

Projektowanie Efektywnych Algorytmow

Projekt

20/12/2023

264472 Jakub Chuchla

(4) Simulated Annealing

Spis treści

1. Sformułowanie zadania	3
1.1. Problem Komiwojazera	3
2. Metoda	4
2.1. Wybór rozwiązania początkowego	4
2.2. Wybór temperatury początkowej	4
2.3. Zwiedzanie sąsiedztwa	5
2.4. Metody chłodzenia	5
3. Algorytm i sposoby implementacji.....	6
3.1. Pseudo kod i schemat blokowy.....	6
3.2. Rozwiązanie początkowe	7
3.3. Wybór temperatury początkowej	8
3.4. Zwiedzanie sąsiedztwa	8
3.5. Schemat chłodzenia	9
4. Dane testowe	9
4.1. Symetryczne	9
4.2. Asymetryczne	10
5. Procedura badawcza	11
6. Badania i wyniki	13
6.1. Specyfikacja sprzętu użytego do badań.....	13
6.2. Parametry	13
6.3. Mierzenie czasu	14
6.4. Pliki wyjściowe	14
6.5. SWAP_LOSOWY_GEO.....	14
6.6. 2OPT_LOSOWO_GEO.....	17
6.7. SWAP_ZACHŁANNY_GEO.....	20
6.8. 2OPT_ZACHŁANNY_GEO.....	23

6.9. SWAP_LOSOWY_LOG.....	26
6.10. 2OPT_LOSOWY_LOG	29
6.11. SWAP_ZACHŁANNY_LOG	32
6.12. 2OPT_ZACHŁANNY_LOG.....	35
6.13. Porównanie kombinacji SA	38
6.14. Porównanie z innymi metodami rozwiązywania TSP	40
7. Analiza i wnioski	41
7.1. Analiza czasu działania metody.....	41
7.2. Analiza stosunku otrzymanych najlepszych rozwiązań, a rozwiązania optymalnego.....	41
7.3. Analiza porównania SA z innymi metodami	41

1. Sformułowanie zadania

Zadanie polegało na opracowaniu, implementacji i zbadaniu efektywności metody symulowanego wyżarzania dla problemu Komiwojażera w wersji optymalizacyjnej. Rozwiązując zadanie należy zbadać wpływ:

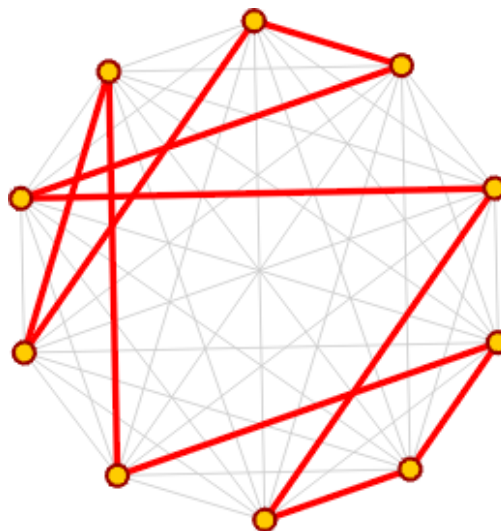
- sposobu wyboru temperatury początkowej T_0 ,
- sposobu wyboru rozwiązania początkowego x^0 ,
- schematu chłodzenia logarytmicznego, liniowego i geometrycznego,
- długości epoki,
- sposobu wyboru rozwiązania w sąsiedztwie rozwiązania bieżącego 2-opt i swap,

na

- jakość uzyskiwanych rozwiązań (wielkość błędu),
- czas uzyskiwania rozwiązań oraz
- zużycie pamięci

1.1. Problem Komiwojażera

Problem komiwojażera, znany również jako problem podróżującego sprzedawcy (TSP – Traveling Salesman Problem), to klasyczne zagadnienie z dziedziny optymalizacji kombinatorycznej i teorii grafów. Polega na znalezieniu najkrótszej możliwej trasy, która odwiedza każde miasto z danego zestawu dokładnie raz i powraca do punktu początkowego - poszukuje się takiego cyklu Hamiltona¹, który ma najmniejszą możliwą sumę wag krawędzi. Kluczowym aspektem jest to, że każde rozwiązanie problemu komiwojażera jest cyklem Hamiltona, ale nie każdy cykl Hamiltona jest rozwiązaniem problemu komiwojażera, ponieważ rozwiązanie musi także minimalizować całkowity dystans lub koszt. Problem ten jest NP-trudny, co oznacza, że nie istnieje znany algorytm, który mógłby rozwiązać wszystkie przypadki tego problemu w czasie wielomianowym.



Rysunek 1. Przykładowy cykl Hamiltona pokazany na grafie pełnym 10-wierzchołkowym

¹ Cykl Hamiltona to ścieżka w grafie, która odwiedza każdy wierzchołek dokładnie raz i kończy się w punkcie początkowym

2. Metoda

Metoda symulowanego wyżarzania (ang. simulated annealing) jest techniką optymalizacji, która naśladuje proces wyżarzania w metalurgii, polegający na powolnym ochładzaniu materiału w celu zmniejszenia defektów strukturalnych. W kontekście problemu komiwojażera, metoda ta jest używana do znalezienia jak najkrótszej ścieżki, która odwiedza każde miasto raz i wraca do punktu początkowego, minimalizując przy tym całkowitą długość trasy.

Metoda symulowanego wyżarzania jest skuteczna w poszukiwaniu globalnego minimum w problemach optymalizacyjnych, takich jak TSP, ponieważ pozwala na uniknięcie utknięcia w lokalnych minimach, dzięki możliwości akceptowania gorszych rozwiązań na wczesnych etapach algorytmu. Dzięki temu algorytm ma szansę zbadać większą przestrzeń rozwiązań i zwiększyć prawdopodobieństwo znalezienia rozwiązania optymalnego lub bliskiego optymalnemu.

2.1. Wybór rozwiązania początkowego

Wybór rozwiązania początkowego w symulowanym wyżarzaniu dla problemu komiwojażera jest istotny, ponieważ stanowi punkt wyjścia dla procesu optymalizacji. Istnieją różne metody wyboru tego rozwiązania, które mogą mieć wpływ na efektywność algorytmu. Zastosowane metody to algorytm zachłanny oraz metoda losowa.

Metoda losowa jest prosta w implementacji i zapewnia algorytmowi SA możliwość rozpoczęcia z różnorodnych miejsc w przestrzeni rozwiązań. Daje to większe szanse na eksplorację obszarów, które mogą być pomijane przez bardziej deterministyczne metody. Nie istnieje ryzyko natychmiastowego wpadnięcia w lokalne minimum, co jest zaletą w przypadku złożonych przestrzeni rozwiązań.

Metoda zachłanna, choć szybko dostarcza dobre rozwiązanie, ma swoje wady. Może prowadzić do lokalnego minimum, z którego trzeba „wyjść” w początkowych etapach działania SA. Metoda wyżarzania co prawda pozwala na akceptację gorszych rozwiązań, szczególnie przy wyższych temperaturach, ale rozpoczęcie od rozwiązania bliskiego lokalnemu optimum może ograniczyć eksplorację przestrzeni rozwiązań.

2.2. Wybór temperatury początkowej

Wybór odpowiedniej temperatury początkowej jest kluczowym elementem w procesie symulowanego wyżarzania. Temperatura w symulowanym wyżarzaniu kontroluje prawdopodobieństwo akceptacji gorszych rozwiązań w początkowych etapach działania metody, co pozwala na eksplorację szerszej przestrzeni rozwiązań jednocześnie unikając lokalnych minimów.

Implementując metodę symulowanego wyżarzania użyto metody estymującej wybór temperatury początkowej:

$$\rho = e^{\frac{\Delta E}{T_0}} \rightarrow T_0 = \frac{\Delta E}{\log\left(\frac{1}{\rho}\right)}$$

Równanie 1. Równanie, z którego wyznaczano temperaturę początkową

gdzie:

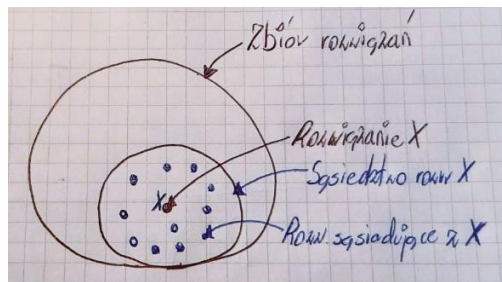
$$\Delta E = \frac{\sum_{k=1}^{100} |\text{Koszt rozw. początkowego} - \text{Koszt wygenerowanego rozw. sąsiadującego}|}{100}$$

T_0 – Temperatura początkowa

ρ – poziom prawdopodobieństwa, akceptacji gorszego rozwiązania

2.3. Zwiedzanie sąsiedztwa

Zwiedzanie sąsiedztwa² w metodzie symulowanego wyżarzania odnosi się do procesu wyszukiwania nowych potencjalnych rozwiązań na podstawie obecnie rozpatrywanego rozwiązania. Zaimplementowano dwie metody przeszukiwania sąsiedztwa – metoda swap oraz 2-opt.



Rysunek 2. Przykładowa ilustracja sąsiedztwa

2.4. Metody chłodzenia

Sposób chłodzenia w metodzie symulowanego wyżarzania dla problemu komiwojażera jest mechanizmem kontrolującym tempo, w jakim algorytm eksploruje przestrzeń rozwiązań - inspirowany fizycznym procesem wyżarzania, gdzie materiał jest stopniowo ochładzany, aby zmniejszyć jego wewnętrzne naprężenia i osiągnąć bardziej stabilny stan. W kontekście SA³, chłodzenie reguluje prawdopodobieństwo akceptacji gorszych rozwiązań. Na początku, kiedy temperatura jest wysoka, algorytm może akceptować gorsze rozwiązania z większym prawdopodobieństwem, co pozwala na eksplorację szerszej przestrzeni rozwiązań i uniknięcie lokalnych minimów. W miarę obniżania temperatury prawdopodobieństwo to maleje, co sprzyja eksploatacji obiecujących obszarów przestrzeni rozwiązań, skupiając się na udoskonalaniu najlepszych znalezionych dotąd tras.

Implementując metodę symulowanego wyżarzania użyto dwóch typów chłodzenia:

- o Geometryczny - Temperatura zmniejsza się o stały czynnik (procent obecnej temperatury)

$$T_{i+1} = T_i * \alpha$$

Równanie 2. Geometryczny schemat chłodzenia

gdzie:

i – numer epoki

α – współczynnik chłodzenia, $\alpha \in (0; 1)$

- o Logarytmiczny - Spadek temperatury następuje wolniej z każdą iteracją, zgodnie z funkcją logarytmiczną.

$$T_{i+1} = \frac{T_i}{1 + \log(1 + i)}$$

Równanie 3. Logarytmiczny schemat chłodzenia

gdzie:

i – numer epoki

² Sąsiedztwo, czyli otoczenie rozwiązania aktualnego oraz sąsiadów, czyli rozwiązania "bliskie" rozwiązaniu aktualnemu – Wykład1 Rzeczywistość i Model

³ SA – Symulowane wyżarzanie, ang. Simulated Annealing

3. Algorytm i sposoby implementacji

Kluczowe kroki metody symulowanego wyżarzania użyte do rozwiązania problemu TSP:

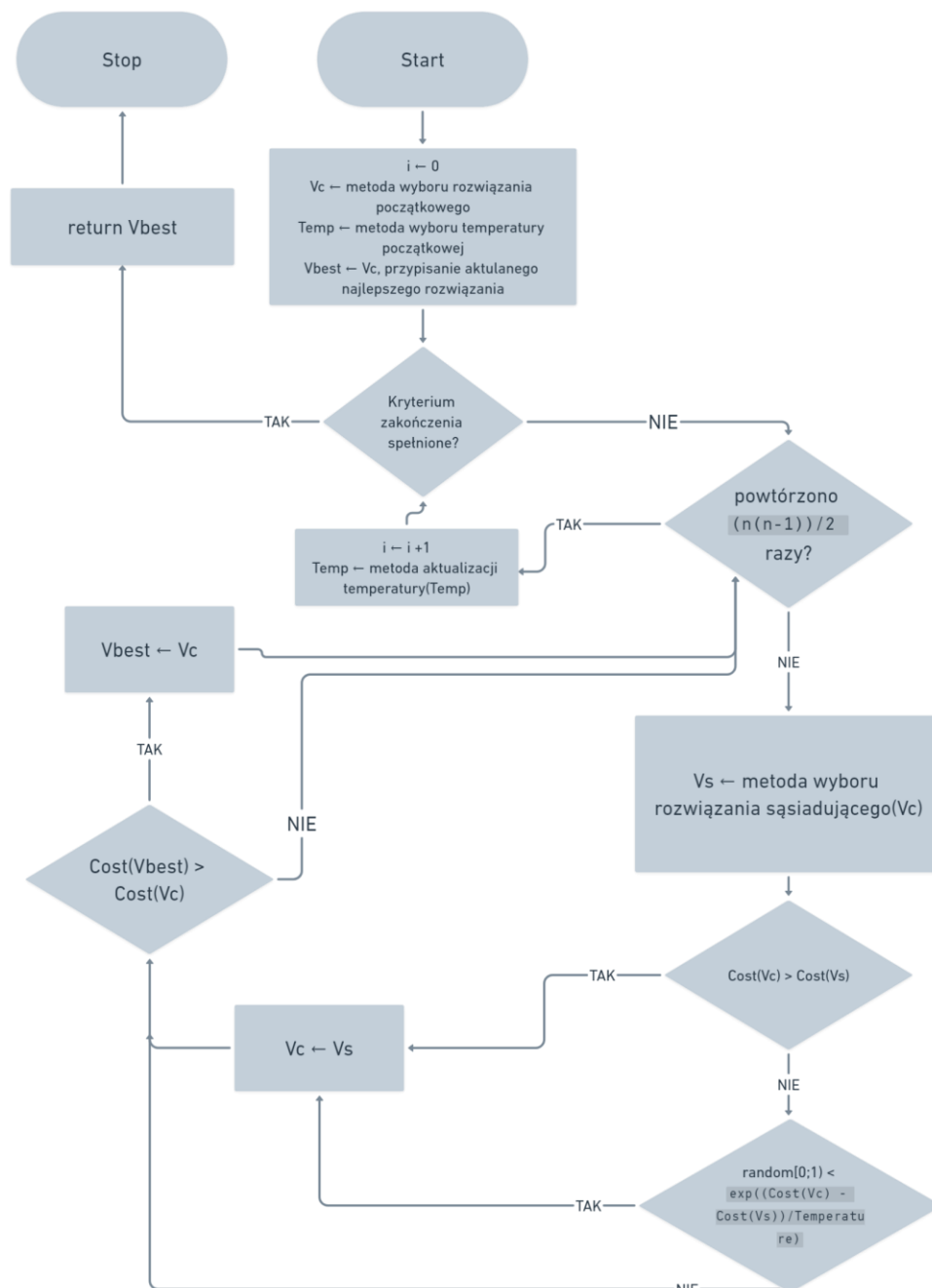
1. Inicjalizacja – Wybranie początkowej trasy (np. losowo) i ustawienie początkowej "temperatury" algorytmu.
2. Iteracje – W każdej iteracji dokonuje się losowej modyfikacji trasy (np. zmieniając kolejność odwiedzanych miast) i oblicza różnicę kosztu (ΔE) między nową a obecną trasą.
3. Akceptacja zmian – Jeśli nowa trasa jest krótsza (ma niższy koszt), jest akceptowana. Jeśli jest dłuższa, istnieje nadal pewne prawdopodobieństwo jej akceptacji, które zależy od różnicy kosztów i obecnej temperatury algorytmu. To prawdopodobieństwo jest wyższe przy wyższych temperaturach i maleje w miarę ochładzania systemu.
4. Schładzanie – Temperatura systemu jest stopniowo obniżana zgodnie z zaplanowanym harmonogramem schładzania.
5. Kryterium zakończenia – Proces jest kontynuowany do momentu spełnienia kryterium zakończenia, na przykład osiągnięcia określonej minimalnej temperatury lub wyczerpania liczby iteracji.

3.1. Pseudo kod i schemat blokowy

```
procedure symulowane wyżarzanie
begin
   $t \leftarrow 0$ 
  inicjuj  $T$ 
  wybierz losowo bieżący punkt  $\mathbf{v}_c$ 
  oceń  $\mathbf{v}_c$ 
  repeat
    repeat
      wybierz nowy punkt  $\mathbf{v}_n$ 
        w otoczeniu  $\mathbf{v}_c$ 
      if  $eval(\mathbf{v}_c) < eval(\mathbf{v}_n)$ 
        then  $\mathbf{v}_c \leftarrow \mathbf{v}_n$ 
      else if  $random[0, 1) < e^{\frac{eval(\mathbf{v}_n) - eval(\mathbf{v}_c)}{T}}$ 
        then  $\mathbf{v}_c \leftarrow \mathbf{v}_n$ 
    until (warunek zakończenia)
     $T \leftarrow g(T, t)$ 
     $t \leftarrow t + 1$ 
  until (kryterium zatrzymania)
end
```

Rysunek 3. Pseudokod symulowanego wyżarzania⁴

⁴ Źródło – Zbigniew Michalewicz „jak to rozwiązać czyli nowoczesna heurystyka” – rozdział 5.1 str.150



Rysunek 4. Schemat blokowy symulowanego wyżarzania

3.2. Rozwiązanie początkowe

Stosowany algorytm zachłanny do wygenerowania początkowego rozwiązania dla TSP:

1. Wybór punktu startowego - Algorytm zaczyna od wybranego miasta lub losowo wybranego punktu startowego.
2. Wybór najbliższego sąsiada - Z obecnego miasta, algorytm szuka najbliższego sąsiada, czyli miasta, które jest najbliżej obecnej lokalizacji i jeszcze nie zostało odwiedzone.
3. Przejście do wybranego miasta - Algorytm przenosi się do wybranego miasta i oznacza je jako odwiedzone.
4. Powtórzenie procesu - Proces wyboru najbliższego sąsiada i przechodzenia do niego powtarza się, aż wszystkie miasta zostaną odwiedzone.

5. Powrót do punktu startowego - Po odwiedzeniu wszystkich miast, algorytm wraca do punktu startowego, kończąc tym samym cykl.

Kroki metody losowej zastosowane do znalezienia rozwiązania początkowego dla TSP:

1. Wybór punktu startowego - Wybierz losowo miasto startowe z listy wszystkich miast. To miasto będzie punktem początkowym i końcowym trasy.
2. Tworzenie listy miast - Utwórz listę wszystkich miast, które muszą być odwiedzone, z wyjątkiem miasta startowego.
3. Losowanie kolejności odwiedzin - Losowo permutuj listę miast, aby stworzyć kolejność, w jakiej będą one odwiedzane. Każde miasto na liście jest wybierane dokładnie raz, co zapewnia, że każde miasto zostanie odwiedzone.
4. Utworzenie trasy - Rozpocznij trasę od miasta startowego, następnie przejdź do pierwszego miasta na losowo uporządkowanej liście, kontynuuj podróż do każdego kolejnego miasta w kolejności wyznaczonej przez losowanie, aż odwiedzisz wszystkie miasta.
5. Zakończenie trasy - Po odwiedzeniu wszystkich miast na liście, wróć do miasta startowego, aby zamknąć cykl i uformować pełną trasę.

3.3. Wybór temperatury początkowej

Działanie metody estymującej w krokach:

1. Wybór rozwiązania początkowego – Wybierz rozwiązanie początkowe metodą losową bądź zachłanną.
2. Znalezienie rozwiązania sąsiedniego – Używając metody zwiedzania sąsiedztwa swap wygeneruj rozwiązanie sąsiednie do początkowego.
3. Obliczenie długości sąsiedniej ścieżki – Dla wygenerowanej ścieżki sąsiedniej oblicz jej długość.
4. Sumowanie różnic – Oblicz wartość bezwzględną z różnicy długości ścieżki początkowej i ścieżki sąsiedniej, następnie wartość dodaj do kontenera pamięci.
5. Powtórzenie procesu – Proces szukania rozwiązania sąsiedniego, wyznaczenia jego drogi oraz obliczenia różnicy wartości dróg powtórz sto razy.
6. Wyznaczenie średniej arytmetycznej – Wyznacz średnią arytmetyczną sumy wartości bezwzględnych różnic dróg ΔE .
7. Określenie atrybutu akceptacji – Określ wartość atrybutu poziomu prawdopodobieństwa, akceptacji gorszego rozwiązania dla 'Równanie 1'.
8. Wyznaczenie temperatury początkowej – Oblicz temperaturę początkową używając 'Równanie 1'.

3.4. Zwiedzanie sąsiedztwa

Działanie zastosowanej metody swap w krokach:

1. Wybór dwóch wierzchołków – Losowo wybierz dwa wierzchołki w obecnej trasie.
2. Zamiana wierzchołków – Zamień miejscami położenie dwóch wierzchołków w obecnej trasie.

Działanie metody 2-opt w krokach:

1. Losowanie indeksów – Losowo wybierz dwa różne indeksy 'i' i 'k' z zakresu długości ścieżki. Powtarzaj losowanie, aż 'k' będzie większe niż 'i'.

2. Zamiana ścieżek – Odwróć kolejność miast w ścieżce między indeksami 'i' i 'k'. Oznacza to, że segment trasy od miasta 'i' do miasta 'k' jest odwracany.
3. Zwróć trasę – Zwróć otrzymaną tym sposobem sąsiednią ścieżkę.

3.5. Schemat chłodzenia

Działanie metody schładzania geometrycznego:

1. Aktualizacja temperatury – Aktualną temperaturę wyżarzania pomnóż przez współczynnik schładzania – Równanie 2.
2. Zwróć iloczyn – Zwróć wartość obliczoną w poprzednim kroku działania metody.

Działanie metody schładzania logarytmicznego

1. Aktualizacja temperatury – Oblicz temperaturę dla następnej epoki korzystając ze wzoru – Równanie 3.
2. Zwróć iloczyn – Zwróć wartość obliczoną w poprzednim kroku działania metody.

4. Dane testowe

Do sprawdzenia poprawności działania algorytmu wybrano następujący zestaw instancji:

[nazwa_pliku; adres_źródła]

1. tsp_6_1.txt,
2. tsp_10.txt,
3. tsp_17_1.txt,
4. tsp_19_1.txt,
5. tsp_20_1.txt; <http://jaroslaw.mierzwa.staff.iiar.pwr.wroc.pl/pea-stud/tsp/>
6. berlin52.tsp,
7. ftv55.tsp
8. ftv170.tsp; <http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/>

4.1. Symetryczne

Do wykonania badań na grafach symetrycznych wybrano następujący zestaw instancji:

[nazwa_pliku; adres_źródła]

1. gr17.tsp
2. gr21.tsp
3. gr24.tsp
4. bays29.tsp
5. eil51.tsp
6. berlin52.tsp
7. st70.tsp
8. kroA100.tsp
9. gr120.tsp
10. kroB150.tsp
11. kroB200.tsp
12. pr226.tsp
13. pr264.tsp
14. pr299.tsp; <http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/tsp/>

4.2. Asymetryczne

Do wykonania badań na grafach asymetrycznych wybrano następujący zestaw instancji:

[nazwa_pliku; adres_źródła]

1. tsp_6_1.txt ; <http://jaroslaw.mierzwa.staff.iiar.pwr.wroc.pl/pea-stud/tsp/>
2. br17.atasp
3. ftv33.atasp
4. ftv35.atasp
5. frv38.atasp
6. p43.atasp
7. ft53.atasp
8. ftv55.atasp
9. ftv70.atasp
10. kro124p.atasp
11. ftv170.atasp
12. rbg323.atasp
13. rbg358.atasp
14. rbg403.atasp
15. rbg443.atasp ; <http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/atasp/>

5. Procedura badawcza

W rozwiązaniu należało zbadać wpływ:

- sposobu wyboru temperatury początkowej T_0
- sposobu wyboru rozwiązania początkowego x^0
- schematu chłodzenia (logarytmicznego i geometrycznego)
- długości epoki
- sposobu wyboru rozwiązania w sąsiedztwie rozwiązania bieżącego (2-opt i swap)

na

- jakość uzyskiwanych rozwiązań (wielkość błędu)
- czas uzyskiwania rozwiązań
- zużycie pamięci

Procedura badawcza polegała na uruchomieniu programu sterowanego plikiem inicjującym .INI, dla różnych kombinacji metod wyboru rozwiązania początkowego, chłodzenia, zwiedzania sąsiedztwa.

Tabela 1. Kombinacje metod użytych w badaniu

Wybór rozwiązania początkowego		Schemat chłodzenia		Metoda zwiedzania sąsiedztwa	
Losowy	Zachłanny	Geometryczny	Logarytmiczny	Swap	2-Opt
x		x		x	
x		x			x
	x	x		x	
	x	x			x
x			x	x	
x			x		x
	x		x	x	
	x		x		x

Format pliku sterującego - [nazwa_instancji liczba_wykonan rozwiązanie_optymalne; nazwa_pliku_wyjściowego]

- Plik sterujący dla instancji symetrycznych:

gr17.tsp 5 2085

gr21.tsp 5 2707

gr24.tsp 5 1272

bays29.tsp 5 2020

eil51.tsp 5 426

berlin52.tsp 5 7542

st70.tsp 5 675

kroA100.tsp 3 21282

gr120.tsp 3 6942

kroB150.tsp 3 26130

kroB200.tsp 3 29437

pr226.tsp 1 80369

pr264.tsp 1 49135

pr299.tsp 1 48191

outputtsp.csv

- Plik sterujący dla instancji asymetrycznych:

tsp_6_1.txt 5 132

br17.atsp 5 39

ftv33.atsp 5 1286

ftv35.atsp 5 1473

ftv38.atsp 5 1530

p43.atsp 5 5620

ftv44.atsp 5 1613

ft53.atsp 5 6905

ftv55.atsp 5 1608

ftv70.atsp 5 1950

kro124p.atsp 3 36230

ftv170.atsp 3 2755

rbg323.atsp 1 1326

rbg358.atsp 1 1163

rbg403.atsp 1 2465

rbg443.atsp 1 2720

output.csv

Każda instancji rozwiązywana była zgodnie z liczbą jej wykonan, np. br17.atsp wykonana została 5 razy. Do pliku wyjściowego output.csv zapisywana była - nazwa instancji, średni koszt najlepszych rozwiązań, średni czas[ms], średnie zużycie pamięci, rozmiar instancji, optymalny koszt, procent błędu.

Plik wyjściowy zapisywany był w formacie csv. Poniżej przedstawiono fragment zawartości pliku wyjściowego.

[Nazwa instancji, Średni koszt najlepszych rozwiązań, Średni czas[ms], Średnie zużycie pamięci, Rozmiar instancji, **Optymalny koszt, Procent błędu**]

gr17.tsp,2085,1814,0,17,**2085,0**,

gr21.tsp,2707,2819,0,21,**2707,0**,

gr24.tsp,1272,3399,0,24,**1272,0**,
 bays29.tsp,2022,5216,0,29,**2020,0.0990099**,
 eil51.tsp,428.982,13841,0,51,**426,0.699927**,
 berlin52.tsp,7544.37,18794,0,52,**7542,0.0313738**,
 st70.tsp,683.557,30208,0,70,**675,1.26774**,
 kroA100.tsp,21393.9,79316,0,100,**21282,0.525725**,
 gr120.tsp,6985,97932,0,120,**6942,0.619418**,
 kroB150.tsp,26417.9,175905,0,150,**26130,1.10179**,
 kroB200.tsp,30151.4,316900,0,200,**29437,2.42675**,
 pr226.tsp,81274.9,447226,0,226,**80369,1.12718**,
 pr264.tsp,49948.2,602870,0,264,**49135,1.65511**,
 pr299.tsp,48572.1,761968,0,299,**48191,0.790823**,

Drugim plikiem wyjściowym był plik csv zawierający wartości, które zostały do niego dodane w trakcie trwania metody dla danej instancji w momencie znalezienia lepszego rozwiązania niż aktualnie najlepsze.

Plik wyjściowy ten zawiera w sobie informacje o – numerze epoki, czasie w którym znaleziono nowe najlepsze rozwiązanie, temperaturę w tym momencie, dystans nowego najlepszego rozwiązania, ścieżka.

Generowany automatycznie, jego nazwa jest taka sama jak nazwa instancji.

[Numer epoki, Czas[ms], Temp, Dystans, Ścieżka] – tutaj tsp_6_1.csv

0,0,21.626,203,<1 4 5 2 0 3 >
 0,0,21.626,179,<1 4 5 2 3 0 >
 0,0,21.626,158,<1 2 3 5 4 0 >
 3,0,21.3032,156,<1 3 4 5 0 2 >
 8,0,20.7759,132,<1 2 3 4 5 0 >

6. Badania i wyniki

6.1. Specyfikacja sprzętu użytego do badań

System : Ubuntu 22.04.3 LTS

Procesor : Intel I7-8665U

RAM : 16GM

6.2. Parametry

Na cały okres trwania badania przyjęto jedne parametry startowe tj.:

- Prawdopodobieństwo z Równanie 1 = 0.5
- Tempo chłodzenia = 0.995

Przyjęte kryteria zakończenia działania metody:

- o Czas trwania przekroczył 60 min
- o Temperatura spadła do poziomu $1e-3$

6.3. Mierzenie czasu

Do mierzenia czasu wykonania jednego powtórzenia metody SA została użyta biblioteka `chrono` w C++.

```
beforeTime = std::chrono::high_resolution_clock::now();
```

```
RunSimulatedAnnealing();
```

```
afterTime = std::chrono::high_resolution_clock::now();
```

```
worktime = std::chrono::duration_cast<std::chrono::milliseconds>(after-time).count();
```

6.4. Pliki wyjściowe

Wyniki zgromadzone zostały w folderach na dysku Google pod adresem; <https://drive.google.com/drive/folders/1z06evlZHqnSb7PRL7zYSg02HvZHNHNCB?usp=sharing>

Format nazw folderów:

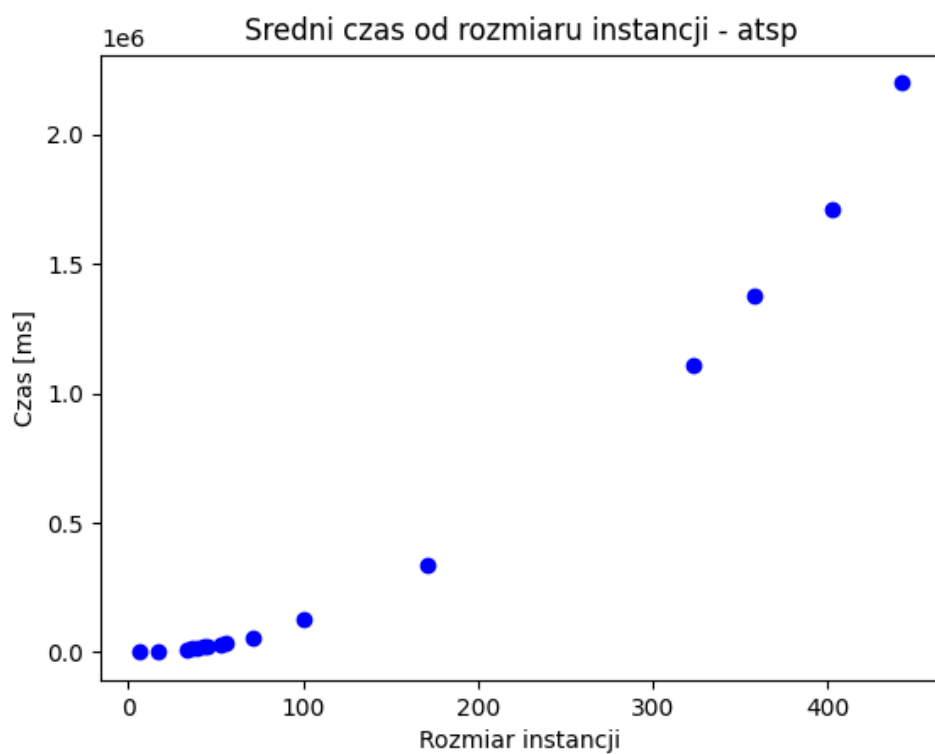
[Instancje symetryczne/asymetryczne - metoda zwiedzania sąsiedztwa, metoda wyboru rozwiązania początkowego, metoda schładzania]

ATSP – 2OPT_LOSOWO_GEO

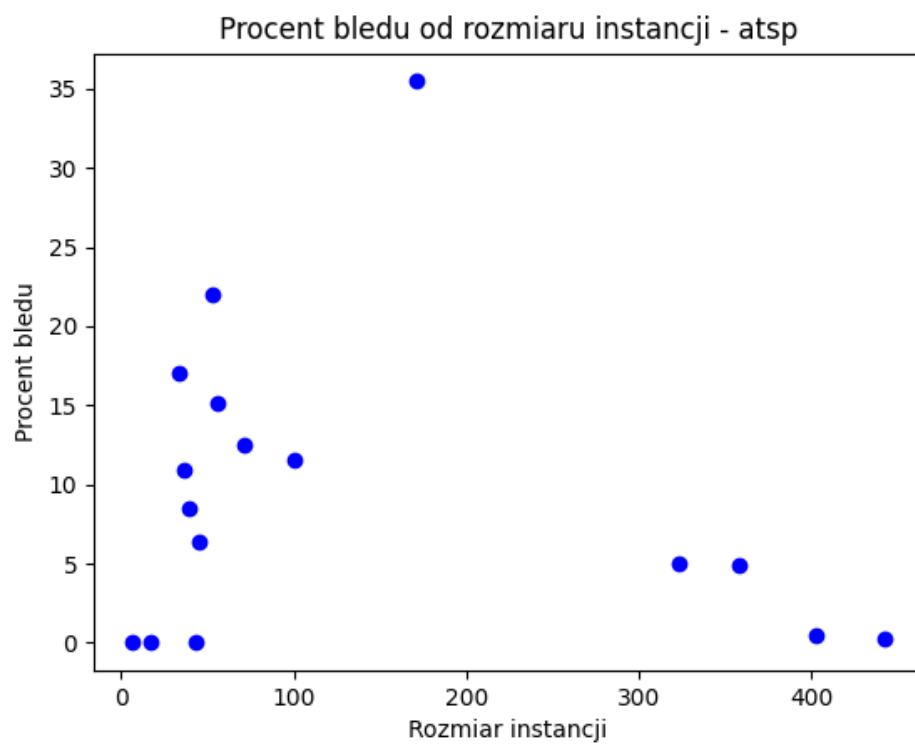
6.5. SWAP_LOSOWY_GEO

Tabela 2. Tabela wyników ATSP - SWAP_LOSOWY_GEO

Nazwa instancji	Średni koszt najlepszych rozwiązań	Średni czas[ms]	Rozmiar instancji	Optymalny koszt	Procent błędu
tsp_6_1.txt	132	305	6	132	0
br17.atsp	39	2864	17	39	0
ftv33.atsp	1505	11965	34	1286	17.0295
ftv35.atsp	1634	13696	36	1473	10.9301
ftv38.atsp	1660	15680	39	1530	8.49673
p43.atsp	5625	19815	43	5620	0.088968
ftv44.atsp	1715	21964	45	1613	6.32362
ft53.atsp	8422	31304	53	6905	21.9696
ftv55.atsp	1851	32906	56	1608	15.1119
ftv70.atsp	2194	53236	71	1950	12.5128
kro124p.atsp	40400	124345	100	36230	11.5098
ftv170.atsp	3732	337286	171	2755	35.4628
rbg323.atsp	1393	1107528	323	1326	5.05279
rbg358.atsp	1220	1376516	358	1163	4.90112
rbg403.atsp	2477	1711060	403	2465	0.486815
rbg443.atsp	2726	2199788	443	2720	0.220588



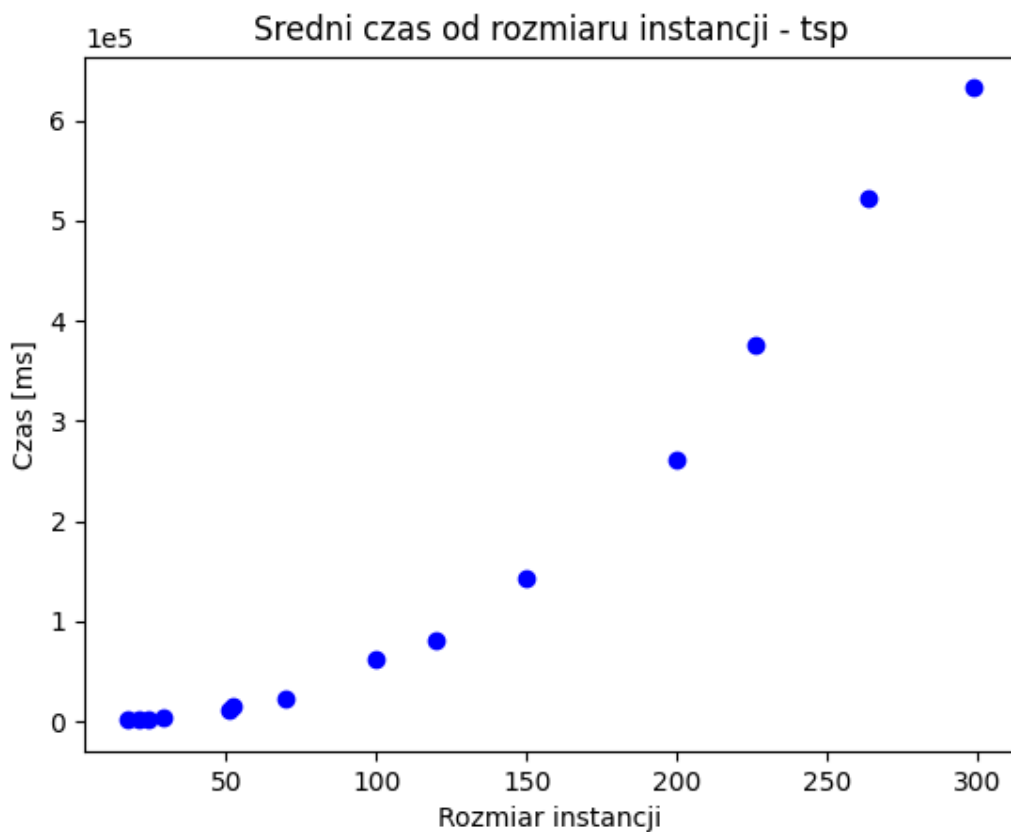
Rysunek 5. Wykres zależności czasu od wielkości instancji dla Tabela 2



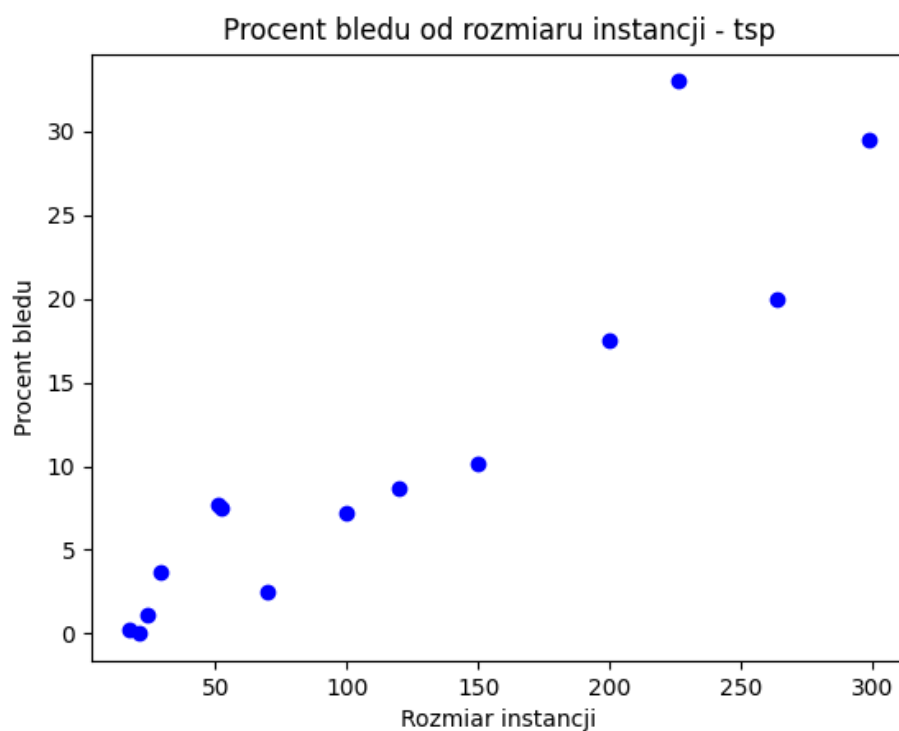
Rysunek 6. Wykres zależności błędu w porównaniu do optymalnego rozwiązania od wielkości instancji dla Tabela 2

Tabela 3. Tabela wyników TSP - SWAP_LOSOWY_GEO

Nazwa instancji	Średni koszt najlepszych rozwiązań	Średni czas[ms]	Rozmiar instancji	Optymalny koszt	Procent błędu
gr17.tsp	2090	1431	17	2085	0.239808
gr21.tsp	2707	2276	21	2707	0
gr24.tsp	1286	2772	24	1272	1.10063
bays29.tsp	2095	4199	29	2020	3.71287
eil51.tsp	458.842	10821	51	426	7.70938
berlin52.tsp	8109.58	15052	52	7542	7.52564
st70.tsp	692.086	21942	70	675	2.53128
kroA100.tsp	22821.8	63083	100	21282	7.23502
gr120.tsp	7544	81519	120	6942	8.67185
kroB150.tsp	28769	143917	150	26130	10.0994
kroB200.tsp	34589.7	260925	200	29437	17.5042
pr226.tsp	106871	376179	226	80369	32.9754
pr264.tsp	58920.3	523064	264	49135	19.9152
pr299.tsp	62381.3	632355	299	48191	29.446



Rysunek 7. Wykres złożoności czasu od wielkości instancji dla Tabela 3

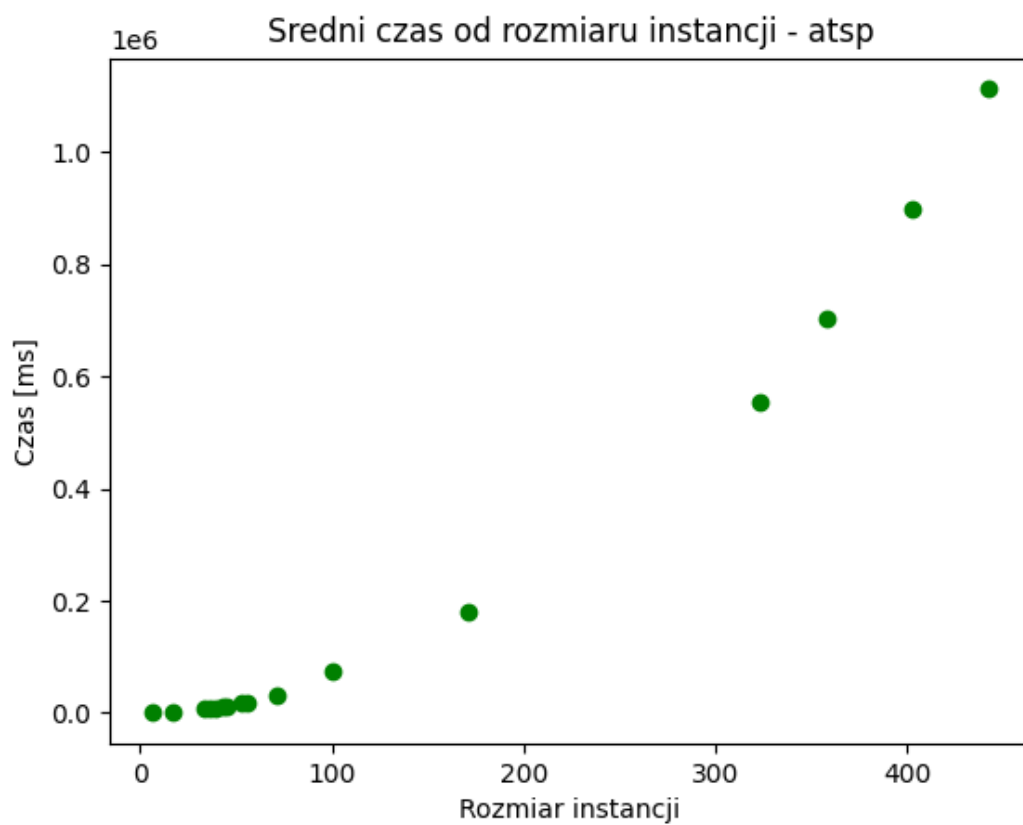


Rysunek 8. Wykres zależności błędu w porównaniu do optymalnego rozwiązania od wielkości instancji dla Tabela 3

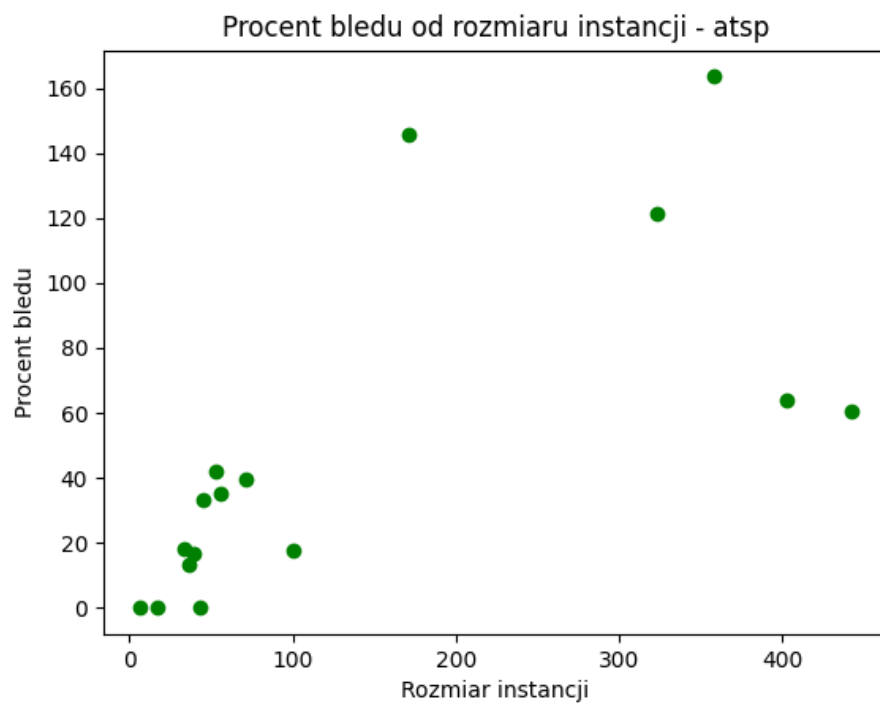
6.6. 2OPT_LOSOWO_GEO

Tabela 4. Tabela wyników ATSP – 2OPT_LOSOWY_GEO

Nazwa instancji	Średni koszt najlepszych rozwiązań	Średni czas[ms]	Rozmiar instancji	Optymalny koszt	Procent błędu
tsp_6_1.txt	132	164	6	132	0
br17.atsp	39	1448	17	39	0
ftv33.atsp	1518	6743	34	1286	18.0404
ftv35.atsp	1667	7532	36	1473	13.1704
ftv38.atsp	1789	8777	39	1530	16.9281
p43.atsp	5624	12161	43	5620	0.0711744
ftv44.atsp	2150	11854	45	1613	33.292
ft53.atsp	9824	18235	53	6905	42.2737
ftv55.atsp	2175	18775	56	1608	35.2612
ftv70.atsp	2723	30398	71	1950	39.641
kro124p.atsp	42654	74065	100	36230	17.7312
ftv170.atsp	6765	180621	171	2755	145.554
rbg323.atsp	2932	555327	323	1326	121.116
rbg358.atsp	3064	703735	358	1163	163.457
rbg403.atsp	4038	898675	403	2465	63.8134
rbg443.atsp	4368	1112134	443	2720	60.5882



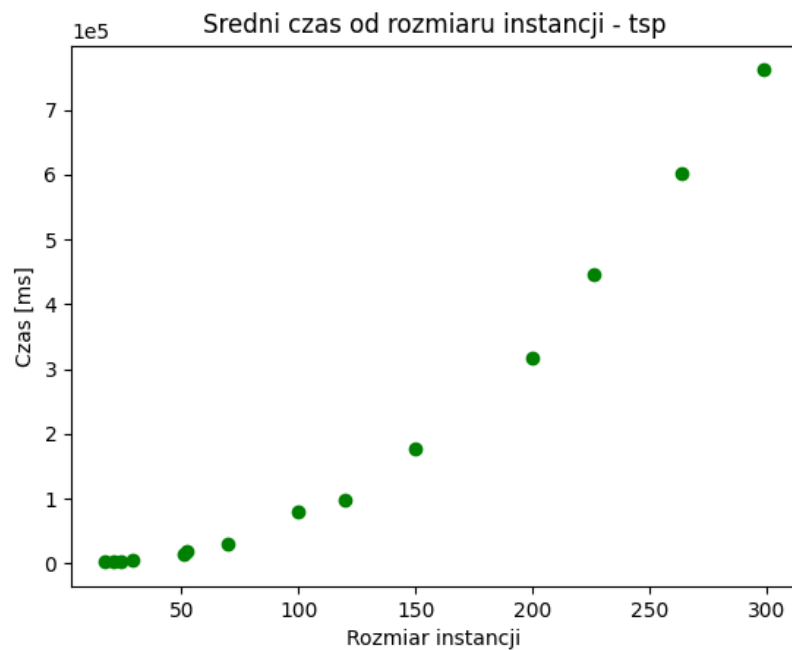
Rysunek 9. Wykres złożoności czasu od wielkości instancji dla Tabela 4



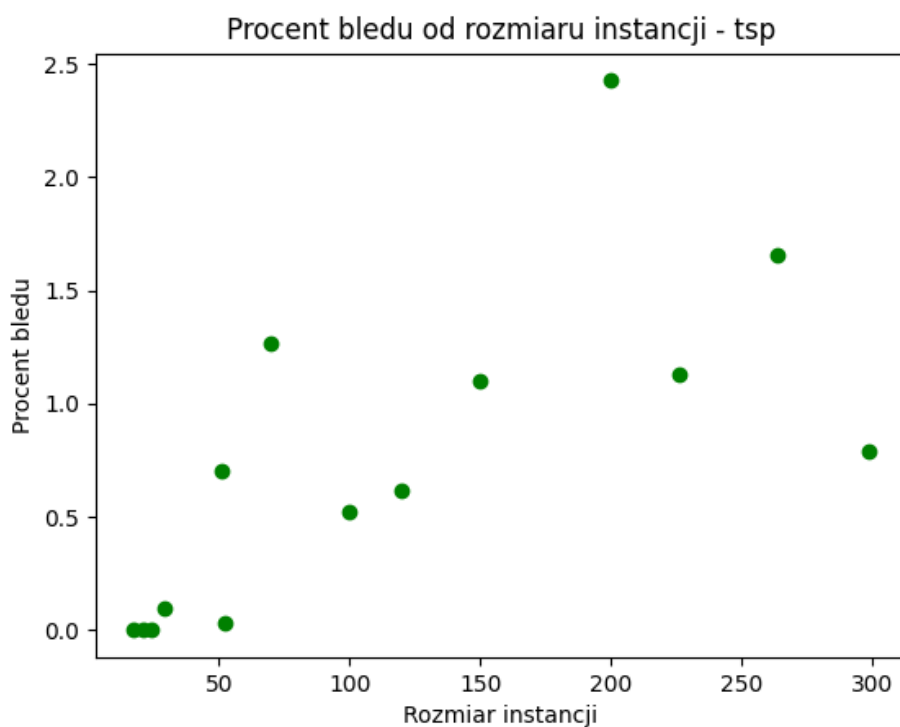
Rysunek 10. Wykres zależności błędu w porównaniu do optymalnego rozwiązania od wielkości instancji dla Tabela 4

Tabela 5. Tabela wyników TSP – 2OPT_LOSOWY_GEO

Nazwa instancji	Średni koszt najlepszych rozwiązań	Średni czas[ms]	Rozmiar instancji	Optymalny koszt	Procent błędu
gr17.tsp	2085	1814	17	2085	0
gr21.tsp	2707	2819	21	2707	0
gr24.tsp	1272	3399	24	1272	0
bays29.tsp	2022	5216	29	2020	0.0990099
eil51.tsp	428.982	13841	51	426	0.699927
berlin52.tsp	7544.37	18794	52	7542	0.0313738
st70.tsp	683.557	30208	70	675	1.26774
kroA100.tsp	21393.9	79316	100	21282	0.525725
gr120.tsp	6985	97932	120	6942	0.619418
kroB150.tsp	26417.9	175905	150	26130	1.10179
kroB200.tsp	30151.4	316900	200	29437	2.42675
pr226.tsp	81274.9	447226	226	80369	1.12718
pr264.tsp	49948.2	602870	264	49135	1.65511
pr299.tsp	48572.1	761968	299	48191	0.790823



Rysunek 11. Wykres złożoności czasu od wielkości instancji dla Tabela 5

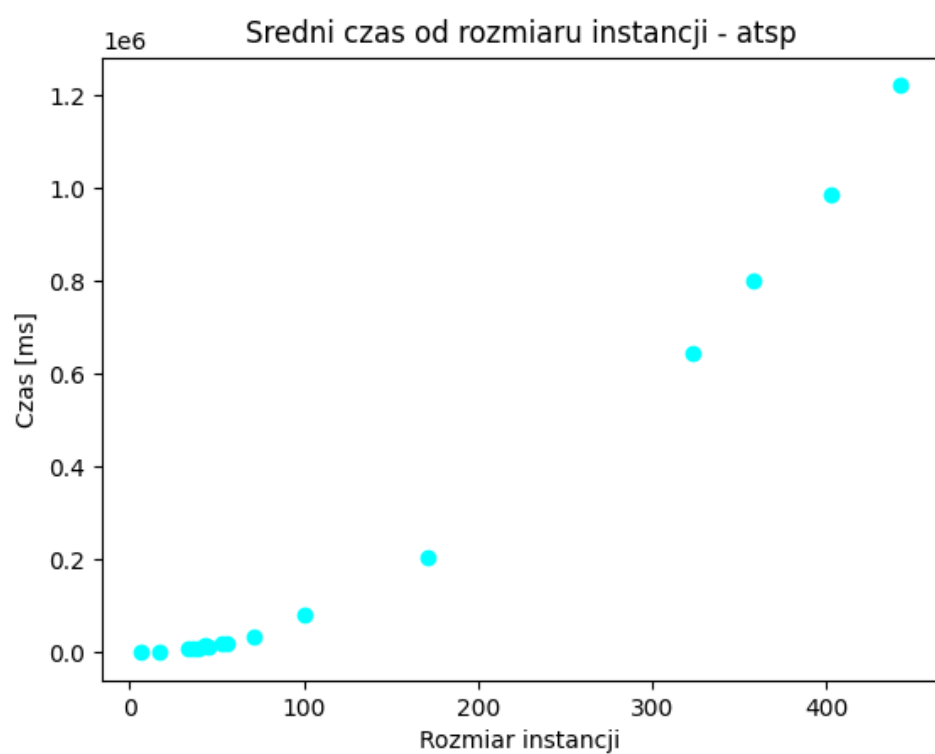


Rysunek 12. Wykres zależności błędu w porównaniu do optymalnego rozwiązania od wielkości instancji dla Tabela 5

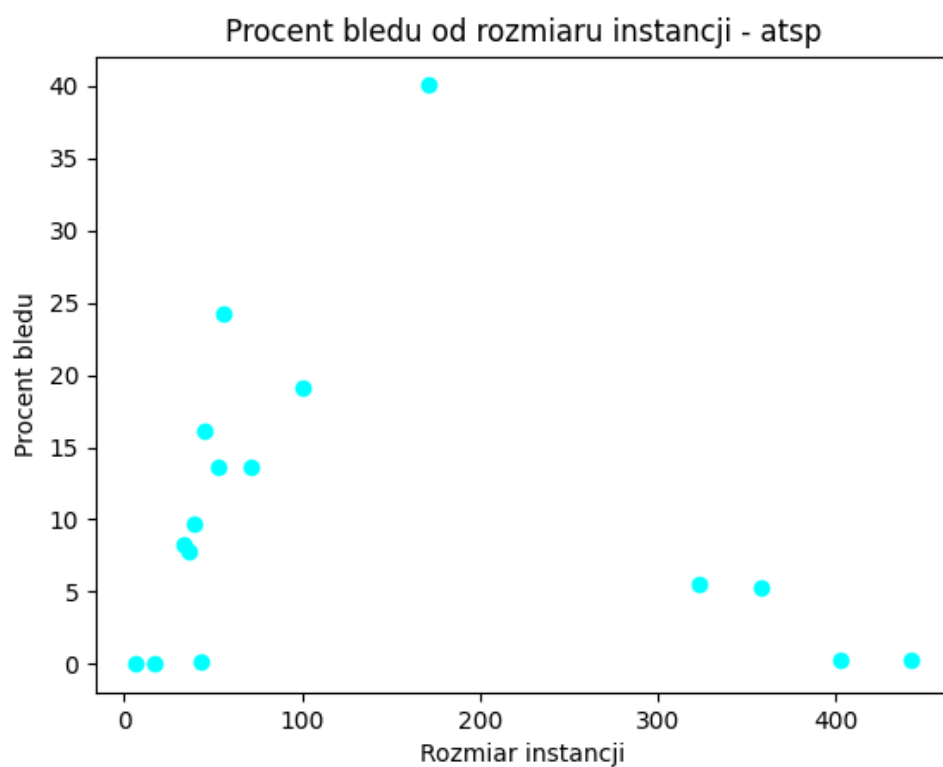
6.7. SWAP_ZACHŁANNY_GEO

Tabela 6. Tabela wyników ATSP – SWAP_ZACHŁANNY_GEO

Nazwa instancji	Średni koszt najlepszych rozwiązań	Średni czas[ms]	Rozmiar instancji	Optymalny koszt	Procent błędu
tsp_6_1.txt	132	184	6	132	0
br17.atsp	39	1510	17	39	0
ftv33.atsp	1392	7361	34	1286	8.24261
ftv35.atsp	1588	8410	36	1473	0.07242
ftv38.atsp	1679	10112	39	1530	9.73856
p43.atsp	5628	14743	43	5620	0.142349
ftv44.atsp	1874	13965	45	1613	16.181
ft53.atsp	7846	20288	53	6905	13.6278
ftv55.atsp	1997	20706	56	1608	24.1915
ftv70.atsp	2215	33685	71	1950	13.5897
kro124p.atsp	43162	81376	100	36230	19.1333
ftv170.atsp	3858	204640	171	2755	40.0363
rbg323.atsp	1400	644731	323	1326	5.58069
rbg358.atsp	1225	800079	358	1163	5.33104
rbg403.atsp	2472	988258	403	2465	0.283976
rbg443.atsp	2727	1221986	443	2720	0.257353



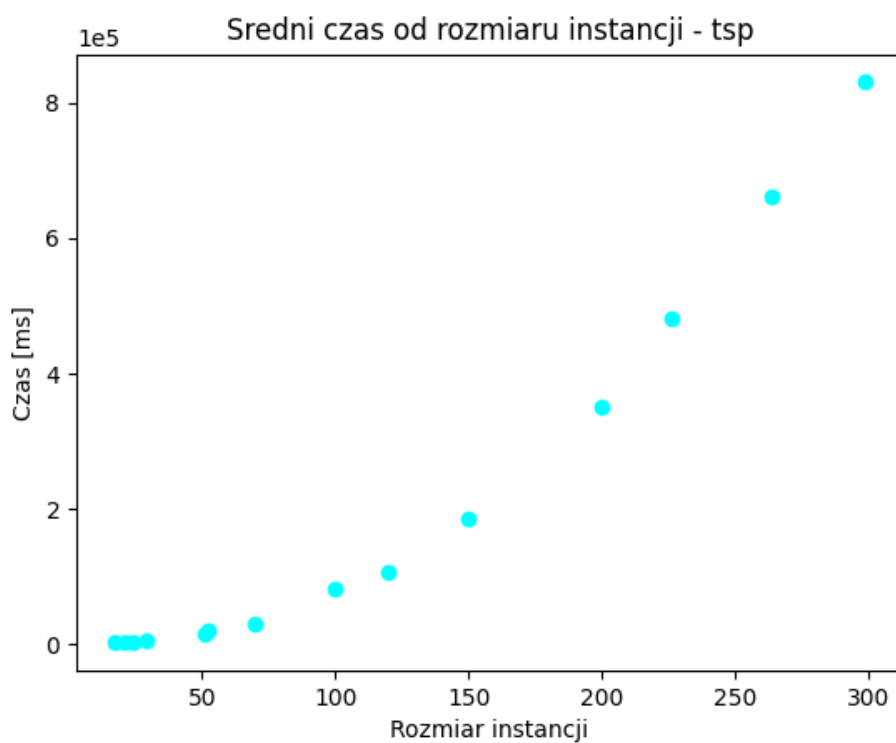
Rysunek 13. Wykres złożoności czasu od wielkości instancji dla Tabela 6



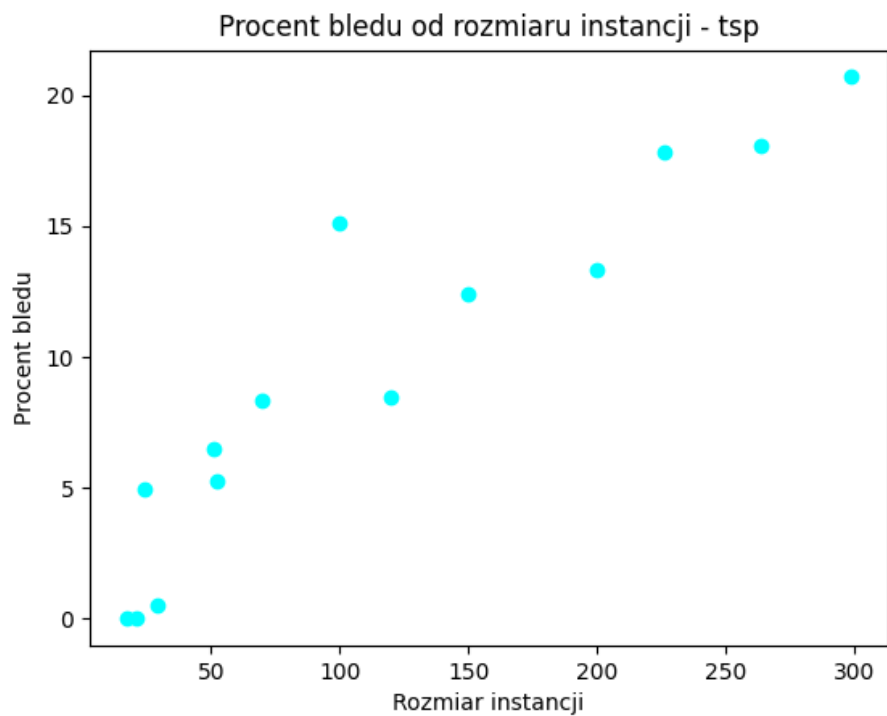
Rysunek 14. Wykres zależności błędu w porównaniu do optymalnego rozwiązania od wielkości instancji dla Tabela 6

Tabela 7. Tabela wyników TSP – SWAP_ZACHŁANNY_GEO

Nazwa instancji	Średni koszt najlepszych rozwiązań	Średni czas[ms]	Rozmiar instancji	Optymalny koszt	Procent błędu
gr17.tsp	2085	1893	17	2085	0
gr21.tsp	2707	2985	21	2707	0
gr24.tsp	1335	3654	24	1272	4.95283
bays29.tsp	2031	5569	29	2020	0.544554
eil51.tsp	453.723	15137	51	426	6.50785
berlin52.tsp	7939.27	19838	52	7542	5.26742
st70.tsp	731.36	30280	70	675	8.34959
kroA100.tsp	24502.8	80887	100	21282	15.1337
gr120.tsp	7529	106377	120	6942	8.45578
kroB150.tsp	29363.5	185813	150	26130	12.3748
kroB200.tsp	33361.8	351004	200	29437	13.333
pr226.tsp	94685.5	480495	226	80369	17.8134
pr264.tsp	58022.9	660556	264	49135	18.0886
pr299.tsp	58164.1	830766	299	48191	20.6949



Rysunek 15. Wykres złożoności czasu od wielkości instancji dla Tabela 7

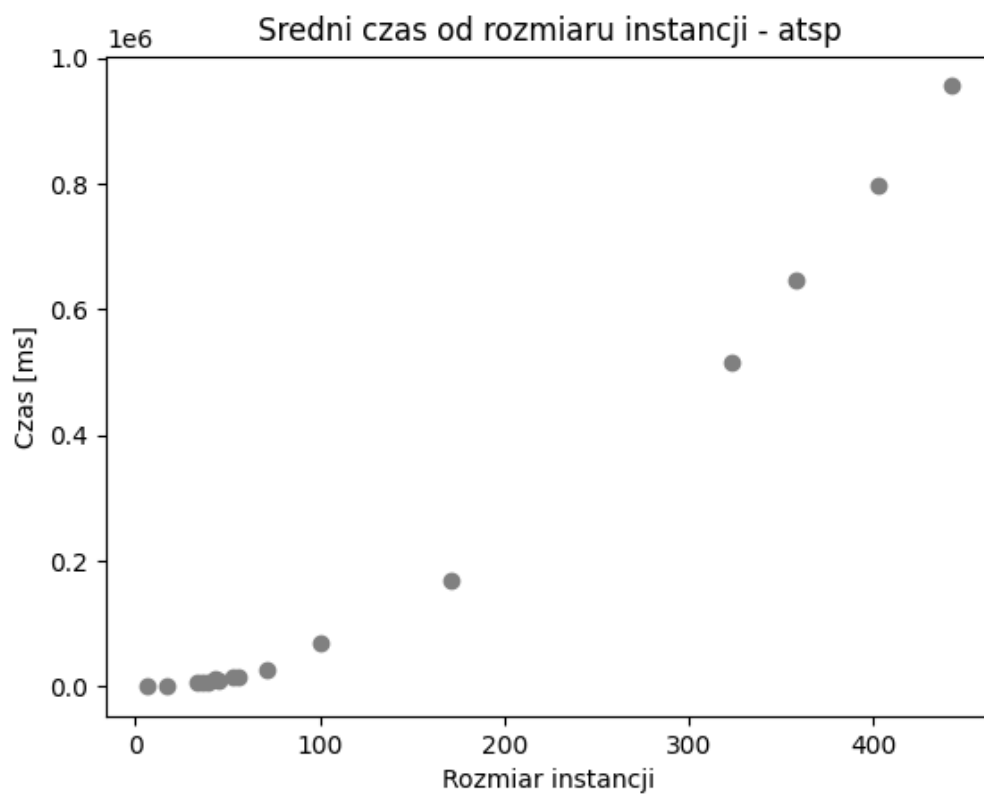


Rysunek 16. Wykres zależności błędu w porównaniu do optymalnego rozwiązania od wielkości instancji dla Tabela 7

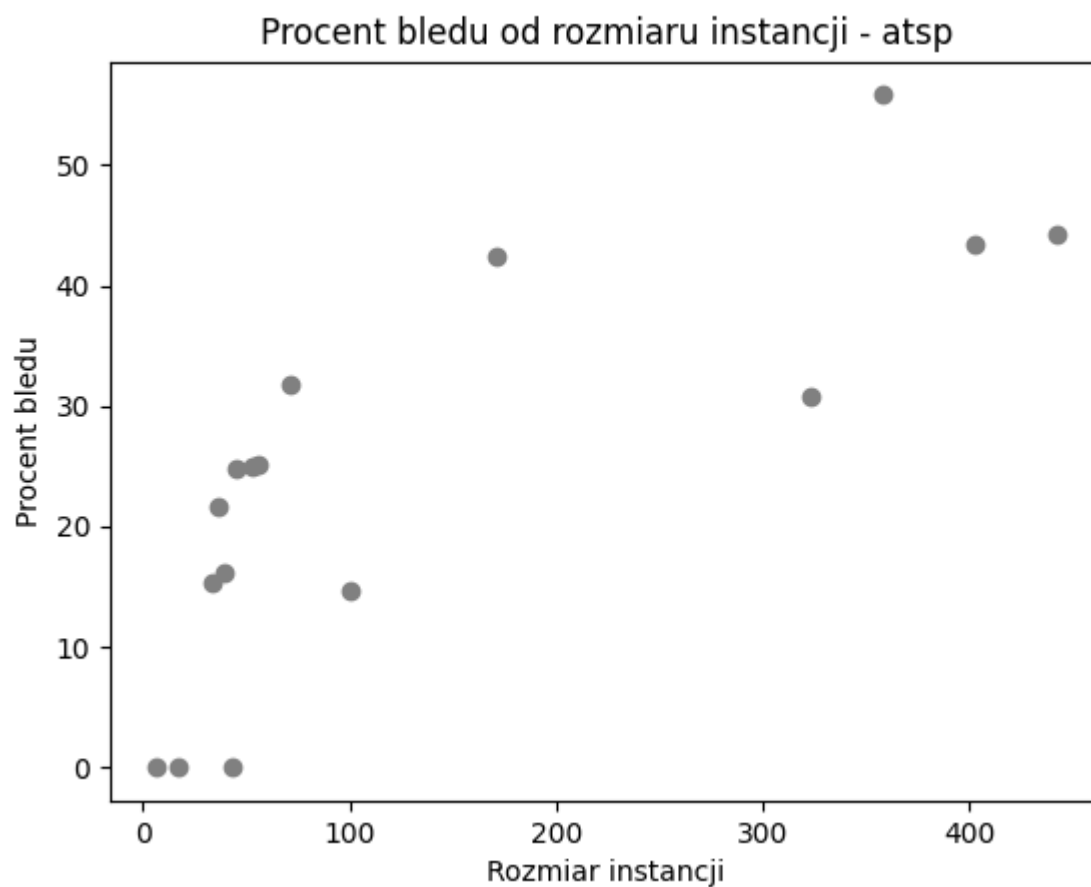
6.8. 2OPT_ZACHŁANNY_GEO

Tabela 8. Tabela wyników ATSP – 2OPT_ZACHŁANNY_GEO

Nazwa instancji	Średni koszt najlepszych rozwiązań	Średni czas[ms]	Średnie zużycie pamięci	Rozmiar instancji	Optymalny koszt	Procent błędu
tsp_6_1.txt	132	147	0	6	132	0
br17.atsp	39	1232	0	17	39	0
ftv33.atsp	1483	6354	0	34	1286	15.3188
ftv35.atsp	1791	6869	0	36	1473	21.5886
ftv38.atsp	1778	8254	0	39	1530	16.2092
p43.atsp	5622	11346	0	43	5620	0.0355872
ftv44.atsp	2014	10889	0	45	1613	24.8605
ft53.atsp	8634	16855	0	53	6905	25.0398
ftv55.atsp	2012	16909	0	56	1608	25.1244
ftv70.atsp	2571	27694	0	71	1950	31.8462
kro124p.atsp	41531	68825	0	100	36230	14.6315
ftv170.atsp	3923	168319	0	171	2755	42.3956
rbg323.atsp	1734	515949	0	323	1326	30.7692
rbg358.atsp	1812	647877	0	358	1163	55.804
rbg403.atsp	3535	796403	0	403	2465	43.4077
rbg443.atsp	3922	955757	0	443	2720	44.1912



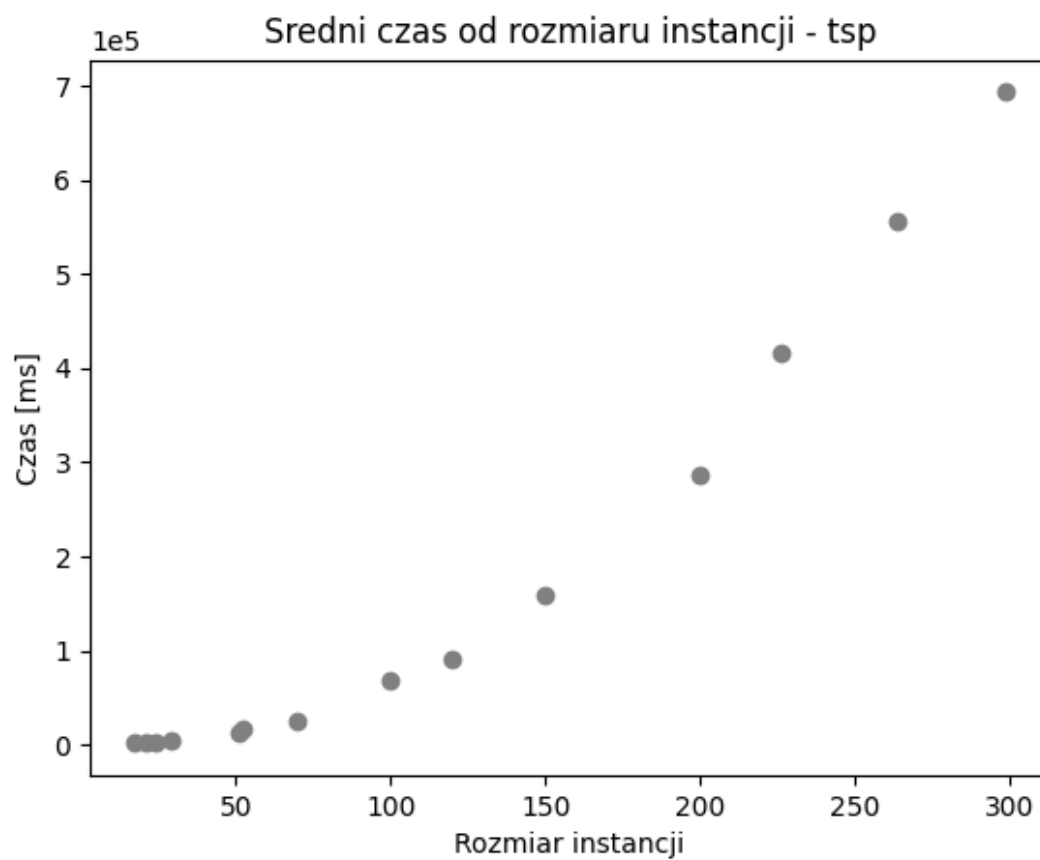
Rysunek 17. Wykres złożoności czasu od wielkości instancji dla Tabela 8



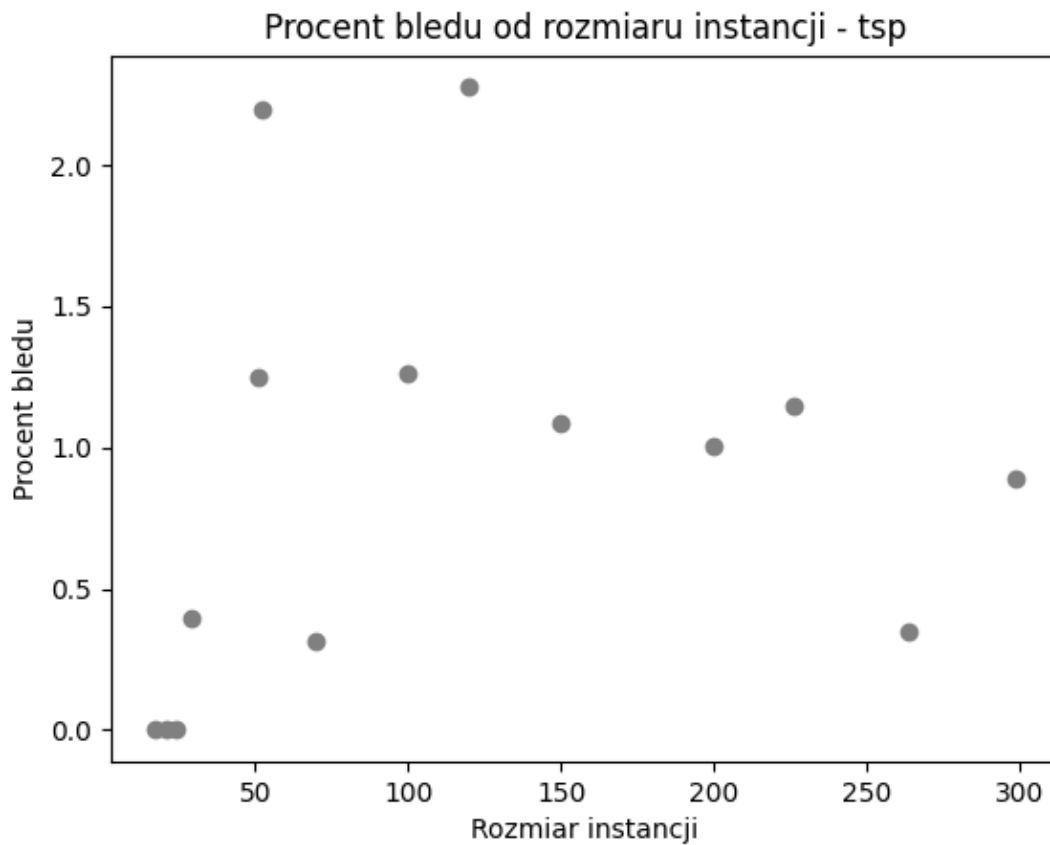
Rysunek 18. Wykres zależności błędu w porównaniu do optymalnego rozwiązania od wielkości instancji dla Tabela 8

Tabela 9. Tabela wyników TSP – 2OPT_ZACHŁANNY_GEO

Nazwa instancji	Średni koszt najlepszych rozwiązań	Średni czas[ms]	Rozmiar instancji	Optymalny koszt	Procent błędu
gr17.tsp	2085	1572	17	2085	0
gr21.tsp	2707	2457	21	2707	0
gr24.tsp	1272	3116	24	1272	0
bays29.tsp	2028	4875	29	2020	0.39604
eil51.tsp	431.321	12364	51	426	1.24915
berlin52.tsp	7707.79	16733	52	7542	2.19818
st70.tsp	677.11	24594	70	675	0.312527
kroA100.tsp	21551.3	69061	100	21282	1.26521
gr120.tsp	7100	90069	120	6942	2.276
kroB150.tsp	26413.9	159691	150	26130	1.08664
kroB200.tsp	29733.6	286591	200	29437	1.00771
pr226.tsp	81289.9	415717	226	80369	1.1458
pr264.tsp	49306.9	556863	264	49135	0.349841
pr299.tsp	48619.2	693468	299	48191	0.888643



Rysunek 19. Wykres złożoności czasu od wielkości instancji dla Tabela 9

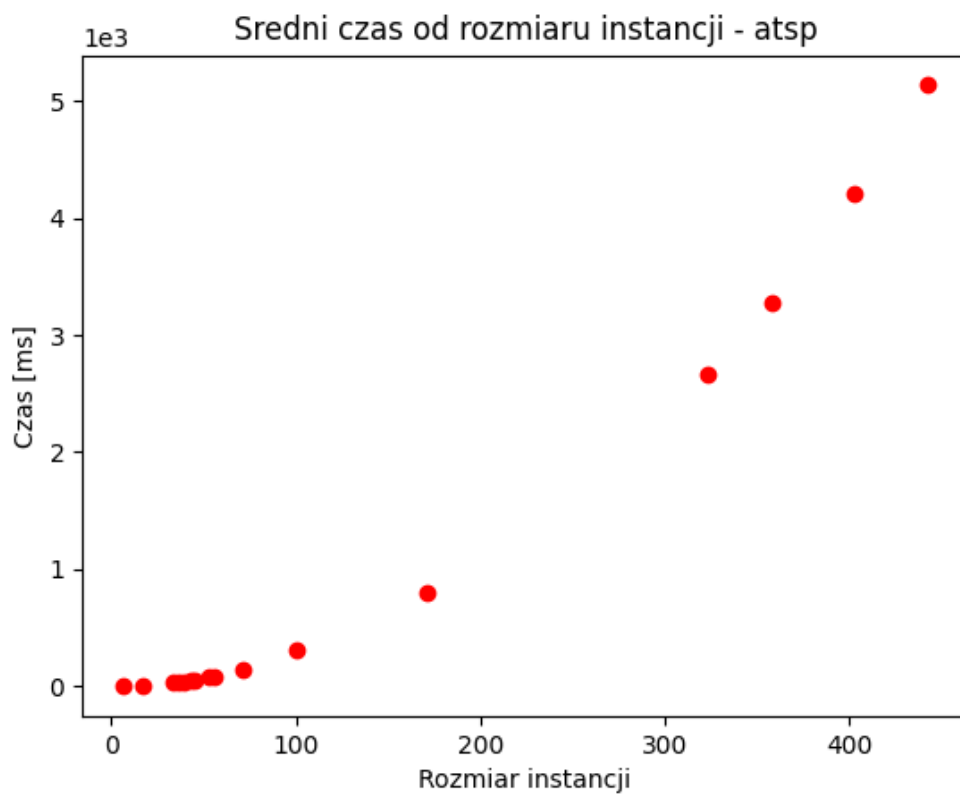


Rysunek 20. Wykres zależności błędu w porównaniu do optymalnego rozwiązania od wielkości instancji dla Tabela 9

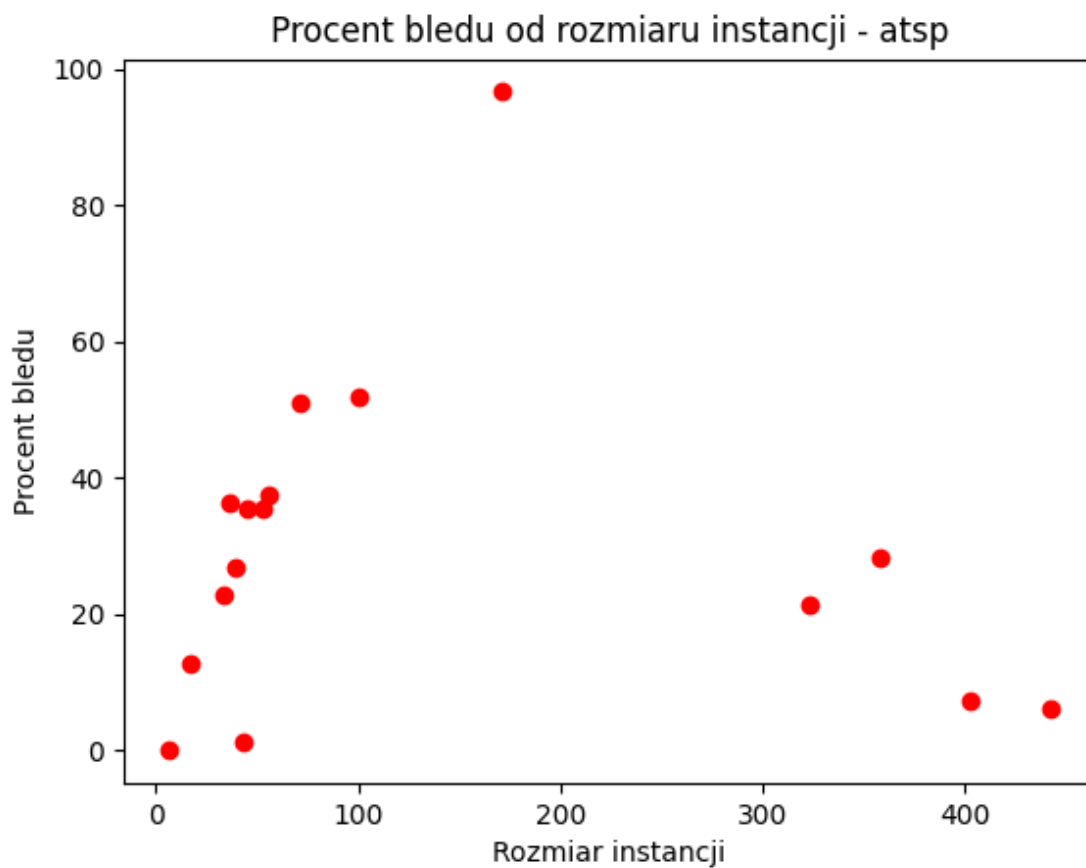
6.9. SWAP_LOSOWY_LOG

Tabela 10. Tabela wyników ATSP – SWAP_LOSOWY_LOG

	Średni koszt najlepszych rozwiązań	Średni czas[ms]	Rozmiar instancji	Optymalny koszt	Procent błędu
tsp_6_1.txt	132	1	6	132	0
br17.atsp	44	9	17	39	12.8205
ftv33.atsp	1581	33	34	1286	22.9393
ftv35.atsp	2009	32	36	1473	36.3883
ftv38.atsp	1940	39	39	1530	26.7974
p43.atsp	5685	51	43	5620	1.15658
ftv44.atsp	2184	59	45	1613	35.3999
ft53.atsp	9353	84	53	6905	35.4526
ftv55.atsp	2210	81	56	1608	37.4378
ftv70.atsp	2943	137	71	1950	50.9231
kro124p.atsp	55050	320	100	36230	51.9459
ftv170.atsp	5416	802	171	2755	96.588
rbg323.atsp	1608	2657	323	1326	21.267
rbg358.atsp	1492	3280	358	1163	28.2889
rbg403.atsp	2646	4207	403	2465	7.3428
rbg443.atsp	2885	5132	443	2720	6.06618



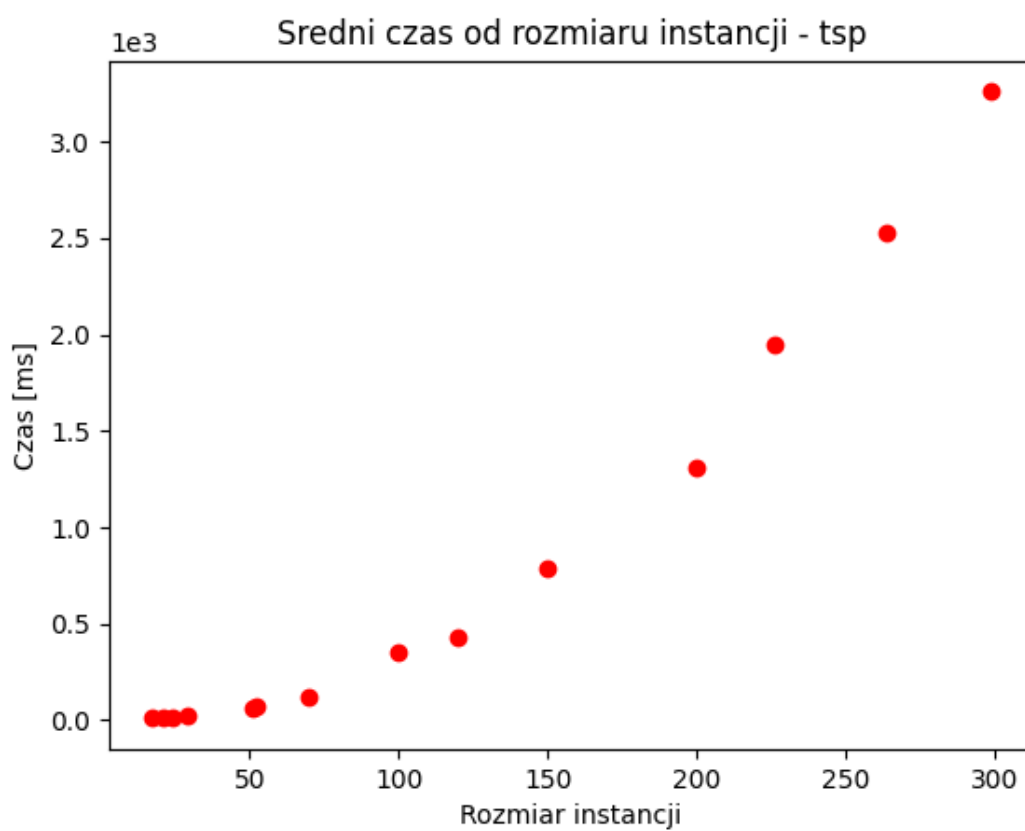
Rysunek 21. Wykres złożoności czasu od wielkości instancji dla Tabela 10



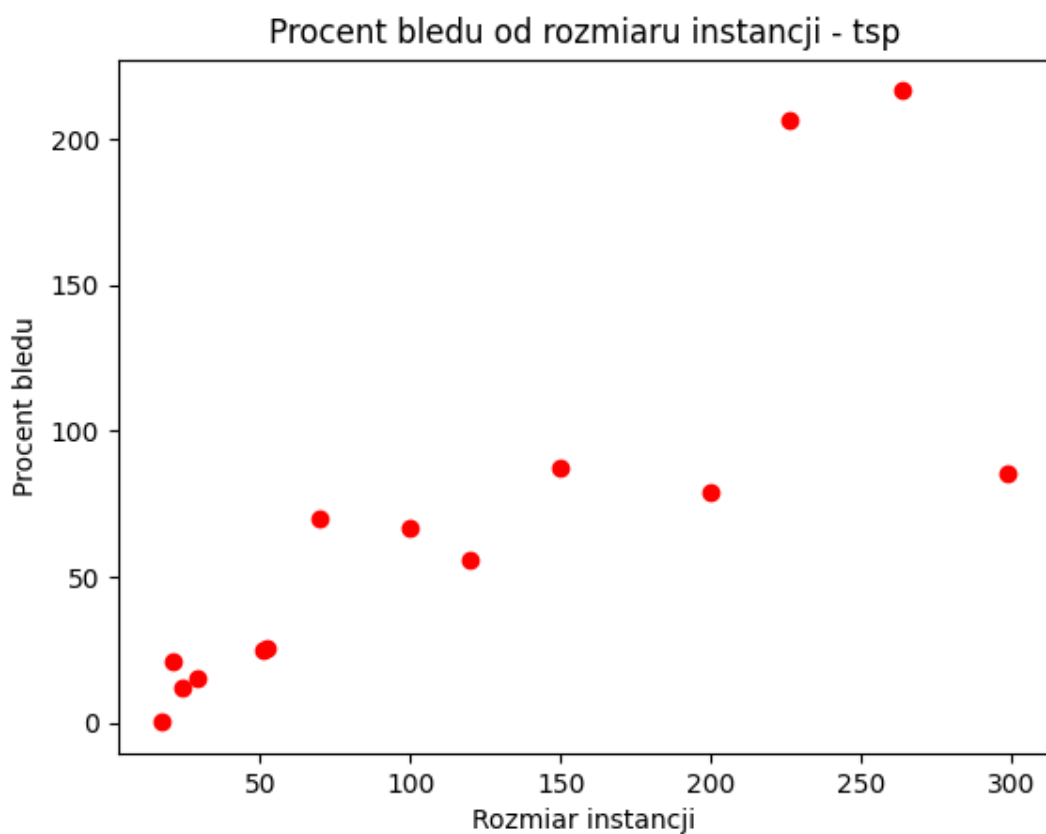
Rysunek 22. Wykres zależności błędu w porównaniu do optymalnego rozwiązania od wielkości instancji dla Tabela 10

Tabela 11. Tabela wyników TSP – SWAP_LOSOWY_LOG

Nazwa instancji	Średni koszt najlepszych rozwiązań	Średni czas[ms]	Rozmiar instancji	Optymalny koszt	Procent błędu
gr17.tsp	2090	10	17	2085	0.239808
gr21.tsp	3283	14	21	2707	21.2782
gr24.tsp	1422	15	24	1272	11.7925
bays29.tsp	2327	23	29	2020	15.198
eil51.tsp	531.634	60	51	426	24.7967
berlin52.tsp	9476.32	75	52	7542	25.6472
st70.tsp	1146.8	119	70	675	69.8969
kroA100.tsp	35467.6	348	100	21282	66.6554
gr120.tsp	10820	426	120	6942	55.8629
kroB150.tsp	48910.9	788	150	26130	87.1829
kroB200.tsp	52793.6	1309	200	29437	79.3445
pr226.tsp	246353	1953	226	80369	206.528
pr264.tsp	155523	2532	264	49135	216.522
pr299.tsp	89383.4	3260	299	48191	85.4773



Rysunek 23. Wykres złożoności czasu od wielkości instancji dla Tabela 11



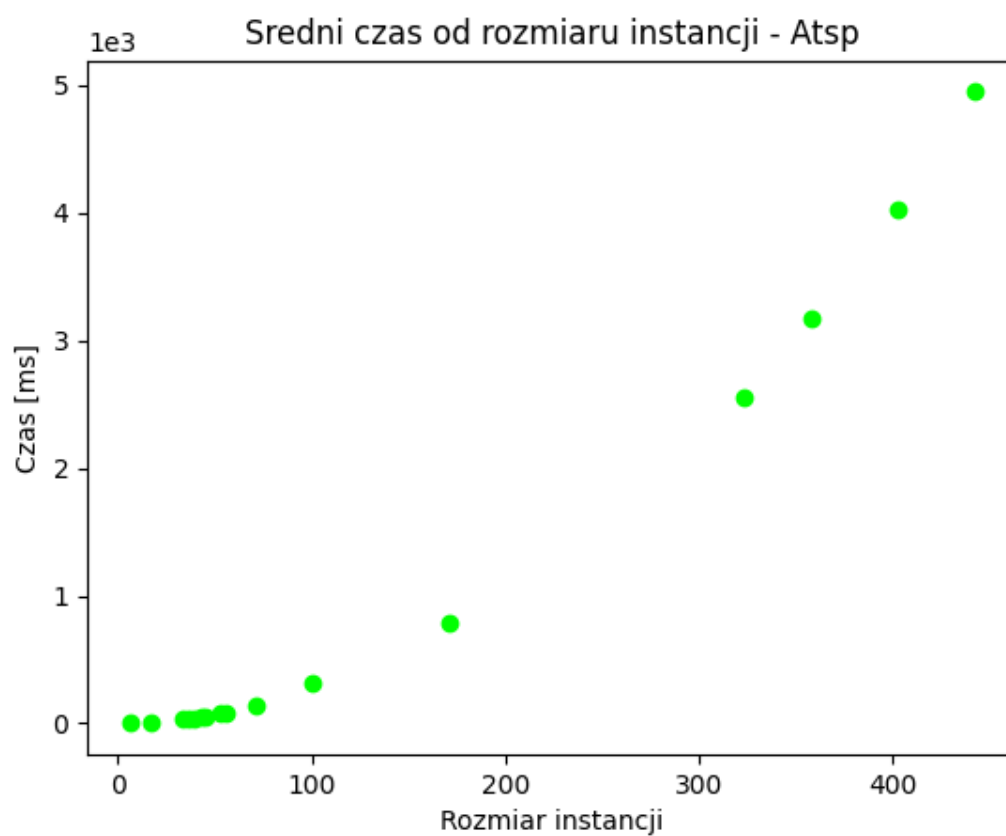
Rysunek 24. Wykres zależności błędu w porównaniu do optymalnego rozwiązania od wielkości instancji dla Tabela 11

=

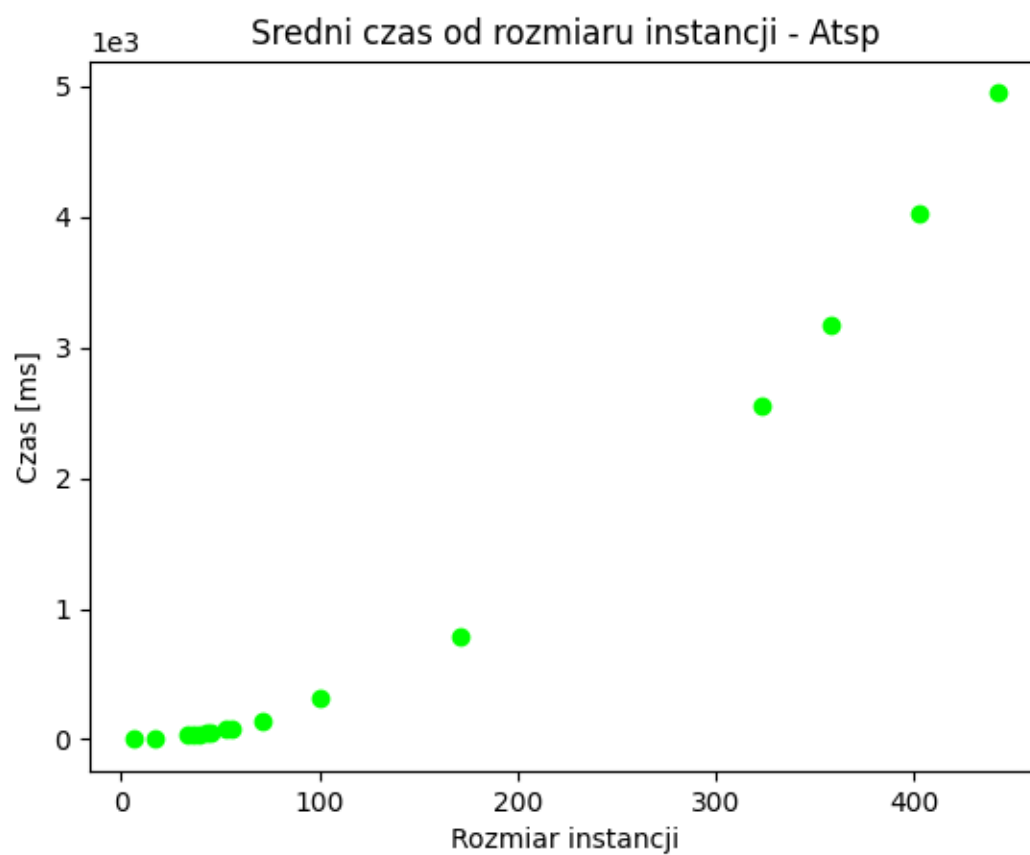
6.10. 2OPT_LOSOWY_LOG

Tabela 12. Tabela wyników ATSP – 2OPT_LOSOWY_LOG

Nazwa instancji	Średni koszt najlepszych rozwiązań	Średni czas[ms]	Rozmiar instancji	Optymalny koszt	Procent błędu
tsp_6_1.txt	154	1	6	132	16.6667
br17.atsp	39	9	17	39	0
ftv33.atsp	1904	33	34	1286	48.056
ftv35.atsp	2482	34	36	1473	68.4997
ftv38.atsp	1999	40	39	1530	30.6536
p43.atsp	5643	57	43	5620	0.409253
ftv44.atsp	3038	56	45	1613	88.3447
ft53.atsp	15276	83	53	6905	121.231
ftv55.atsp	2898	80	56	1608	80.2239
ftv70.atsp	4189	135	71	1950	114.821
kro124p.atsp	51931	321	100	36230	43.337
ftv170.atsp	9302	785	171	2755	237.641
rbg323.atsp	3574	2558	323	1326	169.532
rbg358.atsp	4026	3170	358	1163	246.174
rbg403.atsp	4754	4029	403	2465	92.86
rbg443.atsp	5113	4948	443	2720	87.9779



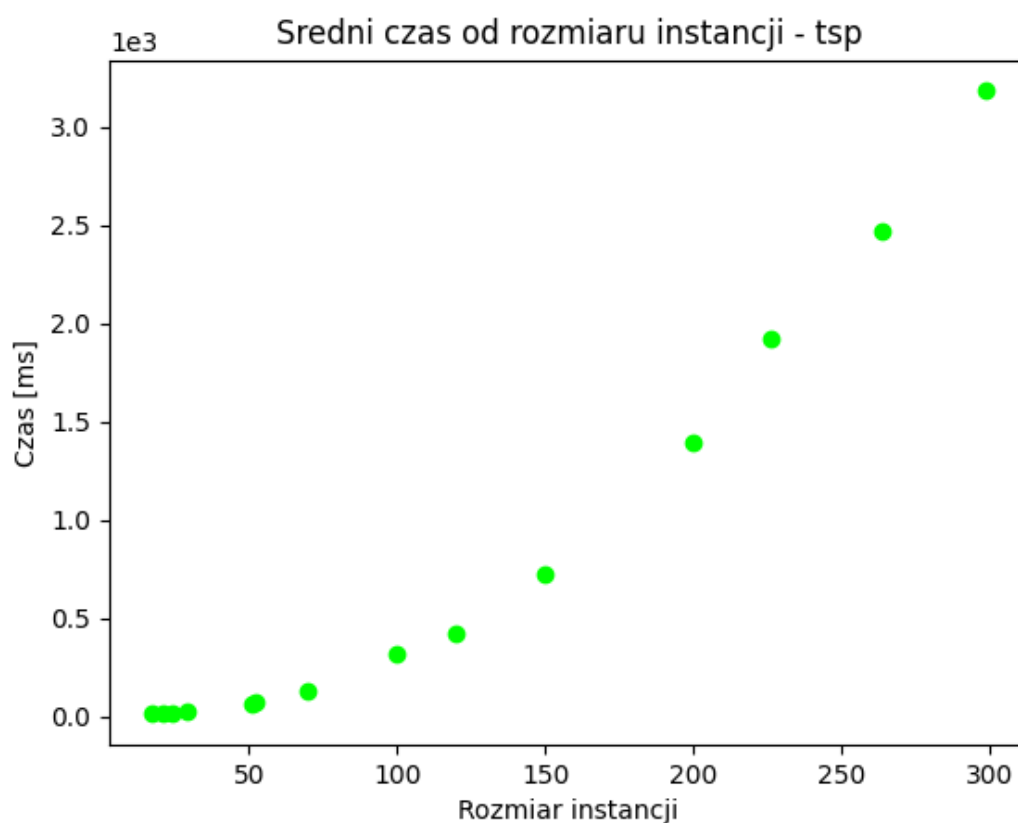
Rysunek 25. Wykres złożoności czasu od wielkości instancji dla Tabela 12



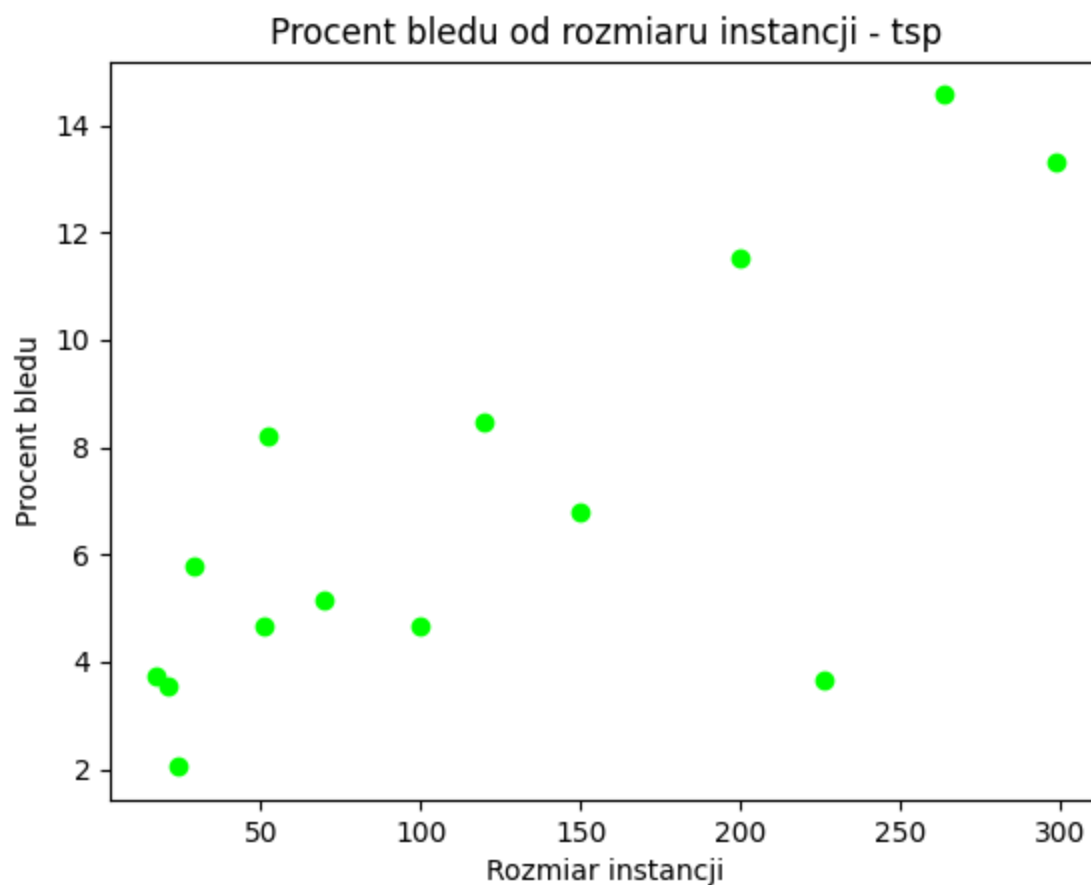
Rysunek 26. Wykres zależności błędu w porównaniu do optymalnego rozwiązania od wielkości instancji dla Tabela 12

Tabela 13. Tabela wyników TSP – 2OPT_LOSOWY_LOG

Nazwa instancji	Średni koszt najlepszych rozwiązań	Średni czas[ms]	Rozmiar instancji	Optymalny koszt	Procent błędu
gr17.tsp	2163	11	17	2085	3.74101
gr21.tsp	2803	14	21	2707	3.54636
gr24.tsp	1298	15	24	1272	2.04403
bays29.tsp	2137	23	29	2020	5.79208
eil51.tsp	445.834	60	51	426	4.65577
berlin52.tsp	8161.63	75	52	7542	8.21576
st70.tsp	709.65	128	70	675	5.13336
kroA100.tsp	22276.1	315	100	21282	4.6712
gr120.tsp	7530	420	120	6942	8.47018
kroB150.tsp	27903.8	725	150	26130	6.78826
kroB200.tsp	32827.7	1393	200	29437	11.5184
pr226.tsp	83316.6	1920	226	80369	3.66755
pr264.tsp	56291.4	2472	264	49135	14.5648
pr299.tsp	54611.9	3181	299	48191	13.3239



Rysunek 27. Wykres złożoności czasu od wielkości instancji dla Tabela 13

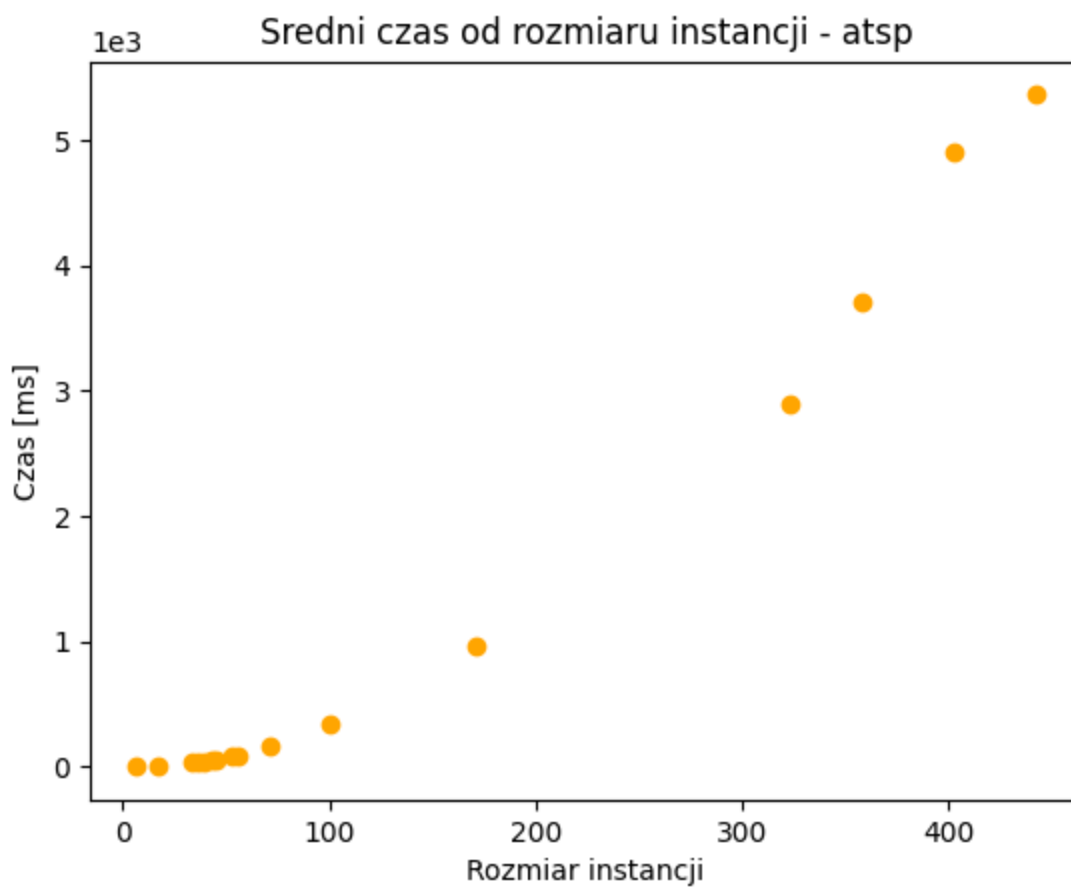


Rysunek 28. Wykres zależności błędu w porównaniu do optymalnego rozwiązania od wielkości instancji dla Tabela 13

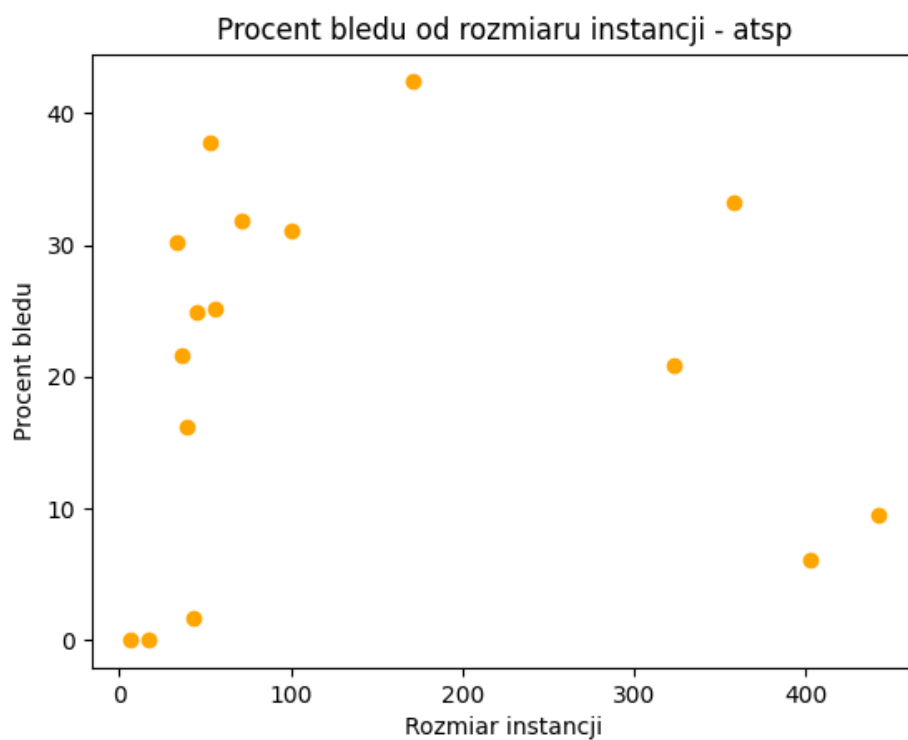
6.1.1. SWAP_ZACHŁANNY_LOG

Tabela 14. Tabela wyników ATSP – SWAP_ZACHŁANNY_LOG

Nazwa instancji	Średni koszt najlepszych rozwiązań	Średni czas[ms]	Rozmiar instancji	Optymalny koszt	Procent błędu
tsp_6_1.txt	132	1	6	132	0
br17.atsp	39	9	17	39	0
ftv33.atsp	1674	33	34	1286	30.1711
ftv35.atsp	1791	35	36	1473	21.5886
ftv38.atsp	1778	42	39	1530	16.2092
p43.atsp	5717	59	43	5620	1.72598
ftv44.atsp	2014	59	45	1613	24.8605
ft53.atsp	9514	89	53	6905	37.7842
ftv55.atsp	2012	87	56	1608	25.1244
ftv70.atsp	2571	166	71	1950	31.8462
kro124p.atsp	47506	347	100	36230	31.1234
ftv170.atsp	3923	964	171	2755	42.3956
rbg323.atsp	1602	2891	323	1326	20.8145
rbg358.atsp	1549	3707	358	1163	33.19
rbg403.atsp	2616	4910	403	2465	6.12576
rbg443.atsp	2978	5361	443	2720	9.48529



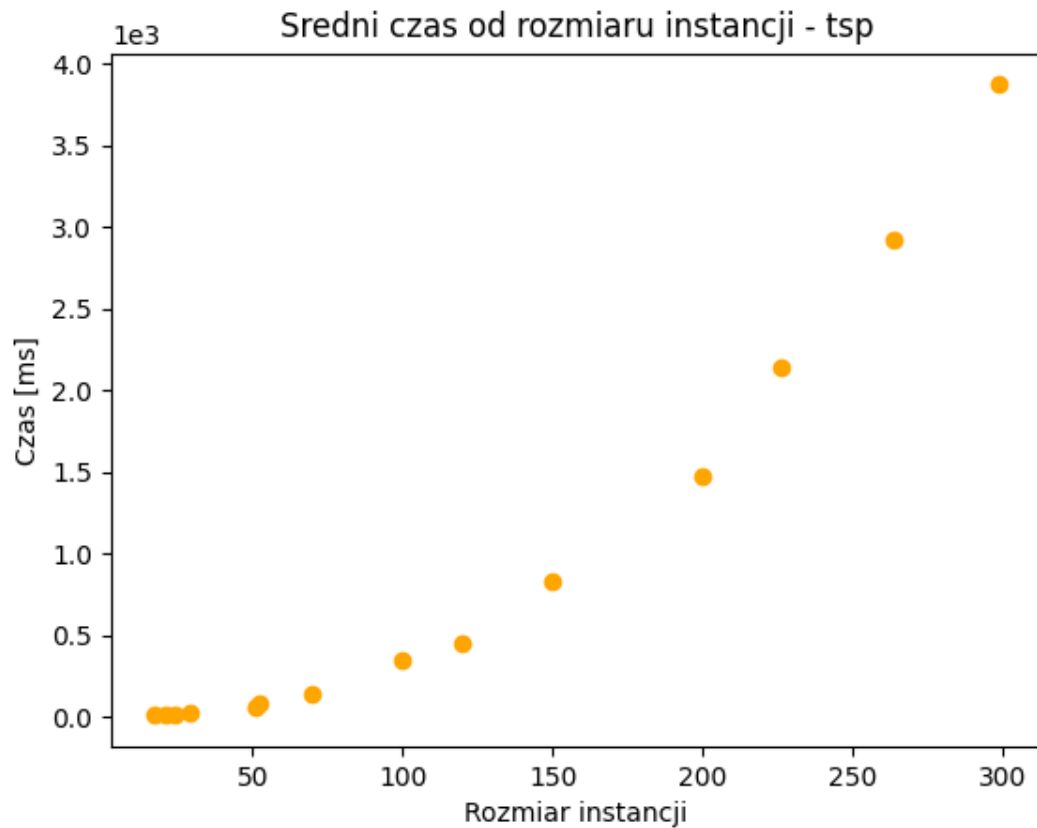
Rysunek 29. Wykres złożoności czasu od wielkości instancji dla Tabela 14



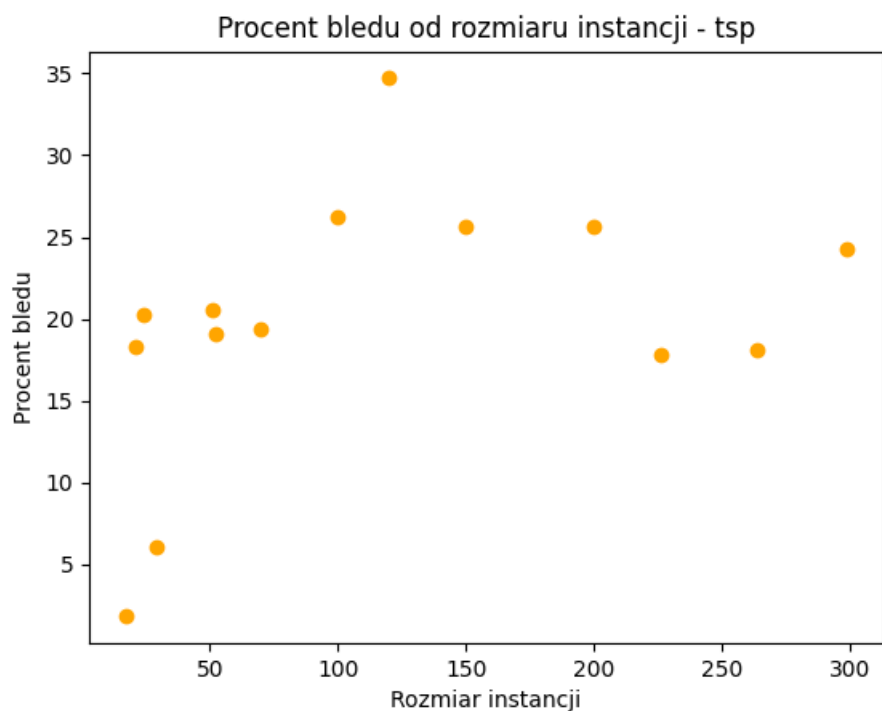
Rysunek 30. Wykres zależności błędu w porównaniu do optymalnego rozwiązania od wielkości instancji dla Tabela 14

Tabela 15. Tabela wyników TSP – SWAP_ZACHŁANNY_LOG

Nazwa instancji	Średni koszt najlepszych rozwiązań	Średni czas[ms]	Rozmiar instancji	Optymalny koszt	Procent błędu
gr17.tsp	2123	10	17	2085	1.82254
gr21.tsp	3203	15	21	2707	18.3229
gr24.tsp	1530	16	24	1272	20.283
bays29.tsp	2142	26	29	2020	6.0396
eil51.tsp	513.61	66	51	426	20.5657
berlin52.tsp	8980.92	83	52	7542	19.0788
st70.tsp	805.531	143	70	675	19.338
kroA100.tsp	26856.4	343	100	21282	26.193
gr120.tsp	9351	457	120	6942	34.7018
kroB150.tsp	32825.8	836	150	26130	25.6248
kroB200.tsp	36981.6	1476	200	29437	25.6296
pr226.tsp	94685.5	2142	226	80369	17.8134
pr264.tsp	58022.9	2925	264	49135	18.0886
pr299.tsp	59899	3871	299	48191	24.295



Rysunek 31. Wykres złożoności czasu od wielkości instancji dla Tabela 15

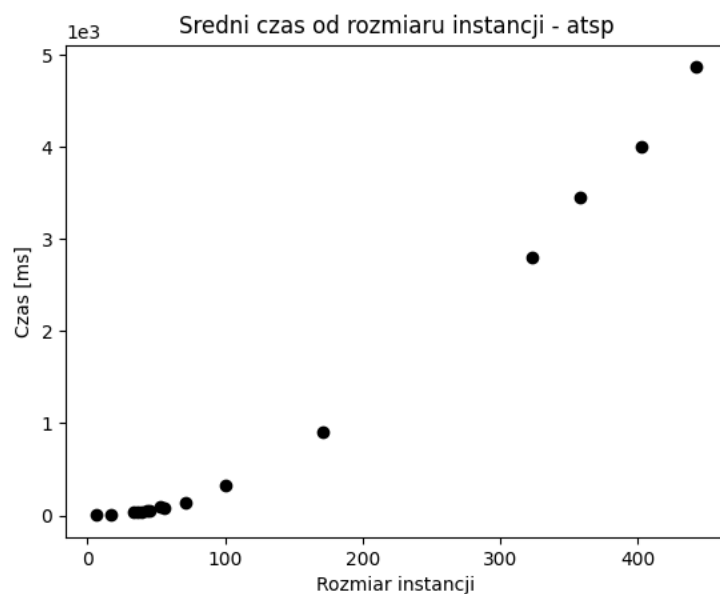


Rysunek 32. Wykres zależności błędu w porównaniu do optymalnego rozwiązania od wielkości instancji dla Tabela 15

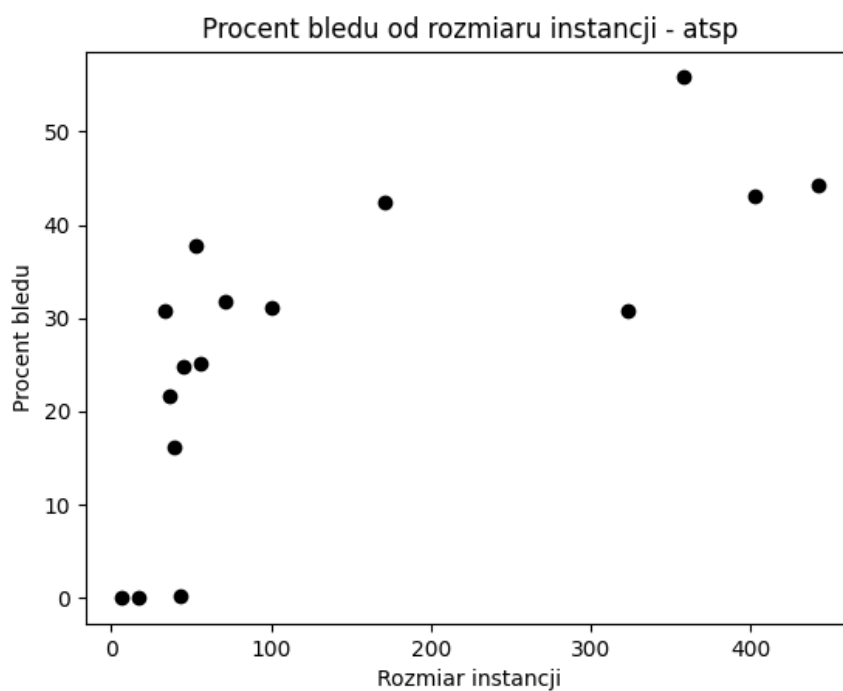
6.12. 2OPT_ZACHŁANNY_LOG

Tabela 16. Tabela wyników ATSP – 2OPT_ZACHŁANNY_LOG

Nazwa instancji	Średni koszt najlepszych rozwiązań	Średni czas[ms]	Rozmiar instancji	Optymalny koszt	Procent błędu
tsp_6_1.txt	132	2	6	132	0
br17.atsp	39	9	17	39	0
ftv33.atsp	1683	35	34	1286	30.8709
ftv35.atsp	1791	35	36	1473	21.5886
ftv38.atsp	1778	42	39	1530	16.2092
p43.atsp	5634	56	43	5620	0.24911
ftv44.atsp	2014	58	45	1613	24.8605
ft53.atsp	9514	89	53	6905	37.7842
ftv55.atsp	2012	87	56	1608	25.1244
ftv70.atsp	2571	145	71	1950	31.8462
kro124p.atsp	47506	331	100	36230	31.1234
ftv170.atsp	3923	902	171	2755	42.3956
rbg323.atsp	1734	2797	323	1326	30.7692
rbg358.atsp	1812	3457	358	1163	55.804
rbg403.atsp	3527	4006	403	2465	43.0832
rbg443.atsp	3922	4865	443	2720	44.1912



Rysunek 33. Wykres złożoności czasu od wielkości instancji dla Tabela 16

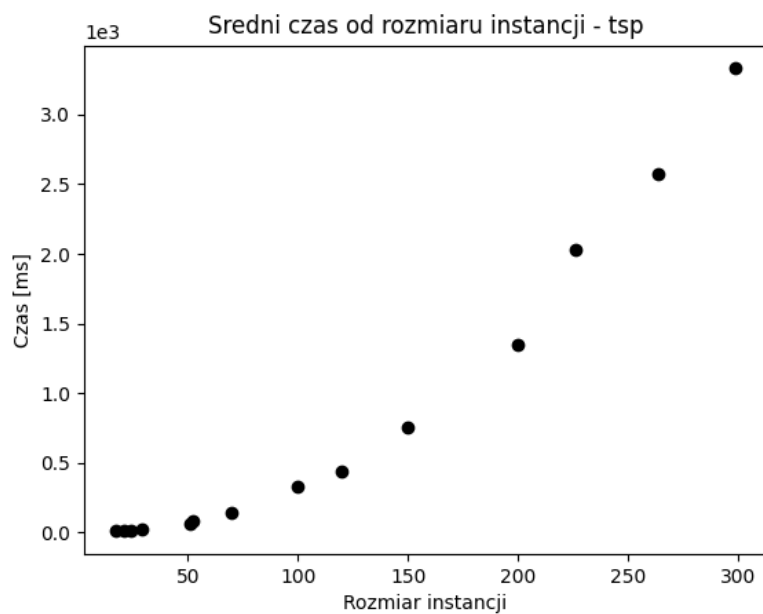


Rysunek 34. Wykres zależności błędu w porównaniu do optymalnego rozwiązania od wielkości instancji dla Tabela 16

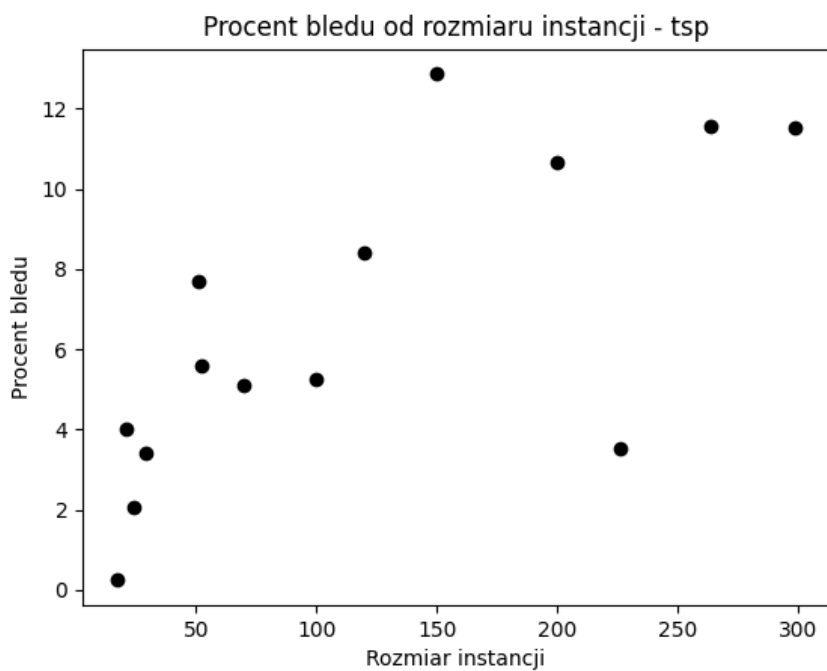
Tabela 17. Tabela wyników TSP – 2OPT_ZACHŁANNY_LOG

Nazwa instancji	Średni koszt najlepszych rozwiązań	Średni czas[ms]	Rozmiar instancji	Optymalny koszt	Procent błędu
gr17.tsp	2090	10	17	2085	0.239808
gr21.tsp	2816	15	21	2707	4.0266
gr24.tsp	1298	16	24	1272	2.04403
bays29.tsp	2089	25	29	2020	3.41584
eil51.tsp	458.799	65	51	426	7.69929
berlin52.tsp	7963.39	84	52	7542	5.5873

st70.tsp	709.335	147	70	675	5.08671
kroA100.tsp	22400.7	331	100	21282	5.25669
gr120.tsp	7526	442	120	6942	8.41256
kroB150.tsp	29492	757	150	26130	12.8664
kroB200.tsp	32576.6	1351	200	29437	10.6655
pr226.tsp	83206	2025	226	80369	3.52997
pr264.tsp	54817.5	2573	264	49135	11.5651
pr299.tsp	53743.5	3330	299	48191	11.5219

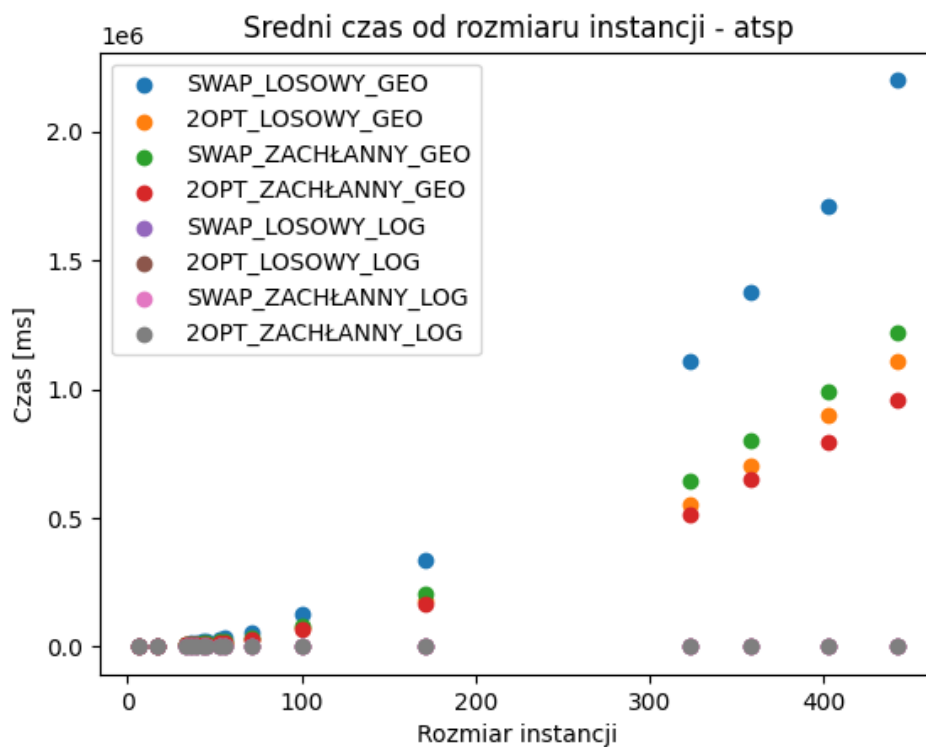


Rysunek 35. Wykres złożoności czasu od wielkości instancji dla Tabela 17

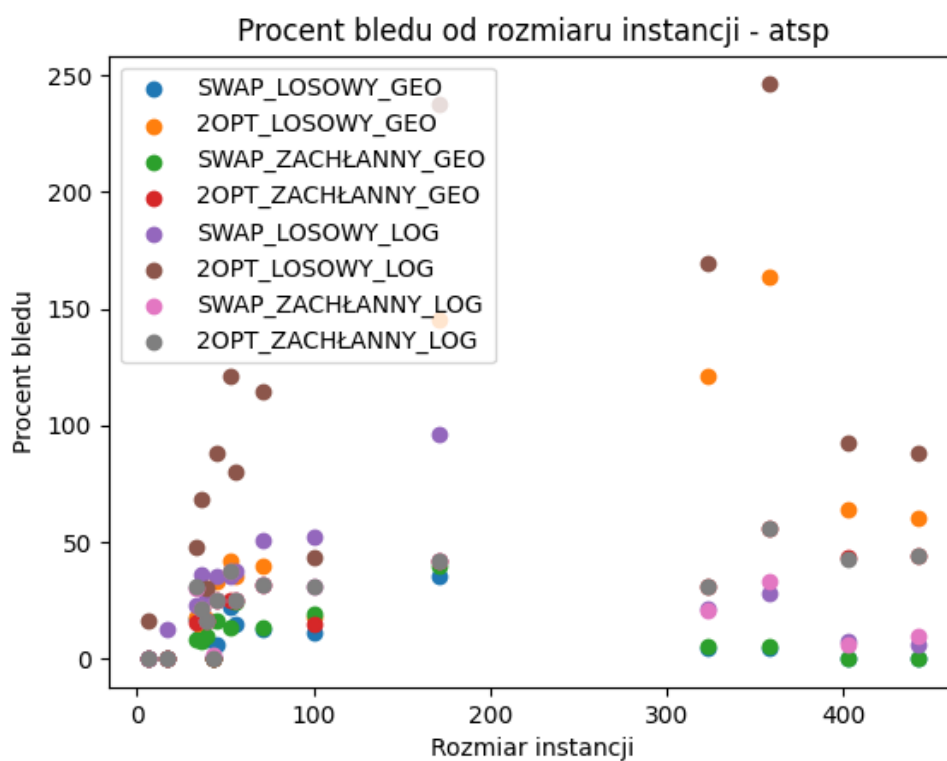


Rysunek 36. Wykres zależności błędu w porównaniu do optymalnego rozwiązania od wielkości instancji dla Tabela 17

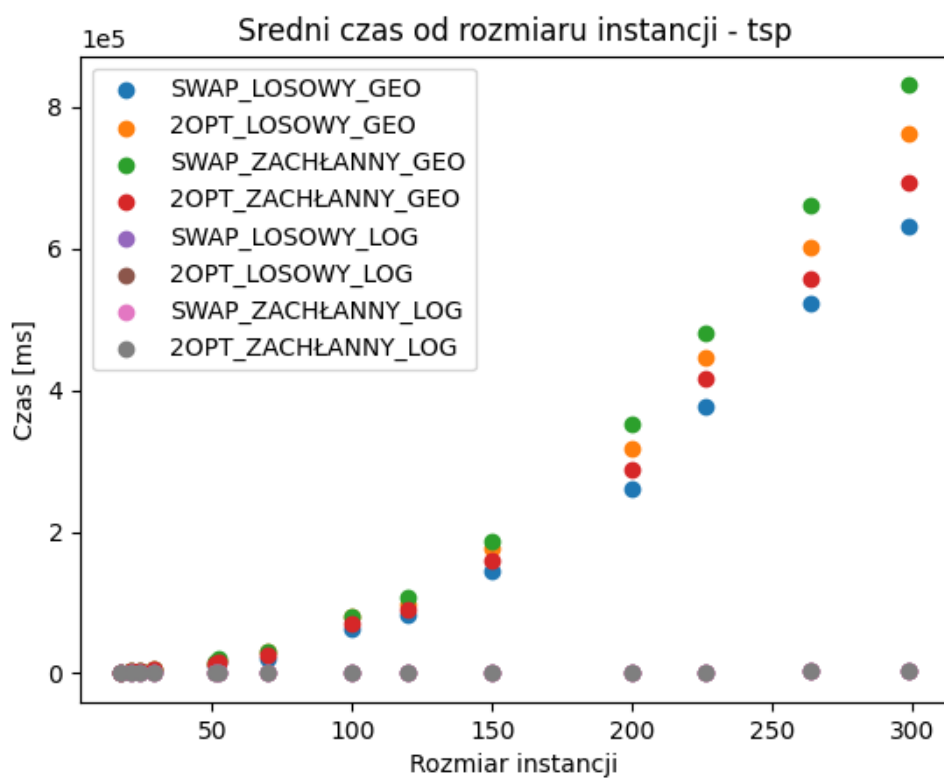
6.13. Porównanie kombinacji SA



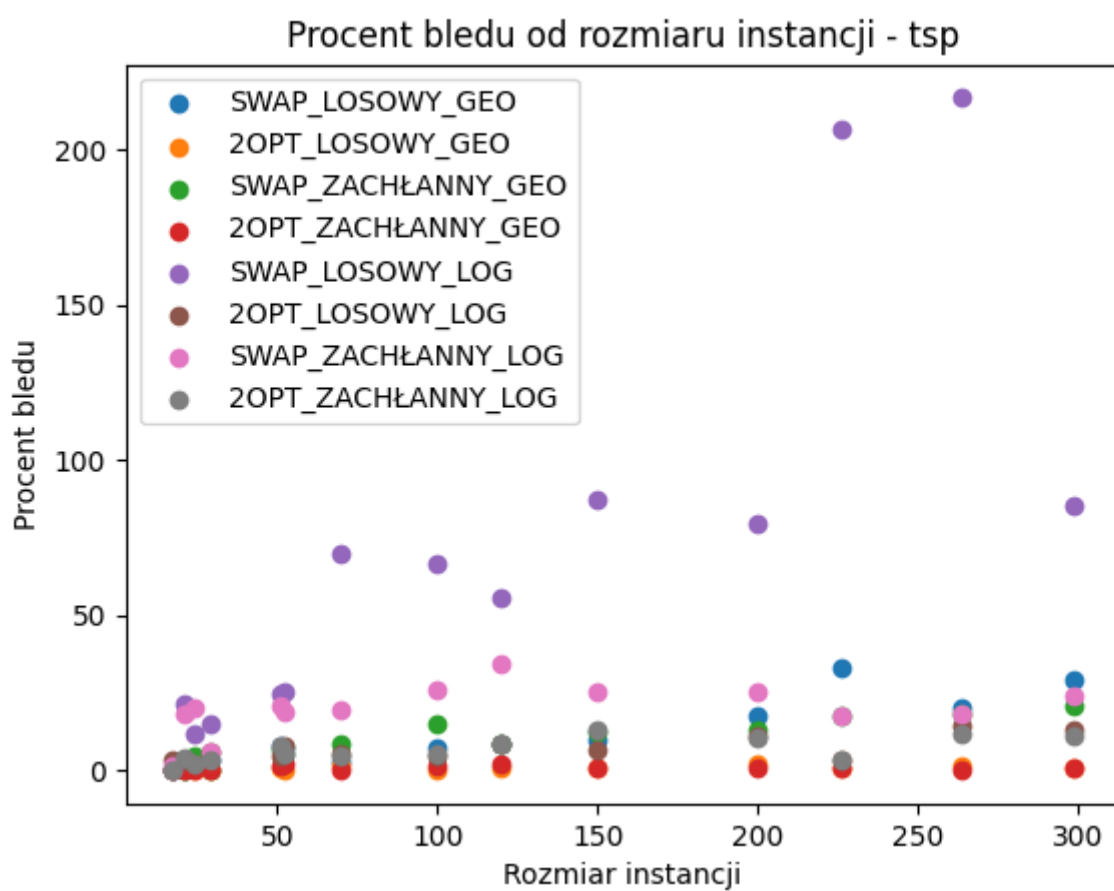
Rysunek 37. ATSP - Porównanie czasów wszystkich kombinacji SA



Rysunek 38. ATSP - Porównanie błędów wszystkich kombinacji SA

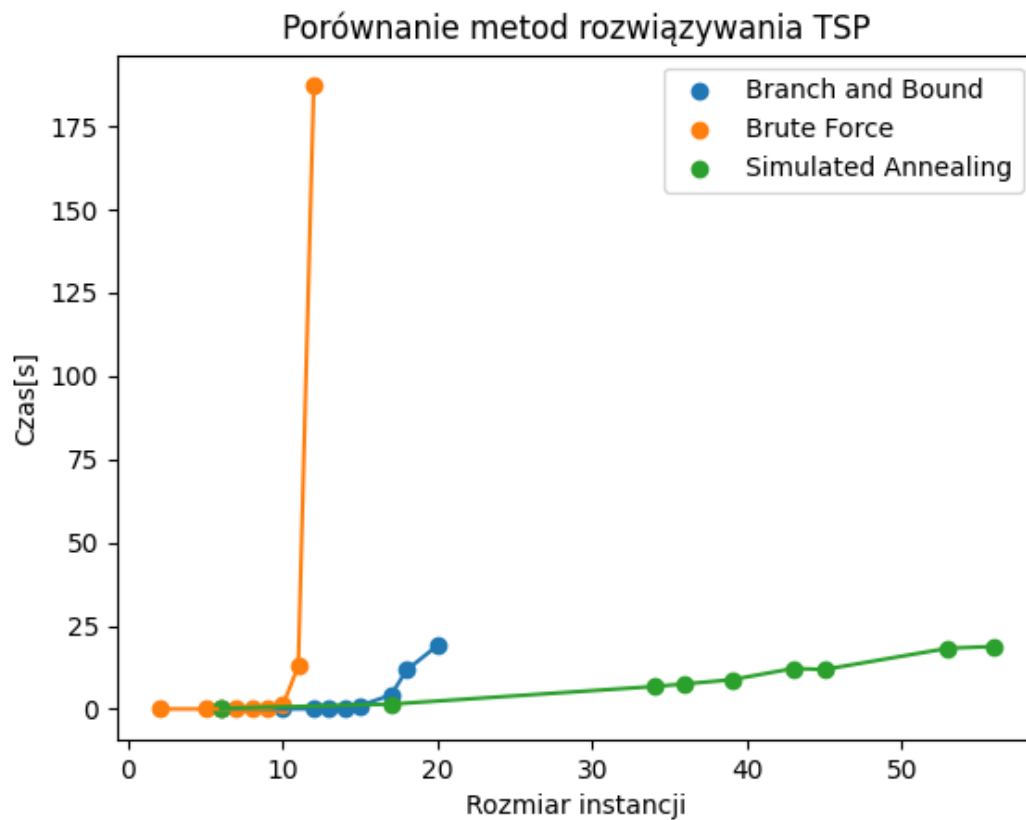


Rysunek 39. TSP - Porównanie czasów wszystkich kombinacji SA

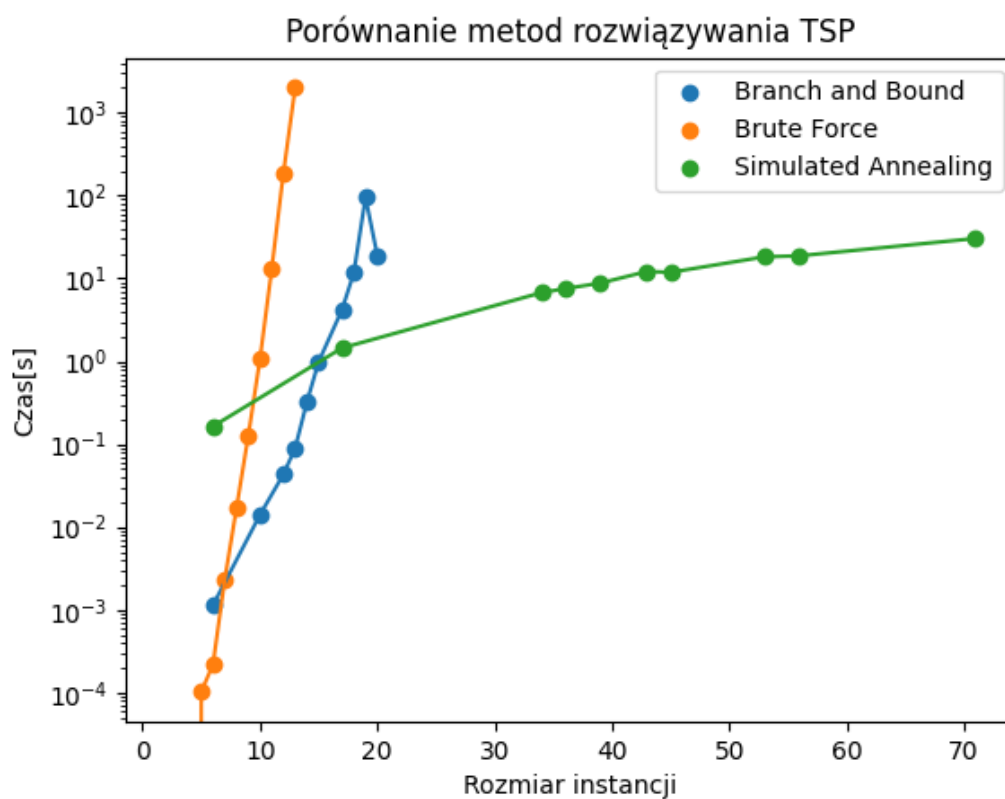


Rysunek 40. TSP - Porównanie błędów wszystkich kombinacji SA

6.14. Porównanie z innymi metodami rozwiązywania TSP



Rysunek 41. Porównanie zależności czasowych od rozmiaru instancji różnych metod rozwiązywania TSP



Rysunek 42. Porównanie zależności czasowych od rozmiaru instancji różnych metod rozwiązywania TSP - skala logarytmiczna

7. Analiza i wnioski

7.1. Analiza czasu działania metody

Z analizy danych wynika, że czas obliczeniowy zastosowanego algorytmu symulowanego wyżarzania wykazuje tendencję wzrostową wraz ze zwiększaniem się rozmiaru macierzy, co ma miejsce niezależnie od specyficznych kombinacji wybranych parametrów wejściowych – na Rysunkach 37 i 39 mogłoby się wdawać, że czas dla logarytmicznego systemu chłodzenia jest prawie stały, ale na Rysunku 35 widać, że nie byłoby to prawdą.

Obserwuje się, że krzywe reprezentujące czas obliczeń algorytmu symulowanego wyżarzania w funkcji wielkości macierzy sugerują eksponentyjny wzrost. Jest to fenomen charakterystyczny dla problemów o wysokiej złożoności obliczeniowej, gdzie czas potrzebny na wykonanie obliczeń rośnie w sposób bardziej gwałtowny niż proporcjonalny do rozmiarów zadania (np. Rysunki 5 i 11).

Zastosowanie metody chłodzenia o charakterze logarytmicznym zdaje się przynosić lepsze rezultaty pod kątem szybkości działania, aczkolwiek zauważalne jest, iż wiąże się to z większym średnim błędem, co sugeruje gorszą dokładność w porównaniu do oczekiwanego rozwiązania optymalnego (Rysunki 38 i 40). Możliwym wyjaśnieniem tego zjawiska jest zbyt intensywne zmniejszanie temperatury w procesie chłodzenia, co może skutkować niewystarczającą eksploracją przestrzeni rozwiązań. Strategie poszukiwania w sąsiedztwie swap i 2-opt prezentują porównywalne czasy obliczeń.

7.2. Analiza stosunku otrzymanych najlepszych rozwiązań, a rozwiązania optymalnego

W kontekście algorytmu symulowanego wyżarzania, kluczowym aspektem wpływającym na precyzję rozwiązań (mniejsze odchylenie od wyniku idealnego) jest strategia chłodzenia. Geometryczny schemat chłodzenia, połączony z metodą przeszukiwania sąsiedztwa typu swap, okazał się najbardziej skuteczny, osiągając wyniki z maksymalnym błędem na poziomie 35% dla analizowanych instancji, co jest znacznie lepszym osiągnięciem w stosunku do wyników uzyskanych przy zastosowaniu schematu logarytmicznego, gdzie najgorszy wynik to błąd na poziomie 246% (Tabela 2 i Tabela 12). Krzywe błędów dla metody geometrycznej wykazują mniejszą wariancję w porównaniu z metodami logarytmicznymi, co wskazuje na bardziej stabilne osiąganie wyników zbliżonych do optymalnych przy użyciu chłodzenia geometrycznego (Rysunki 38 i 40).

Zastosowanie metody 2-opt do przeszukiwania sąsiedztwa pokazało polepszenie jakości rozwiązań dla instancji TSP, nie wpływając na czas obliczeń. Metoda ta również bardzo dobrze sprawdzała się przy testach logarytmicznego schematu chłodzenia.

7.3. Analiza porównania SA z innymi metodami

Metoda Brute Force jest zdecydowanie najbardziej czasochłonną metodą, co jest wyraźnie widoczne na obu wykresach (Rysunki 41 i 42). Już dla relatywnie małych instancji czas obliczeń gwałtownie rośnie, co wskazuje na jej niepraktyczność dla instancji o większym rozmiarze.

Metoda Branch and Bound pokazuje lepszą efektywność niż Brute Force, jednakże jej czas obliczeń również wzrasta eksponentyjnie wraz z rozmiarem instancji, zwłaszcza na wykresie ze skalą logarytmiczną (Rysunek 42). To sugeruje, że metoda ta ma ograniczenia i może nie być efektywna dla dużych instancji TSP.

Metoda Simulated Annealing (SA) demonstruje znacznie niższą czasochłonność w porównaniu z dwiema innymi metodami. Na wykresie liniowym SA utrzymuje stabilną, niską czasochłonność, nawet przy rosnącym rozmiarze instancji, co jest pozytywną charakterystyką tej metody.

Na wykresie logarytmicznym SA wykazuje lekkie zwiększanie czasu obliczeń, ale jest to wzrost bardzo powolny, co może wskazywać na to, że metoda ta jest skalowalna i może być efektywnie stosowana dla dużych instancji problemu.

Ogólne wnioski:

Metoda Simulated Annealing to metoda heurystyczna, co oznacza, że może nie zawsze znajdować rozwiązanie optymalne, ale daje wystarczająco dobre rozwiązanie w znacznie krótszym czasie, co jest kluczowe dla praktycznego zastosowania, szczególnie w większych instancjach problemu.

Zastosowanie metod deterministycznych takich jak Brute Force czy Branch and Bound jest ograniczone ze względu na ich wysoką złożoność czasową, szczególnie w przypadku dużych instancji TSP, gdzie czas obliczeń rośnie w sposób eksponencjalny.

Wykres z skalą logarytmiczną pozwala na lepsze zrozumienie różnic w skali czasu obliczeń między metodami, szczególnie kiedy różnice te są bardzo duże i obejmują wiele rzędów wielkości.