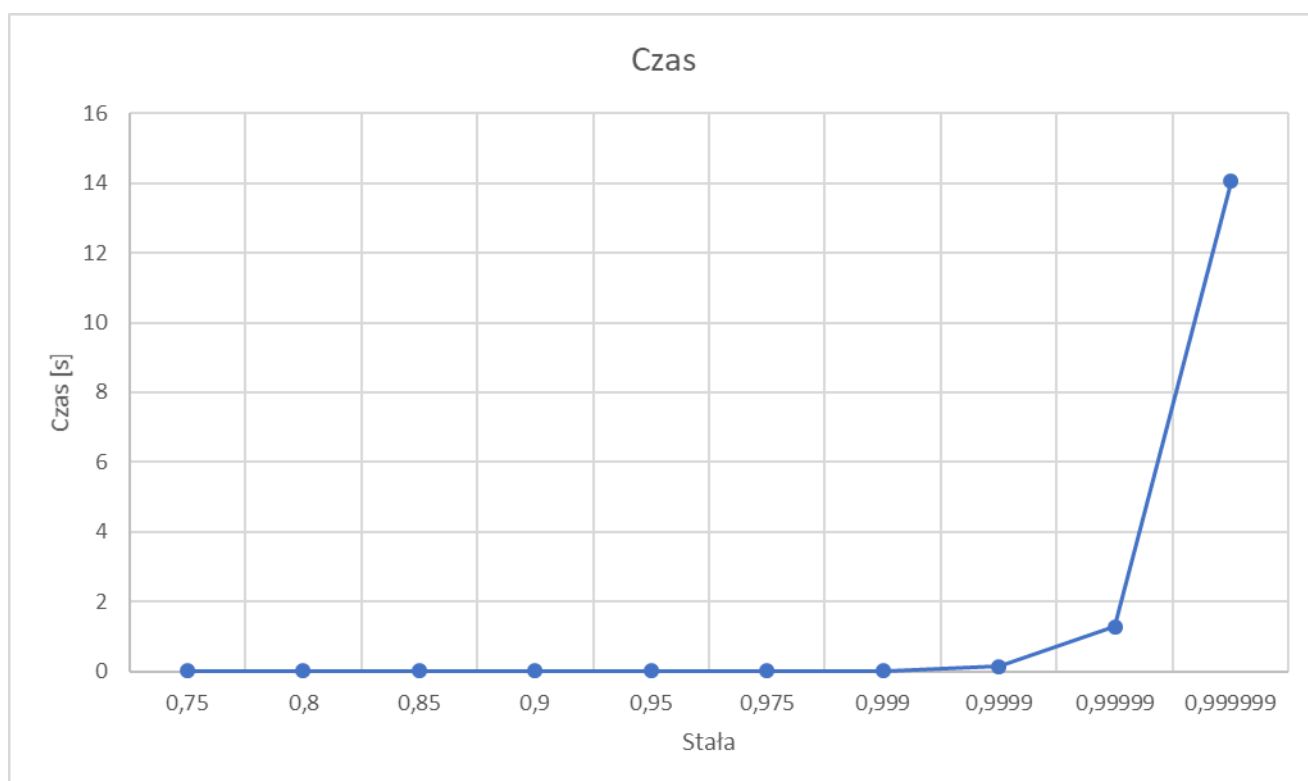


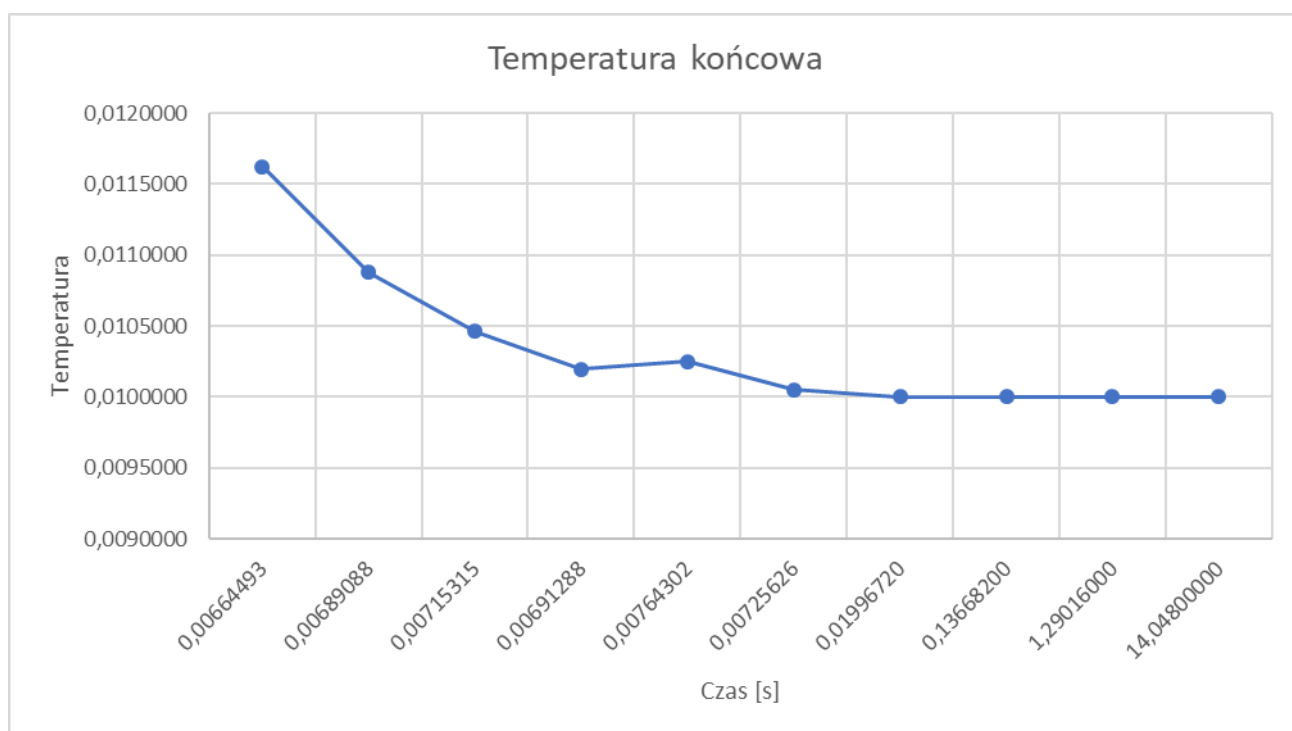
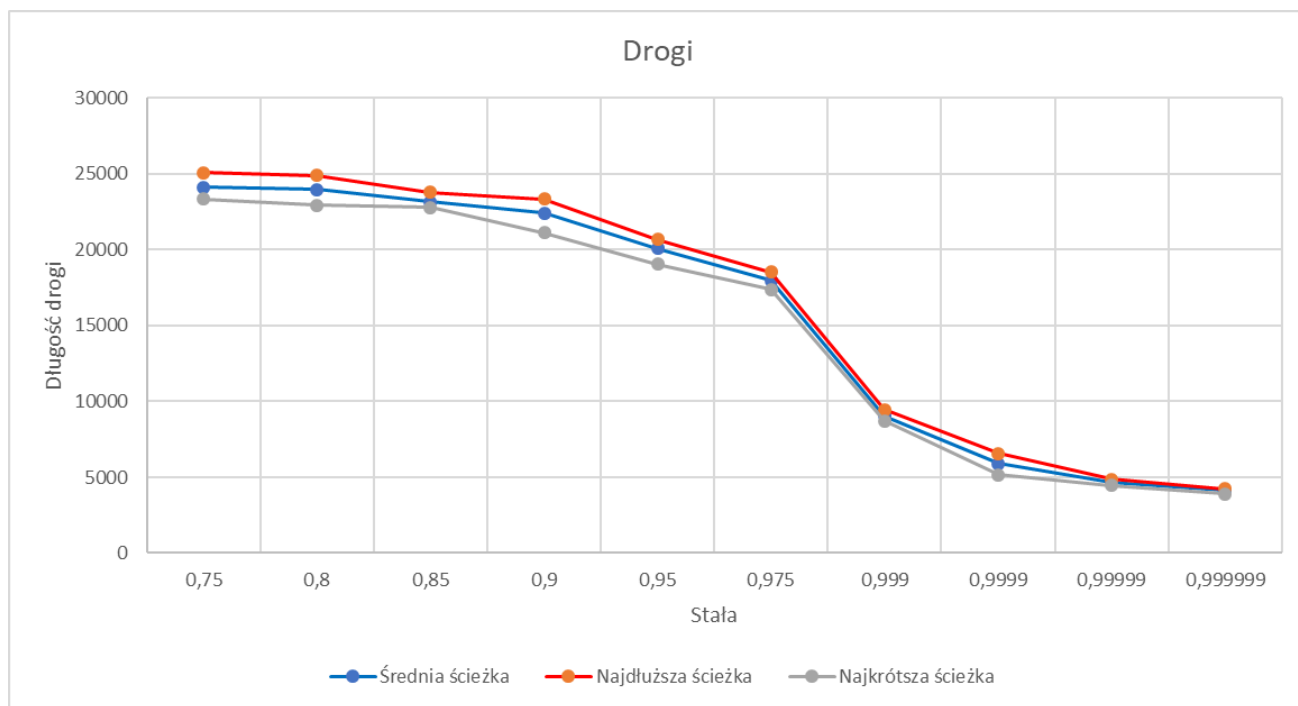




#### 4.2.2. Dla 170 miast

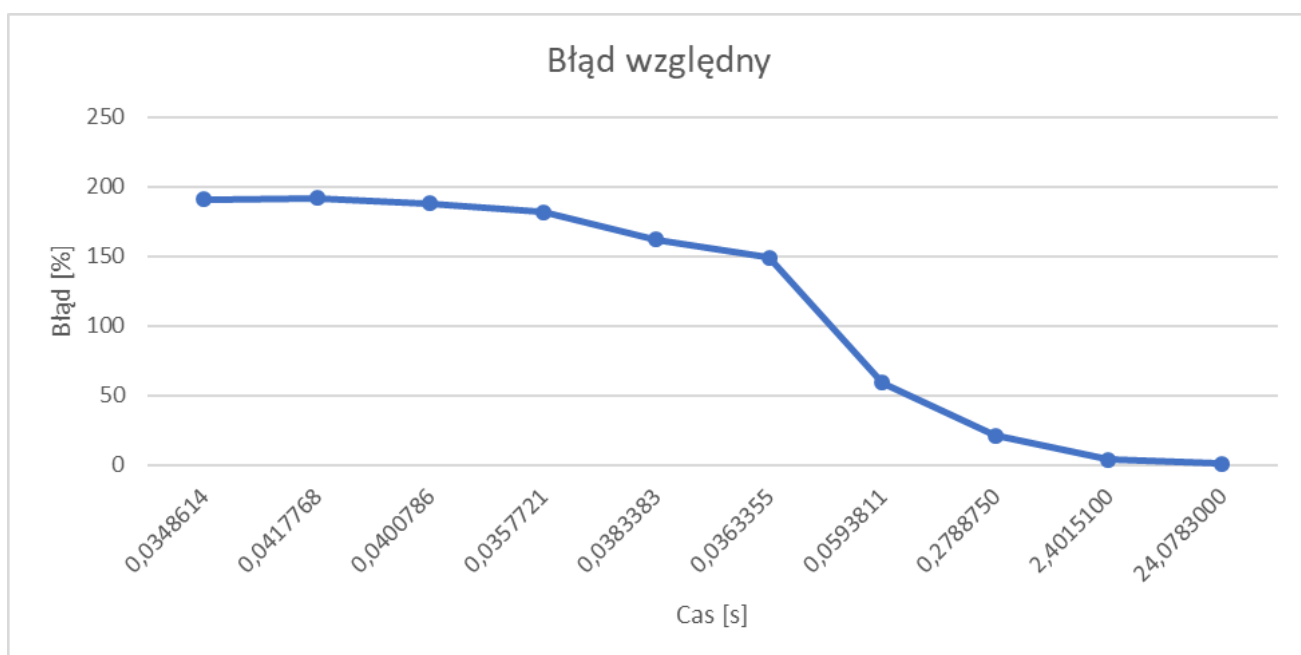
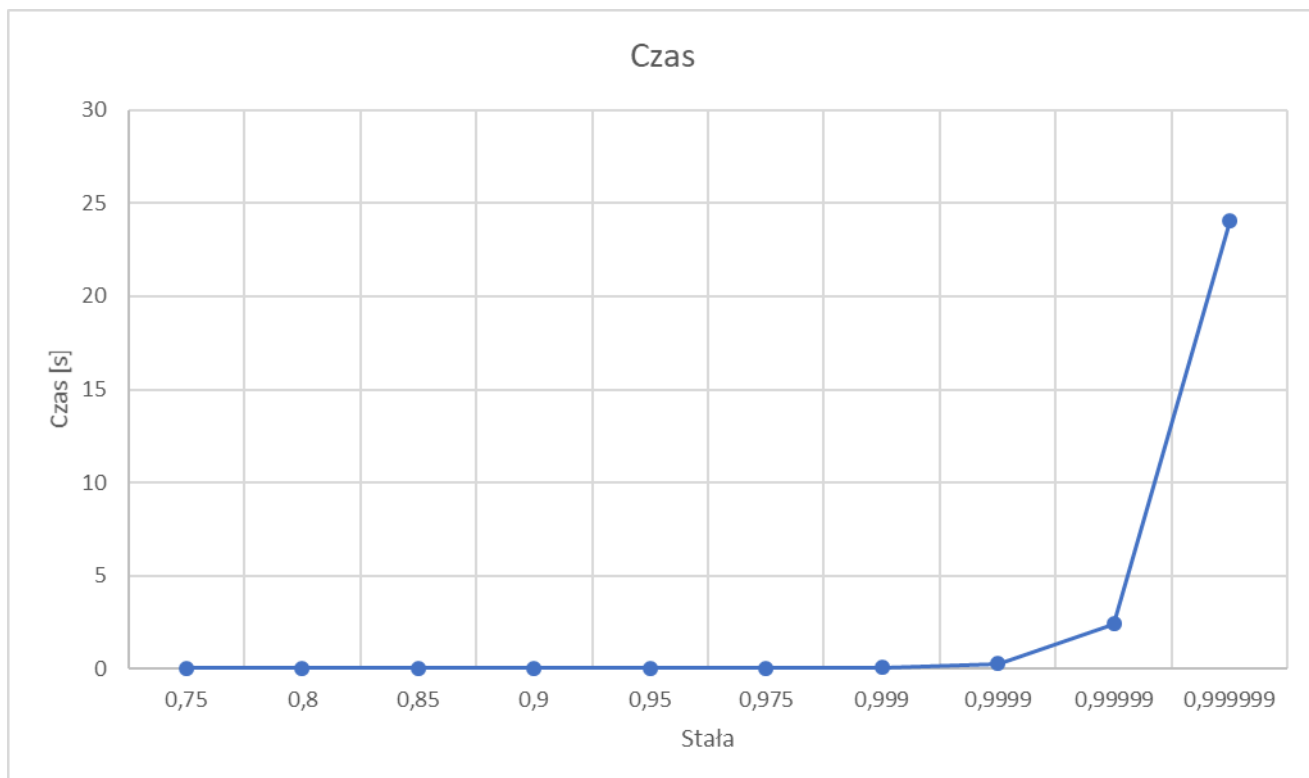
Średni czas [s]	Najkrótsza ścieżka	Średnia ścieżka	Najdłuższa ścieżka	Wartość optymalna	Błąd [%]	Temperatura końcowa
0,00664493	23332	24133	25091	2775	740	0,0116237
0,00689088	22938	23980	24883	2775	726	0,0108783
0,00715315	22782	23164	23803	2775	720	0,0104639
0,00691288	21114	22418	23361	2775	660	0,0101953
0,00764302	19059	20076	20689	2775	586	0,0102477
0,00725626	17378	17987	18544	2775	526	0,0100495
0,01996720	8701	9051	9455	2775	213	0,0100019
0,13668200	5147	5922	6563	2775	85	0,0100005
1,29016000	4443	4686	4880	2775	60	0,0100001
14,04800000	3896	4123	4252	2775	40	0,0100000

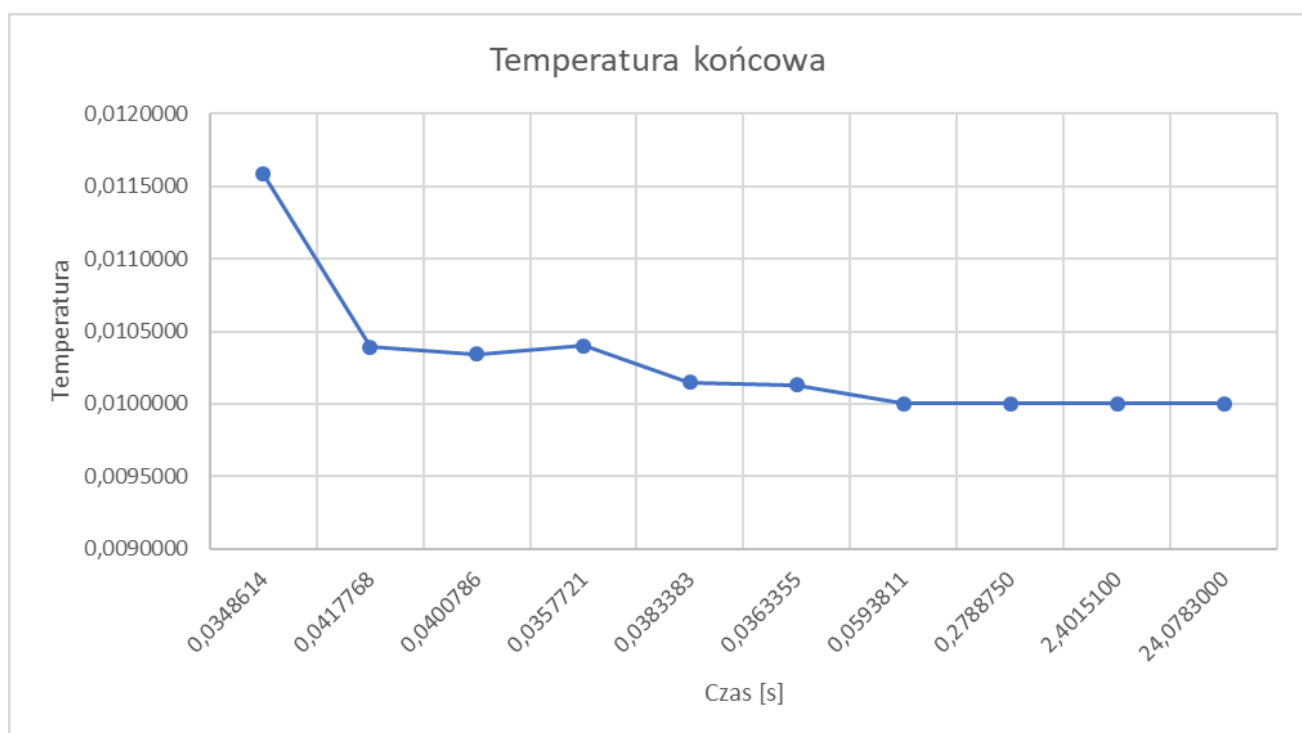
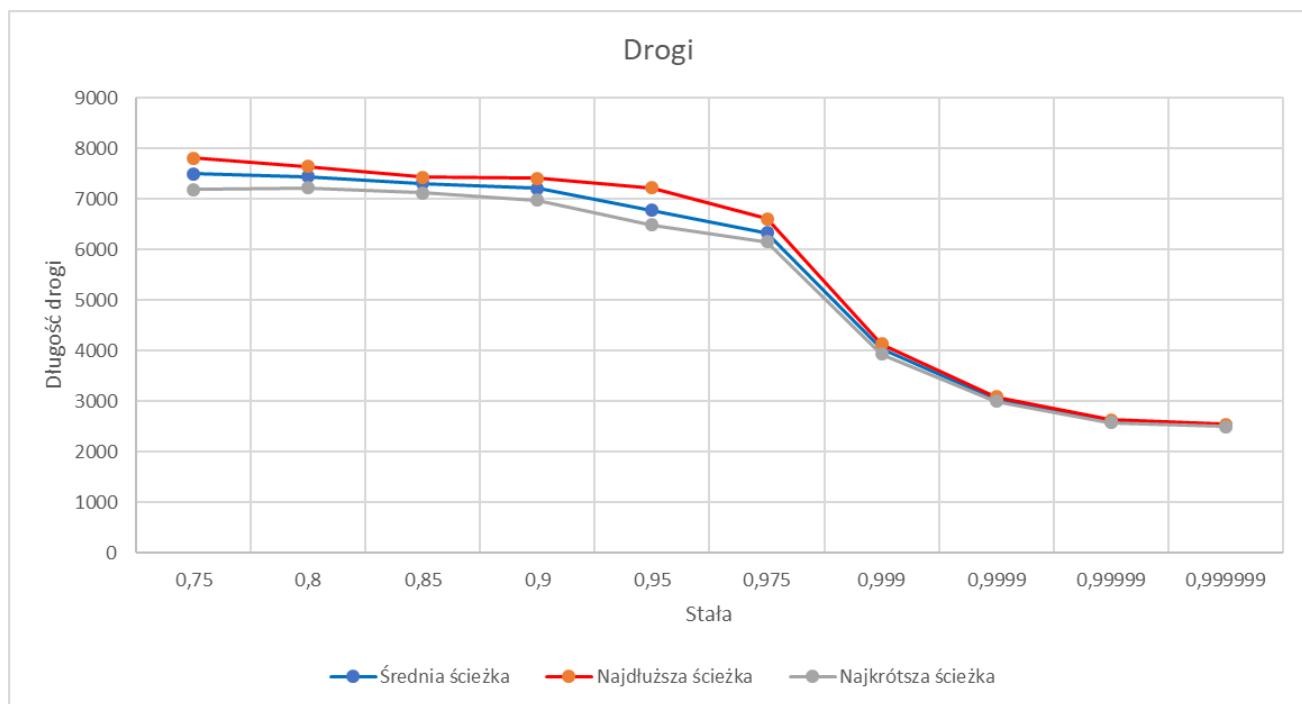




#### 4.2.3. Dla 403 miast

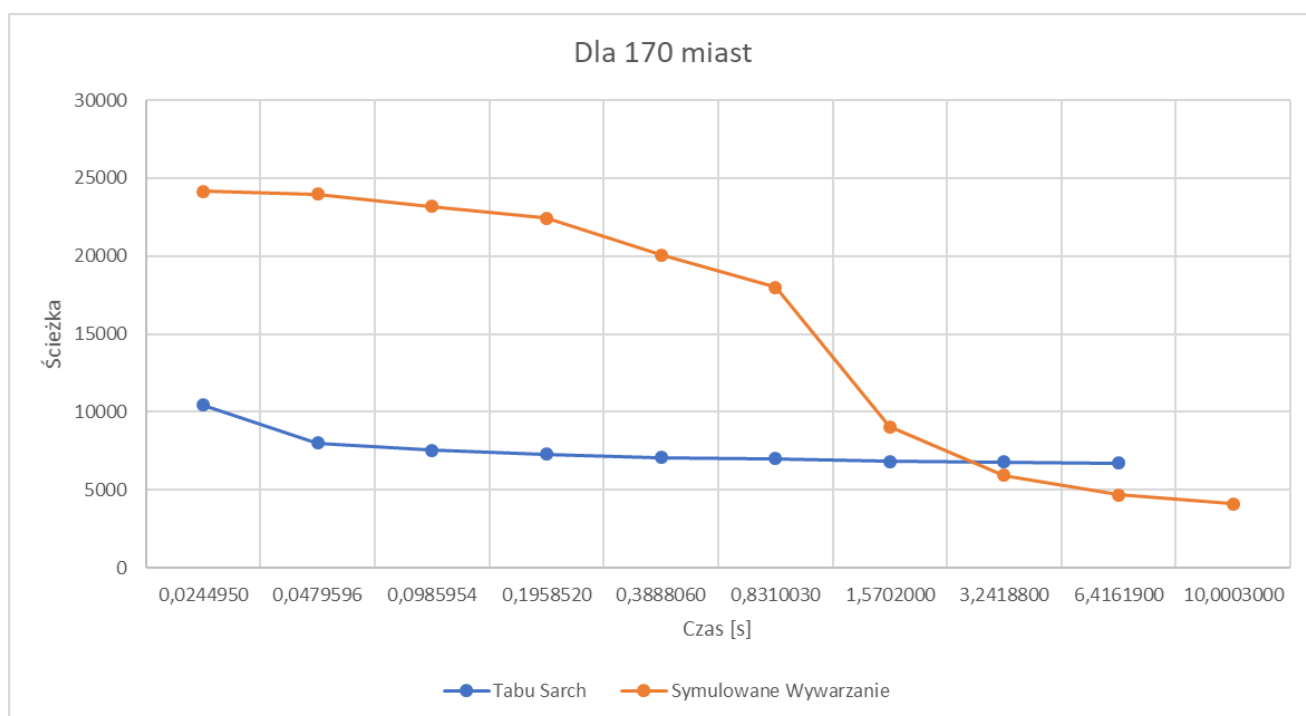
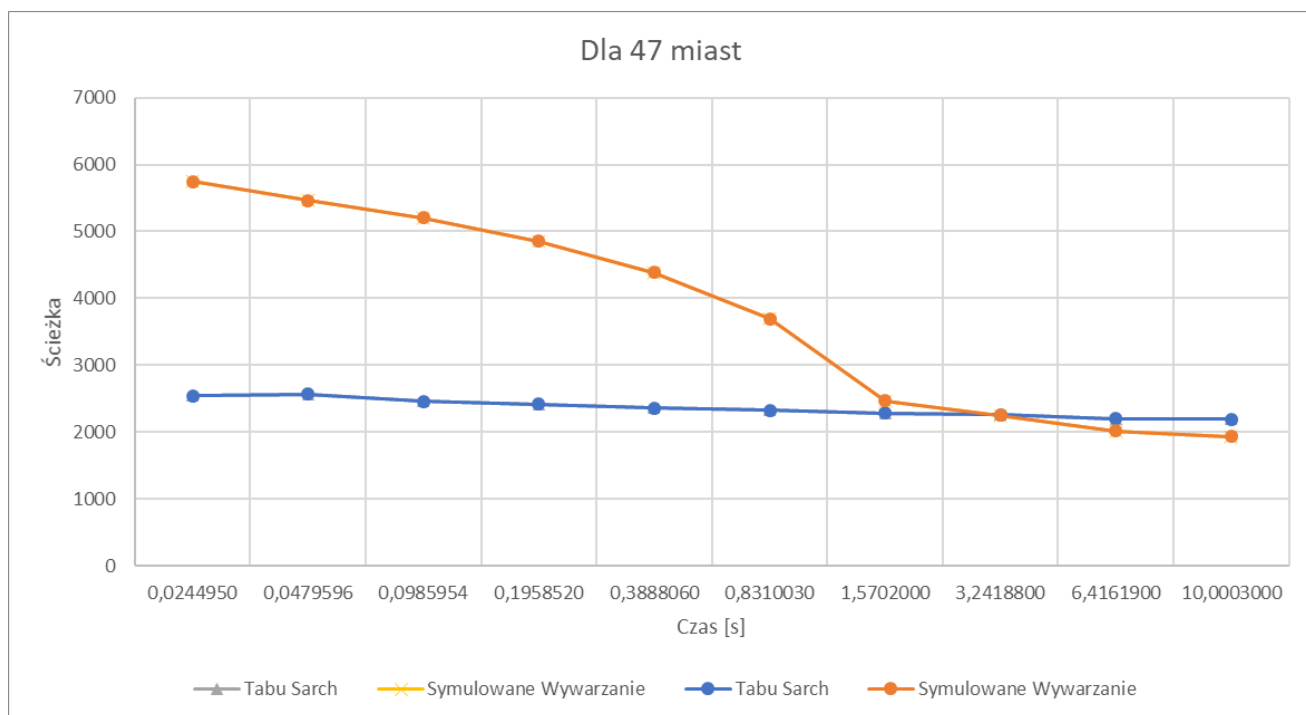
Stała	Średni czas [s]	Najkrótsza ścieżka	Średnia ścieżka	Najdłuższa ścieżka	Wartość optymalna	Błąd [%]	Temperatura końcowa
0,75	0,0348614	7181	7497	7801	2465	191	0,0115852
0,8	0,0417768	7211	7442	7639	2465	192	0,0103933
0,85	0,0400786	7114	7310	7423	2465	188	0,0103414
0,9	0,0357721	6970	7202	7405	2465	182	0,0103987
0,95	0,0383383	6480	6776	7212	2465	162	0,0101477
0,975	0,0363355	6152	6327	6610	2465	149	0,0101283
0,999	0,0593811	3923	4037	4123	2465	59	0,0100037
0,9999	0,2788750	2989	3042	3087	2465	21	0,0100010
0,99999	2,4015100	2569	2602	2632	2465	4	0,0100001
0,999999	24,0783000	2498	2522	2541	2465	1	0,0100000

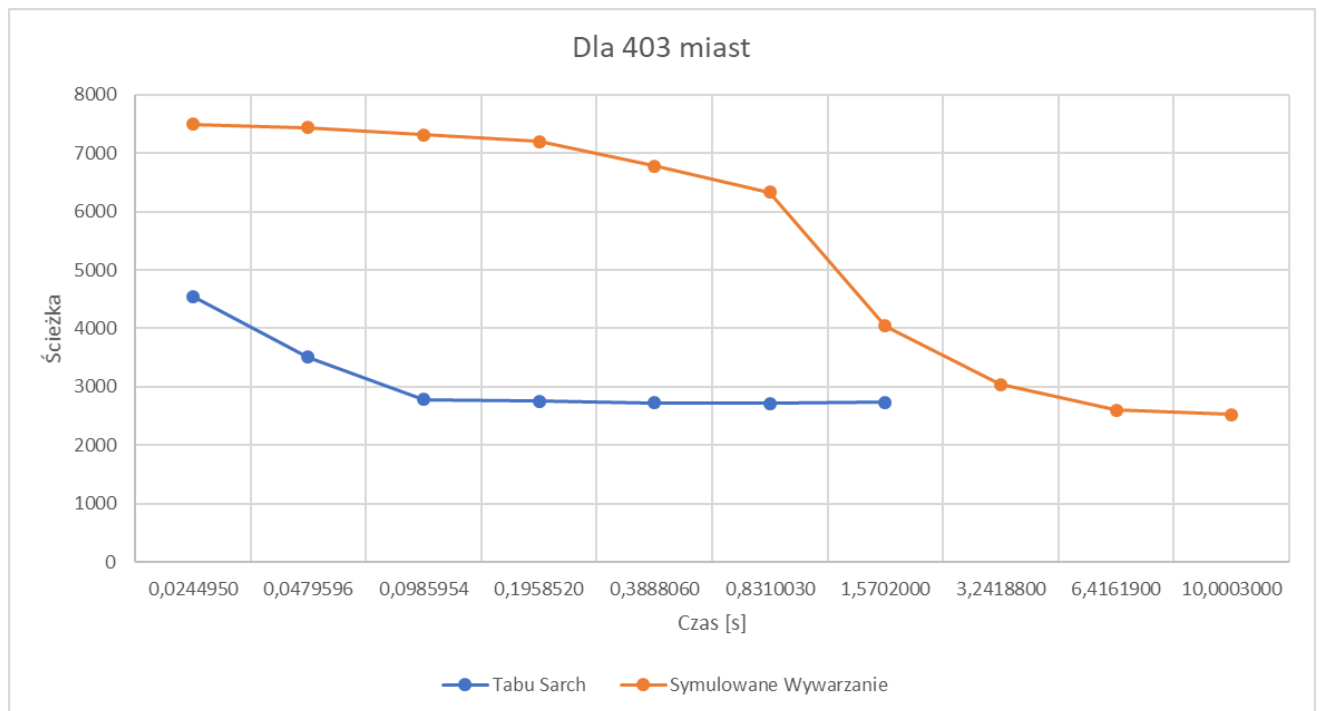






### 4.3. Porównanie algorytmów





## 5. Wnioski

Zaimplementowane algorytmy dają zróżnicowane wyniki, niektóre trasy są z błędami akceptowalnymi, niektóre nie. W porównaniu z poprzednim projektem algorytmy działają o wiele szybciej oraz potrafią obliczać większe instancje. Porównując analizowane algorytmy lepsze wyniki oferuje algorytm Symulowanego Wyżarzania. Natomiast Tabu Search wydaje się być bardziej stabilnym, ponieważ dawał mniejsze błędy względne. Czas wykonywania algorytmu SW rośnie wraz z dokładnością stałą, natomiast TS zwiększa się wraz ze wzrostem liczby iteracji.