

# Raport 2

## **Przedmiot: Projektowanie Efektywnych algorytmów - Projekt**

Data złożenia: 23.01.2025

**Temat zadania:** Rozwiązanie problemu komiwojażera (TSP)

**Nr i treść zadania:**

Zadanie nr 3 i 4.

Rozwiązanie problemu komiwojażera metodami: symulowanego wyżarzania, przeszukiwania z zakazami i algorytmów mrówkowych.

Piotr Strycharczyk, 272995

# Punkt 1 - Opis problemu

## 1. Problem komiwojażera.

Zadanie polega na znalezieniu ścieżki, dla której suma wag będzie miała minimalną wartość w grafie, tak aby każdy wierzchołek został odwiedzony dokładnie raz. Dodatkowym założeniem jest fakt, że testowane grafy posiadają cykle Hamiltona. Ścieżka ma się zakończyć w tym samym wierzchołku, w którym zaczęliśmy naszą podróż.

## 2. Jakie wyniki chcemy uzyskać

Wyniki, które chcemy uzyskać w tej części projektu mają być akceptowalnym przybliżeniem dokładnego wyniku rozwiązania problemu.

Dla większych grafów akceptujemy rozwiązania o większym błędzie:

*Grafy mniejsze:  $V < 25$ , błąd  $\approx 0\%$ ,  
grafy średnie:  $V \in [25; 74]$ , błąd  $\leq 50\%$   
grafy większe:  $V \in [75; 449]$ , błąd  $\leq 100\%$*

Dodatkowo chcemy aby otrzymane wyniki miały akceptowalny czas realizacji tzn.  $\leq 15min$ .

## 3. Implementacja wybranych metod rozwiązywania problemu TSP:

W każdej z moich metod posiadam implementację zegara, który sprawdza czy nie upłynął maksymalny czas jaki ma wykonywać się dany algorytm – 15 minut, jeśli tak się stanie to algorytm jest niezwłocznie kończony.

Dodatkowo w implementacji symulowanego wyżarzania posiadam warunki zakończenia takie jak: licznik iteracji bez poprawy, %LB (dolnego ograniczenia), temperatura minimalna, znalezione optimum dla instancji i wspomniany wyżej czas.

W implementacji tabu search i algorytmów mrówkowych we wszystkich wersjach posiadam warunki zakończenia: licznik iteracji bez poprawy, %LB, znalezione optimum dla instancji oraz czas.

### a. Metoda symulowanego wyżarzania

Algorytm rozpoczyna się od wygenerowania początkowego rozwiązania, które może być wyznaczone na podstawie heurystyki najbliższego sąsiada lub w sposób losowy. Następnie obliczane jest dolne ograniczenie (za pomocą MST) dla jednego z warunków zatrzymania algorytmu. Ustalana jest temperatura początkowa, która zależy od wybranego schematu chłodzenia. Algorytm wchodzi w pętlę, która trwa dopóki spełniane są warunki wymienione powyżej. Wewnątrz pętli w której jest obniżana temperatura posiadam pętlę, w której wykonywane są próby ulepszenia bieżącego rozwiązania poprzez wygenerowanie nowych sąsiednich rozwiązań. Są one tworzone poprzez niewielkie zmiany w aktualnej trasie, na przykład przez zamianę dwóch elementów (swap – losowe wylosowanie 2 różnych wierzchołków i ich zamiana) i poprzez metodę 2opt (wylosowanie 2 różnych wierzchołków i odwrócenie trasy między nimi). Dla każdego nowego rozwiązania obliczany jest jego koszt. Jeśli koszt jest lepszy niż najlepsze dotychczasowe rozwiązanie, to rozwiązanie to jest zapamiętywane jako nowe najlepsze, a licznik iteracji bez poprawy jest resetowany. Jeśli koszt jest gorszy, to rozwiązanie może być zaakceptowane z prawdopodobieństwem obliczanym ze wzoru:

$$p = \frac{1}{1 + e^{\frac{x_a - x_k}{T}}} \quad ZW_p = [0, 1),$$

gdzie  $x_k$  – rozwiązanie obecnie wygenerowane w sąsiedztwie,  
 $x_a$  – dotychczasowe najlepsze rozwiązanie  
 $T$  – obecna temperatura

,które maleje wraz ze spadkiem temperatury. Dzięki temu algorytm może wychodzić z lokalnych minimów i eksplorować różne obszary przestrzeni rozwiązań. Wygenerowane rozwiązanie gorsze jest akceptowane gdy wylosowana liczba z zakresu  $[0, 1)$ , jest mniejsza od obliczonego prawdopodobieństwa. W kolejnej iteracji głównej pętli (zewnętrznej) po wcześniejszym obniżeniu temperatury, przeglądamy sąsiedztwo tegoż rozwiązania.

#### b. Metoda tabu search (przeszukiwanie z zakazami)

Metoda przeszukiwania z zakazami, zwana algorytmem tabu search, to heurystyczne podejście do rozwiązywania problemów optymalizacyjnych, które unika utknięcia w lokalnych minimach poprzez tymczasowe zakazywanie powrotu do niedawno odwiedzonych rozwiązań.

Algorytm rozpoczyna się od wygenerowania początkowego rozwiązania, które jest zazwyczaj losowe lub wygenerowane z alg NN. To rozwiązanie zostaje zapisane jako bieżące, a jego koszt jako najlepszy koszt początkowy. Tworzona jest również lista zakazów (tabu list), czyli struktura pamięci, która śledzi czy przypadkiem nie będziemy przeglądać sąsiedztwa rozwiązania już przeglądanego wcześniej.

Algorytm działa w ramach głównej pętli, która trwa do momentu spełnienia jednego z warunków zakończenia wymienionych powyżej w omawianym teraz punkcie. W każdej iteracji generowane jest sąsiedztwo bieżącego rozwiązania poprzez wprowadzenie małych zmian, takich jak zamiana elementów (swap) w strukturze rozwiązania lub 2opt. Każde sąsiednie rozwiązanie jest oceniane pod kątem jego kosztu.

Z sąsiedztwa wybierane jest najlepsze rozwiązanie, nawet jeśli jest gorsze od obecnego, pod warunkiem, że zmiana prowadząca do tego rozwiązania nie znajduje się na liście zakazów. Dzięki temu algorytm unika cyklicznego powracania do tych samych rozwiązań i eksploruje nowe obszary przestrzeni poszukiwań. Jeśli jednak wybrane rozwiązanie poprawia najlepszy dotychczasowy wynik, zostaje zapisane jako nowe najlepsze rozwiązanie.

Po wybraniu nowego bieżącego rozwiązania aktualizowana jest lista zakazów. Nowo wykonana zmiana zostaje dodana do listy, a najstarsze zakazy są usuwane, aby lista miała stałą, ograniczoną długość. Również kadencja dla każdego elementu na liście tabu jest zmniejszana o 1 z każdą iteracją głównej pętli algorytmu. Mechanizm ten zapewnia równowagę między eksploracją nowych rozwiązań a zachowaniem różnorodności w przestrzeni poszukiwań.

Proces ten jest powtarzany, aż do spełnienia warunku zakończenia. Wynikiem algorytmu jest najlepsze znalezione rozwiązanie wraz z jego kosztem. Dzięki zastosowaniu listy zakazów metoda ta skutecznie unika lokalnych minimów i znajduje rozwiązania wysokiej jakości, nawet w trudnych problemach optymalizacyjnych.

#### c. Metoda ACOCAS (ilościowa)

ACOCAS (Ant Colony Optimization Cycle Ant System) to algorytm optymalizacji, który opiera się na cyklicznej aktualizacji feromonów przez mrówki poruszające się po grafie. Ilość feromonu zostawianego przez pojedynczą mrówkę na danej krawędzi jest

obliczana ze wzoru:  
 $\frac{Q}{cost}$ , gdzie  $Q$  to ilość feromonu, natomiast  $cost$  to koszt całej ścieżki przebytej przez daną mrówkę. Stąd wynika, że na każdej krawędzi przebytej

przez  $k$ -tą mrówkę jest tyle samo pozostawionego feromonu. Feromony są odparowywane na całym grafie co każdą iterację algorytmu, co zapobiega nadmiernej koncentracji feromonów na jednej krawędzi. Proces jest powtarzany do momentu spełnienia warunku zakończenia, podanego powyżej w omawianym punkcie.

**d. Metoda ACODAS (gęstościowa)**

Metoda ta jest niemalże taka sama jak poprzednia z tą różnicą, że mrówka na każdej krawędzi pozostawia tę samą ilość feromonu  $Q$ . Redukuje to wpływ długości ścieżki na to ile zostawia mrówka feromonu – nie ma to znaczenia w tej odmianie algorytmu mrówkowego, bo na każdej krawędzi.

**e. Metoda ACOQAS (ilościowa)**

W tej wersji algorytmu mrówkowego ilość pozostawianego feromonu przez  $k$ -tą mrówkę na danej krawędzi jest obliczana ze wzoru:  $\frac{Q}{\text{koszt krawędzi}}$ . Na ilość pozostawionego feromonu przez mrówkę ma wpływ długość krawędzi – im dłuższa tym mniej feromonu pozostawia mrówka, natomiast im krótsza tym więcej. Stanowi to różnicę w stosunku do algorytmu ACOCAS, gdzie ilość pozostawianego feromonu na całej ścieżce dla danej mrówki była taka sama.

#### 4. Testowane instancje

Zostały przetestowane instancje symetryczne i asymetryczne. Moje instancje testowe o  $V > 13$ , pochodzą z TSPLIB, natomiast te mniejsze zostały wygenerowane przeze mnie w poprzednich zadaniach i obliczone również zostały im drogi optymalne, by móc policzyć błąd rozwiązań dla testów z zadania 3 i 4. Największymi instancjami są: dla grafów symetrycznych instancja o 200 wierzchołkach i instancja o 171 wierzchołkach dla asymetrycznych.

W zadaniu 3 i 4 zdecydowałem się testować instancje większe niż w zadaniu 1 i 2 ze względu na charakterystykę testowanych algorytmów (błąd otrzymany dla małych instancji w zad 3 i 4 był niemal zawsze równy 0 i czas również bliski 0).

#### 5. Hipotezy badawcze:

1. W algorytmach w zadaniu 3 generowanie początkowego rozwiązania (nn i random) ma znaczący wpływ na błąd rozwiązania uzyskanego przez algorytm.
2. W algorytmie symulowanego wyżarzania dobór odpowiednich parametrów metody ma kluczowy wpływ na jakość i czas otrzymania rozwiązania.
3. W algorytmie przeszukiwania z zakazami dobór odpowiednich parametrów metody ma kluczowy wpływ na jakość i czas otrzymania rozwiązania.
4. Algorytmy mrówkowe osiągają mniejsze błędy niż algorytmy symulowanego wyżarzania i przeszukiwania z zakazami.
5. W algorytmach mrówkowych parametry alfa (odpowiadająca za znaczenie śladu feromonowego) i beta (preferowanie przez mrówki krawędzi w zależności od jej długości) powinny zostać odpowiednio dobrane, tak aby zachować balans i algorytm przynosił jak najlepsze rezultaty.
6. Algorytmy testowane w zadaniu 3 i 4 osiągają lepsze wyniki czasowe i są w stanie otrzymywać stosunkowo małe błędy rozwiązania w stosunku do algorytmów testowanych w zadaniu 1 i 2 (BF, Random, DFS, BFS, Lowest Cost)

## Punkt 2 – instancje i specyfikacja

**TABELA NR.1.1 ORAZ 1.2 PRZEDSTAWIA NAZWĘ INSTANCJI DLA GRAFU SYMETRYCZNEGO I ASYMETRYCZNEGO, LICZBĘ WIERZCHOŁKÓW ORAZ WARTOŚĆ OPTYMALNĄ.**

Nazwa instancji(sym)	Liczba wierzchołków	Wartość optymalna
file_5s.txt	5	98
file_6s.txt	6	118
file_7s.txt	7	90
file_8s.txt	8	115
file_9s.txt	9	105
file_10s.txt	10	126
file_11s.txt	11	156
file_12s.txt	12	177
file_13s.txt	13	174
file_14s.txt (tsplib burma14)	14	3323
file_29s.txt (tsplib bays29)	29	1610
file_100s.txt (tsplib kroA100)	100	21282
file_137s.txt (tsplib gr137)	137	69853
file_150s.txt (tsplib kroB150)	150	26130
file_200s.txt (tsplib kroB200)	200	29437

Nazwa instancji (asym)	Liczba wierzchołków	Wartość optymalna
file_5a.txt	5	200
file_6a.txt	6	355
file_7a.txt	7	580
file_8a.txt	8	925
file_9a.txt	9	1375
file_10a.txt	10	1985
file_11a.txt	11	2775
file_12a.txt	12	3765
file_13a.txt	13	4985
file_17a.txt (tsplib br17)	17	77
file_34a.txt (tsplib ftv33)	34	1286
file_65a.txt (tsplib ftv64)	65	1839
file_100a.txt (tsplib kro124)	100	36230
file_171a.txt (tsplib ftv170)	171	2755

### Specyfikacja sprzętowa:

Procesor: AMD Ryzen 5 5500U

Liczba rdzeni: 6

Maksymalne taktowanie zegara: 4,2 GHz

Wielkość pamięci cache: 19MB

Wielkość pamięci RAM: 16GB

Prezentacja programu odbywała się na tym samym sprzęcie, co jego testy.

Laptop miał odpalone testy podczas zasilania go prądem.

W trakcie działania testów nie działały inne programy na komputerze.

## Punkt 3 – Opis procedury badawczej

### **Ilość badanych instancji oraz liczba iteracji:**

W moich badaniach, używałem 9 instancji symetrycznych i 9 asymetrycznych do 13 wierzchołków. Poza tym 5 instancji większych – asymetrycznych oraz 6 dla symetrycznych grafów. Łącznie 29 instancji.

Ustawiony limit czasowy był 15 minutowy. Dodatkowo moje czasy we wszystkich tabelkach i te, które są ukazane na wszystkich wykresach to średnia z 5 pomiarów.

Osobno na wykresach przedstawiłem wyniki dla grafów symetrycznych i asymetrycznych. Błędy zaznaczyłem na tabelkach kolorem czerwonym, mogły one wynikać z charakterystyki danej instancji, która była badana.

Dodatkowo w algorytmach z zad 3 procent dolnego ograniczenia dla instancji <11 było równe 0%, dla instancji <21 było równe 10, natomiast dla większych - 20%.

Zaś w algorytmach z zad 4 procent dolnego ograniczenia dla instancji <25 było równe 0%, dla instancji <75 było równe 3, natomiast dla większych - 8%.

Wybrany procent dolnego ograniczenia nie był przypadkowy – algorytmy mrówkowe działały dużo lepiej (pod względem błędu) niż SA i TS szczególnie dla rozwiązania początkowego inicjowanego przez algorytm nn.

Dla TS i SA iteracje bez poprawy to 200.

Warto również wspomnieć, że w algorytmach mrówkowych liczba mrówka była tożsama z liczbą wierzchołków grafu a wartość feromonu  $Q = 5$ . Iteracje bez poprawy to 10.

## Punkt 4 – Wyniki badań

Wyniki w tabelkach zaznaczone na kolor czerwony są błędami, które nie pasują do trendu wyznaczonego przez pozostałe pomiary.

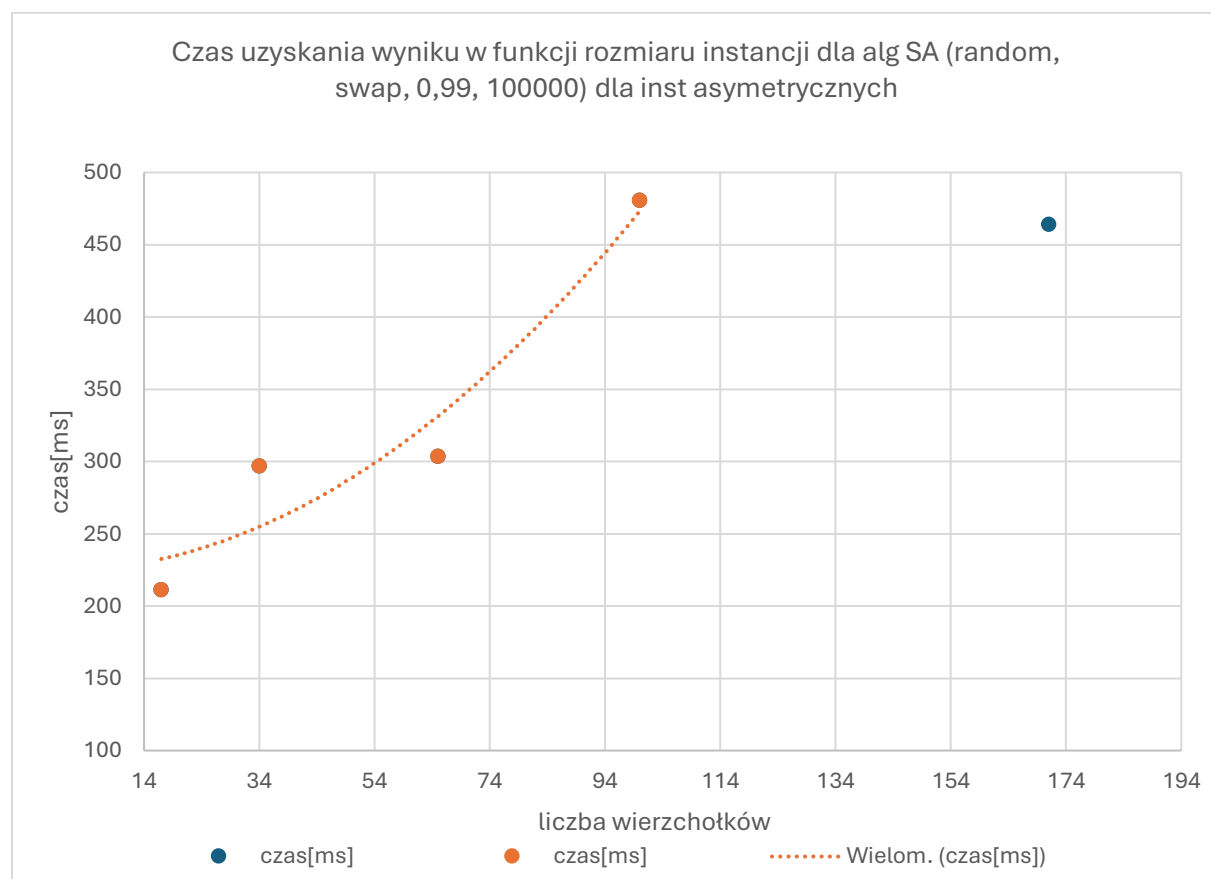
Litera V w tabelach oznacza rozmiar badanej instancji.

W podpisach wybranych metod np. SA(random, swap, 0,99, 1000) – wartości w nawiasie to parametry z jakimi była testowana dana metoda.

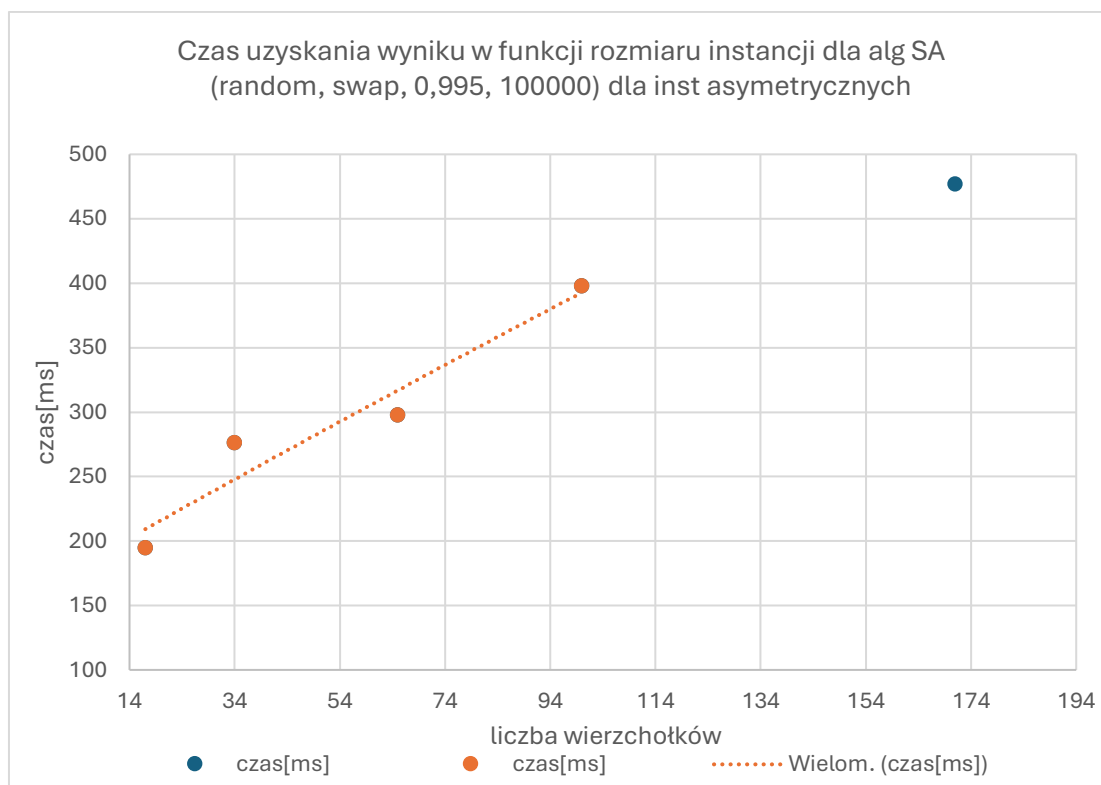
### WYNIKI ALGORYTMU SA – TABELKI I WYKRESY:

**TABELA 2 PRZEDSTAWIA CZAS WYKONANIA ALGORYTMU SA DLA INSTANCJI ASYMETRYCZNYCH DLA CHŁODZENIA GEOMETRYCZNEGO DLA RÓŻNYCH WSPÓŁCZYNNIKÓW ALFA**

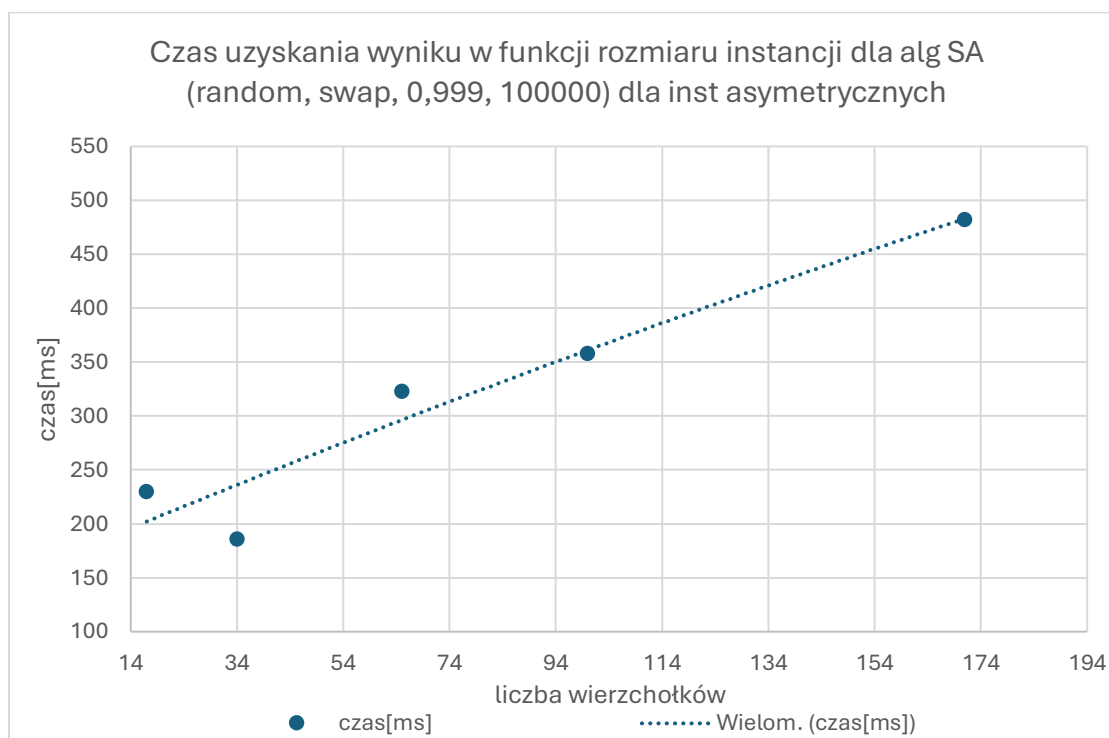
alfa	0,99	0,995	0,999	0,99	0,995	0,999
V	czas[ms]			błąd[%]		
17	211,07	194,57	230,09	138,7	148,31	127,53
34	296,7	276,33	185,69	143,58	141,99	146,31
65	303,4	297,87	322,8	276,85	276,27	281,03
100	480,58	397,93	357,98	323,48	322,13	324,68
171	463,91	476,86	481,95	729,92	733,39	733,65



**WYKRES 1 PRZEDSTAWIA ZALEŻNOŚĆ CZASU WYKONANIA ALGORYTMU OD WIELKOŚCI GRAFU DLA SA DLA INSTANCJI ASYMETRYCZNYCH**

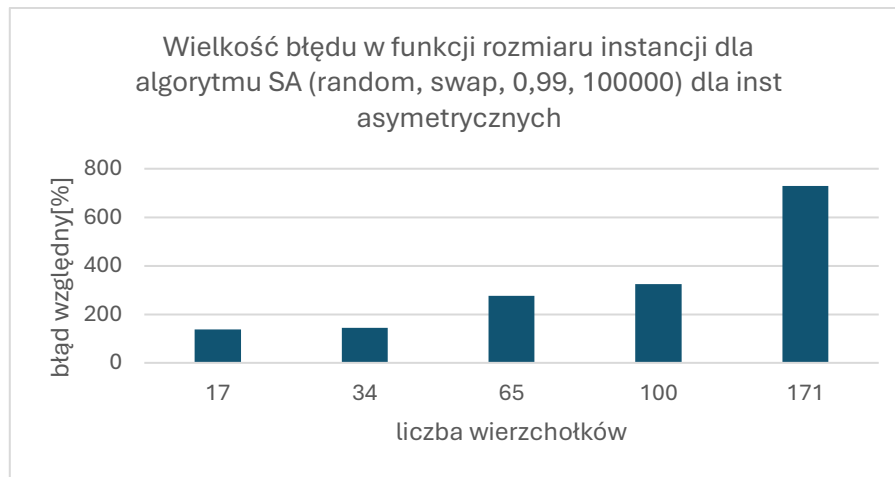


**WYKRES 2 PRZEDSTAWIA ZALEŻNOŚĆ CZASU WYKONANIA ALGORYTMU SA OD WIELKOŚCI GRAFU SA DLA INSTANCJI ASYMETRYCZNYCH**

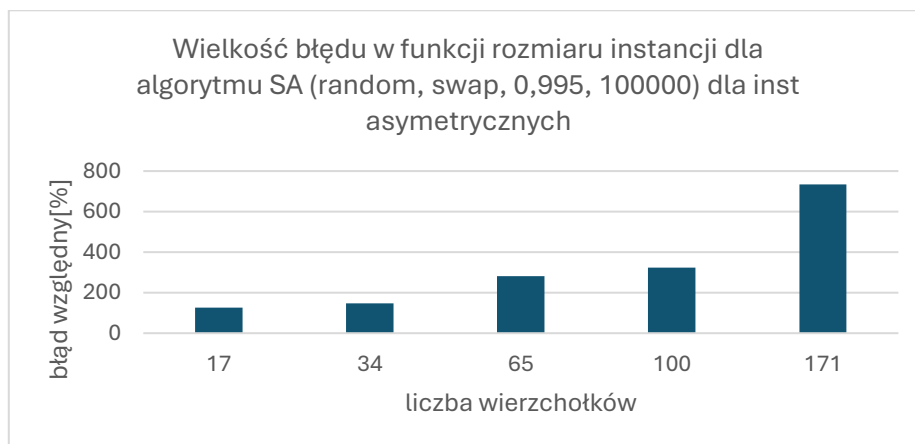


**WYKRES 3 PRZEDSTAWIA ZALEŻNOŚĆ CZASU WYKONANIA ALGORYTMU SA OD WIELKOŚCI GRAFU SA DLA INSTANCJI ASYMETRYCZNYCH**

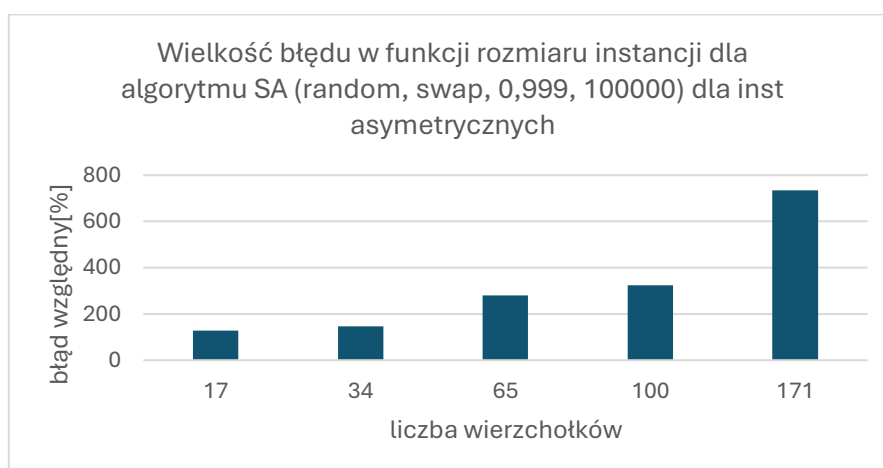




**WYKRES 4 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI ASYMETRYCZNYCH**



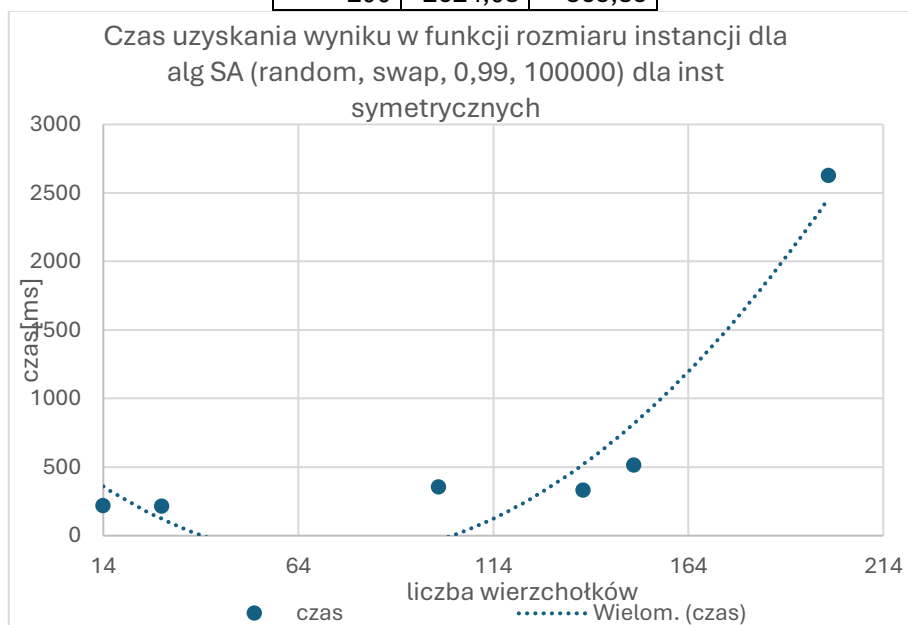
**WYKRES 5 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI ASYMETRYCZNYCH**



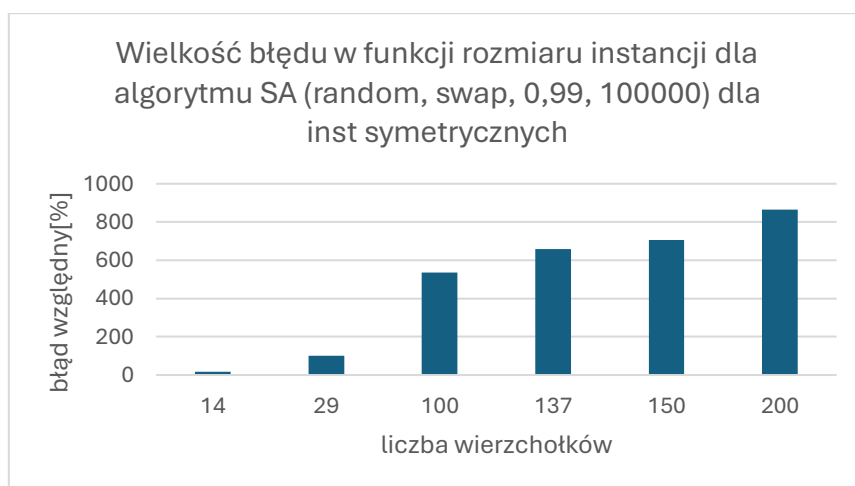
**WYKRES 6 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI ASYMETRYCZNYCH**

**TABELA 3 PRZEDSTAWIA CZAS WYKONANIA ALGORYTMU SA(RANDOM, SWAP, 0,99, 100000) DLA INSTANCJI SYMETRYCZNYCH**

V	czas[ms]	błąd[%]
14	215,68	16,55
29	213,4	99,91
100	351,41	535,09
137	329,95	658,41
150	511,94	705,03
200	2624,03	865,35



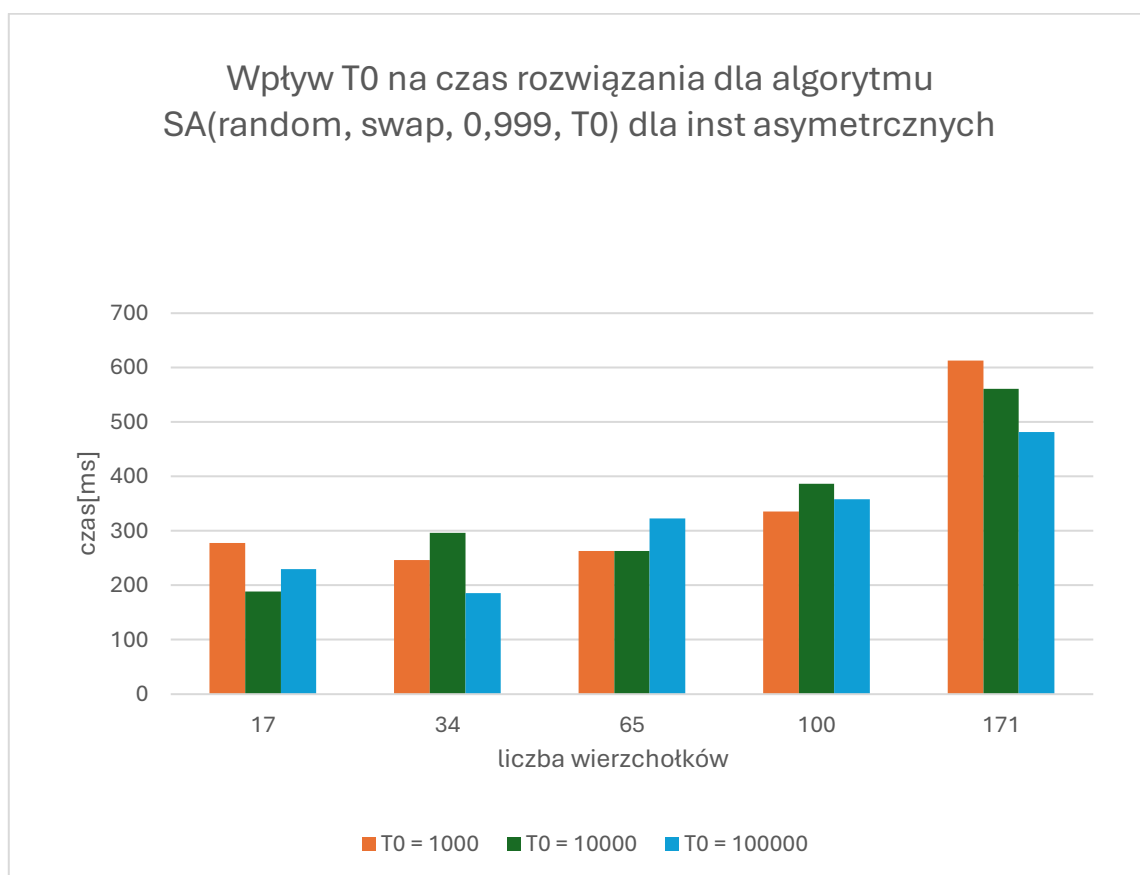
**WYKRES 7 PRZEDSTAWIA ZALEŻNOŚĆ CZASU WYKONANIA OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI SYMETRYCZNYCH**



**WYKRES 8 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU OD WIELKOŚCI GRAFU WYKONANIA ALGORYTMU SA DLA INSTANCJI SYMETRYCZNYCH**

**TABELA 4 PRZEDSTAWIA CZAS WYKONANIA ALGORYTMU SA DLA INSTANCJI ASYMETRYCZNYCH DLA RÓŻNYCH TEMPERATUR POCZĄTKOWYCH**

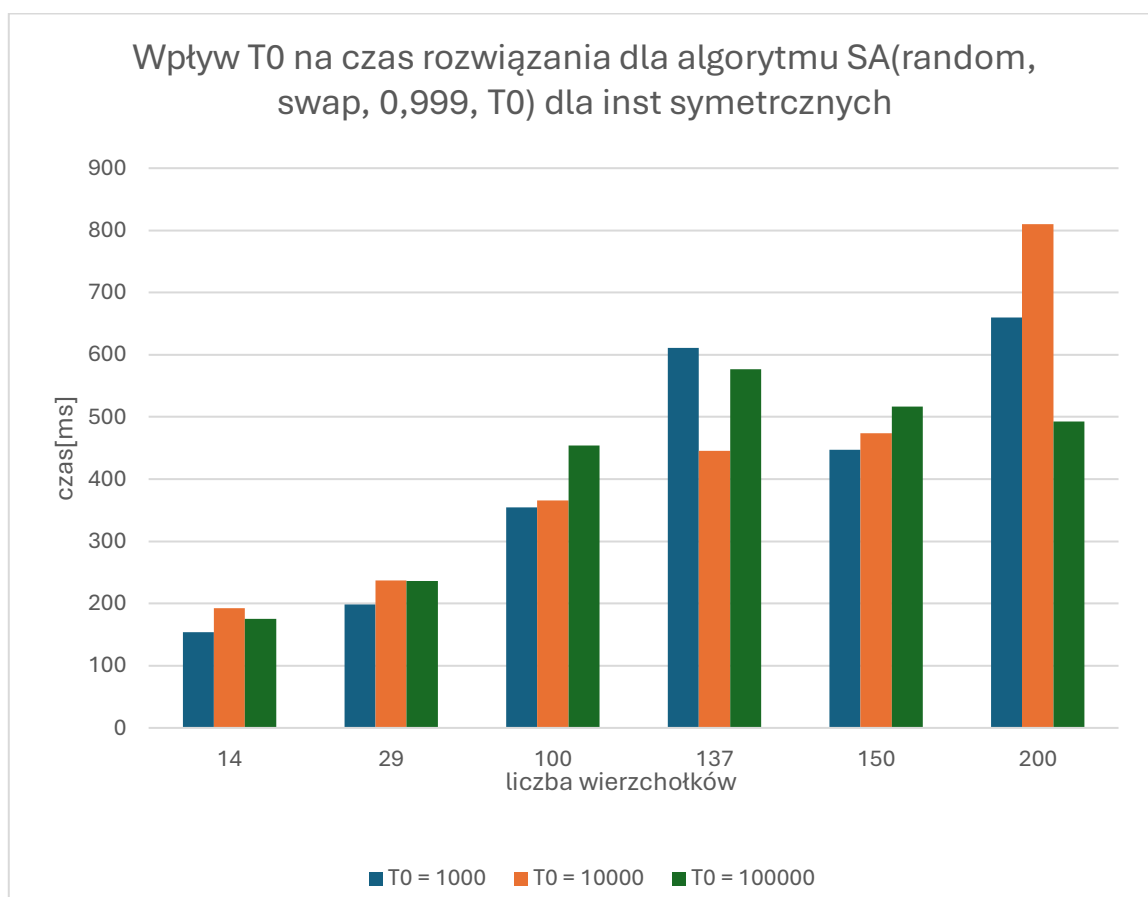
VT	czas[ms]		
	1000	10000	100000
17	278,02	188,9	230,09
34	246,77	296,67	185,69
65	262,92	263,43	322,8
100	335,99	386,85	357,98
171	612,5	560,7	481,95



**WYKRES 9 PRZEDSTAWIA ZALEŻNOŚĆ CZASU WYKONANIA OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI ASYMETRYCZNYCH DLA RÓŻNYCH TEMPERATUR POCZĄTKOWYCH**

**TABELA 5 PRZEDSTAWIA CZAS WYKONANIA ALGORYTMU SA DLA INSTANCJI SYMETRYCZNYCH DLA RÓŻNYCH TEMPERATUR POCZĄTKOWYCH**

VT	Czas[ms]		
	1000	10000	100000
17	278,02	188,9	230,09
34	246,77	296,67	185,69
65	262,92	263,43	322,8
100	335,99	386,85	357,98
171	612,5	560,7	481,95

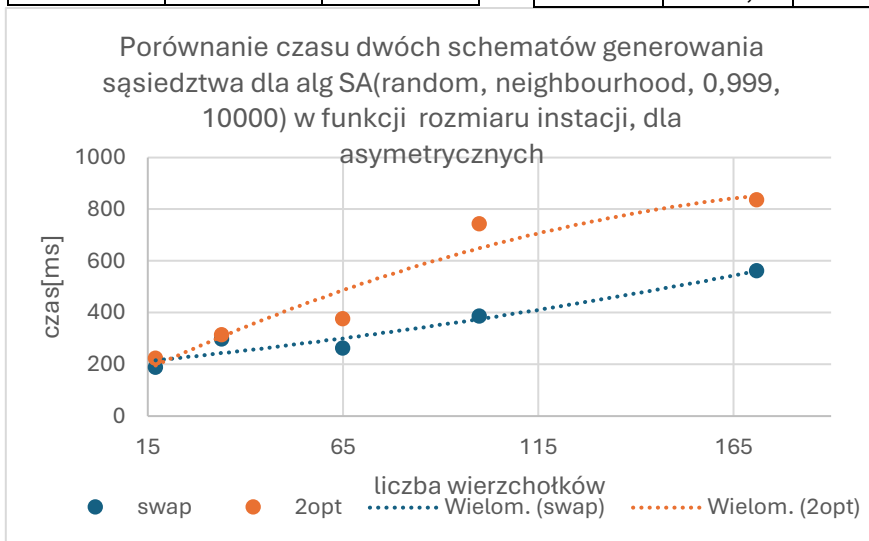


**WYKRES 10 PRZEDSTAWIA ZALEŻNOŚĆ CZASU WYKONANIA OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI SYMETRYCZNYCH DLA RÓŻNYCH TEMPERATUR POCZĄTKOWYCH**

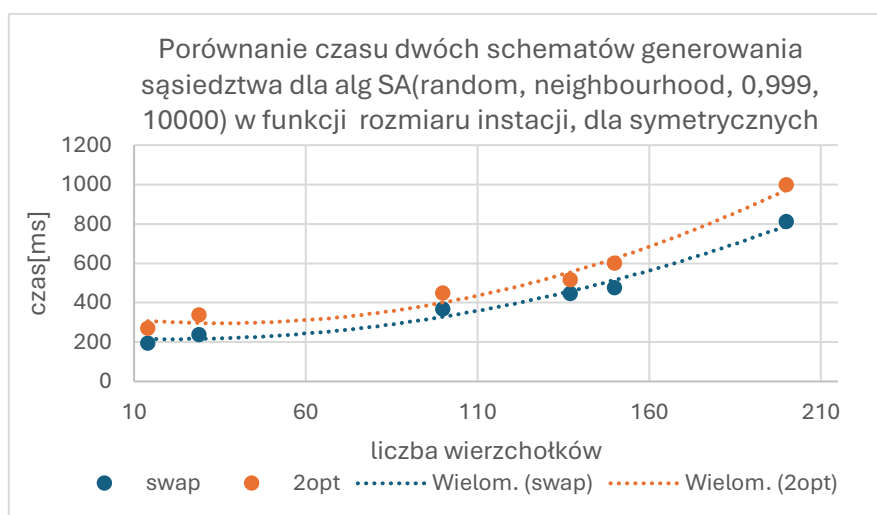
**TABELA 6.1 I 6.2 PRZEDSTAWIA ODPOWIEDNIO CZAS WYKONANIA ALGORYTMU SA DLA INSTANCJI ASYMETRYCZNYCH I SYMETRYCZNYCH DLA RÓŻNYCH SPOSOBÓW GENEROWANIA SĄSIEDZTW**

	swap	2opt
V(asym)	Czas[ms]	
17	188,9	222,63
34	296,67	314,92
65	263,43	375,83
100	386,85	742,31
171	560,7	836,31

	swap	2opt
V(sym)	Czas[ms]	
14	192,45	267,28
29	236,7	336,59
100	366,05	446,53
137	445,6	513,85
150	473,49	597,62
200	810,01	995,77



**WYKRES 11 PRZEDSTAWIA ZALEŻNOŚĆ CZASU WYKONANIA OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI ASYMETRYCZNYCH DLA RÓŻNYCH SPOSOBÓW GENEROWANIA SĄSIEDZTW**

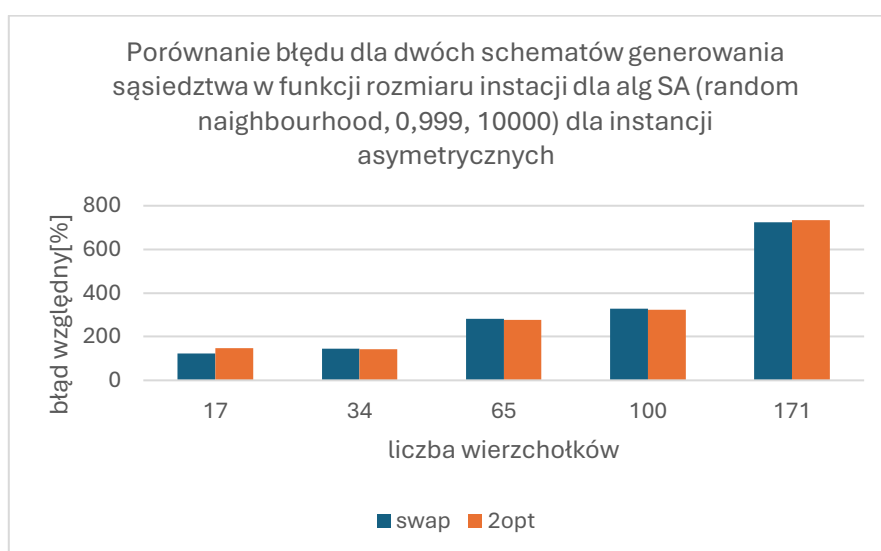


**WYKRES 12 PRZEDSTAWIA ZALEŻNOŚĆ CZASU WYKONANIA OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI SYMETRYCZNYCH DLA RÓŻNYCH SPOSOBÓW GENEROWANIA SĄSIEDZTW**

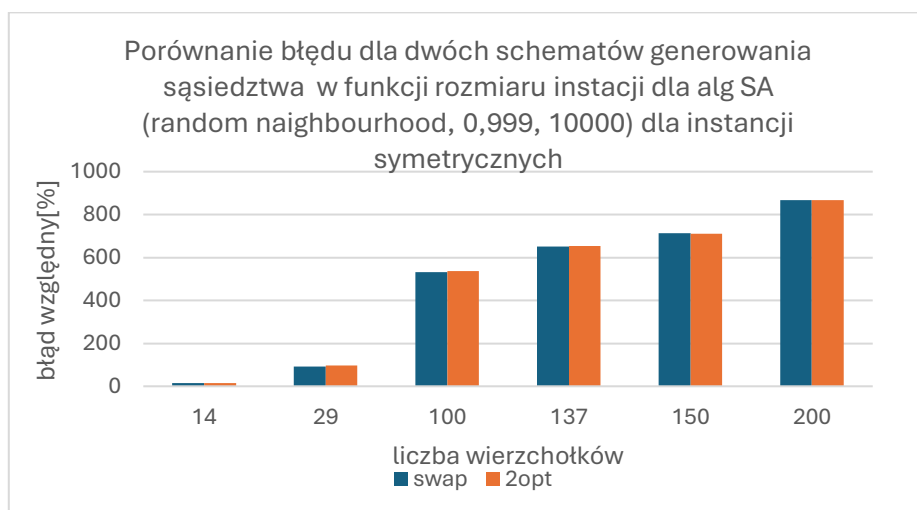
**TABELA 7.1 I 7.2 PRZEDSTAWIA ODPOWIEDNIO BŁĘDY WZGLĘDNE W % WYKONANIA ALGORYTMU SA DLA INSTANCJI ASYMETRYCZNYCH I SYMETRYCZNYCH DLA RÓŻNYCH SPOSOBÓW GENEROWANIA SĄSIEDZTW**

	geo	log
V	Czas[ms]	
17	124,16	147,01
34	143,97	142,49
65	281,98	277,44
100	327,1	324,09
171	723,93	733,96

	swap	2opt
V	Czas[ms]	
14	13,85	15,39
29	92,93	97,47
100	530,81	536,9
137	651,77	652,9
150	714,35	711,01
200	867,25	866,48



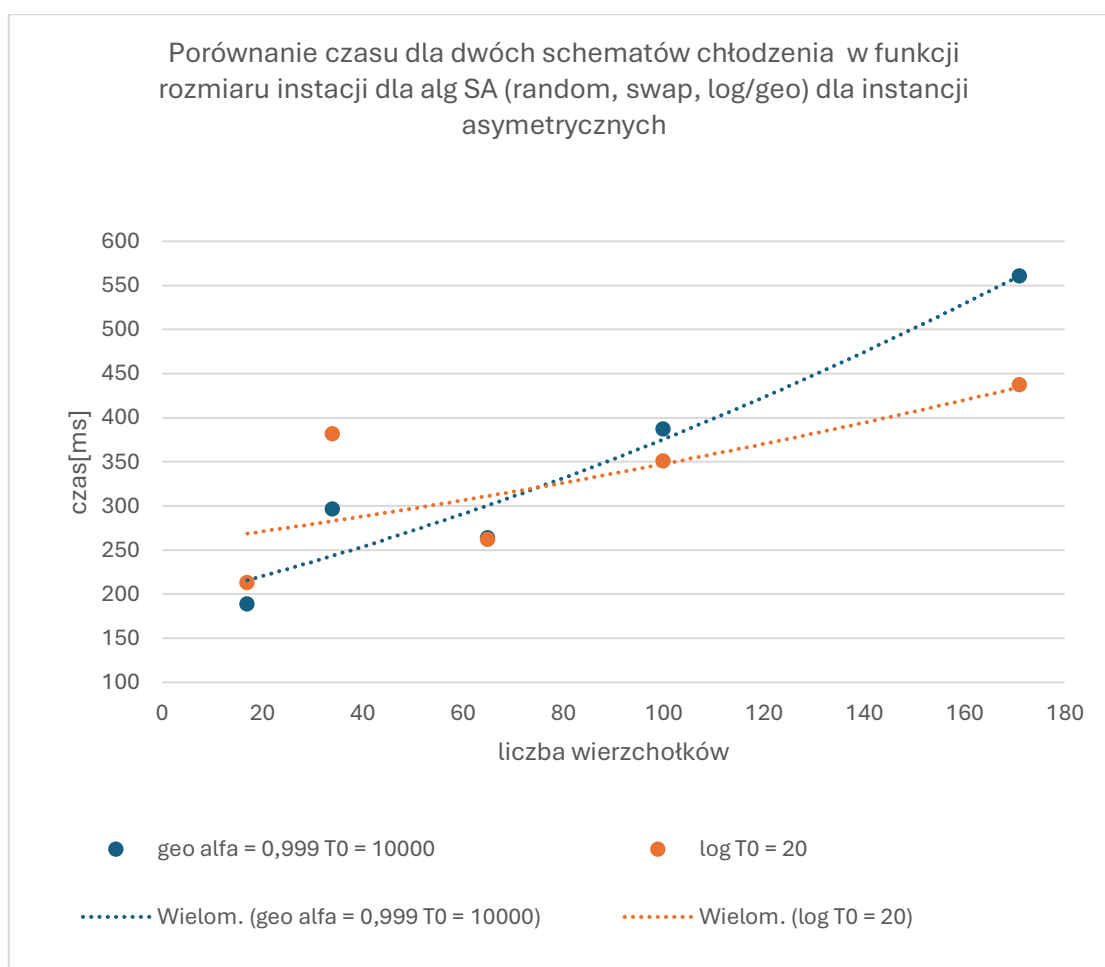
**WYKRES 13 PRZEDSTAWIA ZALEŻNOŚĆ CZASU WYKONANIA OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI ASYMETRYCZNYCH DLA RÓŻNYCH SPOSOBÓW GENEROWANIA SĄSIEDZTW**



**WYKRES 14 PRZEDSTAWIA ZALEŻNOŚĆ CZASU WYKONANIA OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI SYMETRYCZNYCH DLA RÓŻNYCH SPOSOBÓW GENEROWANIA SĄSIEDZTW**

**TABELA 8 PRZEDSTAWIA ZALEŻNOŚĆ CZASU WYKONANIA OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI ASYMETRYCZNYCH DLA RÓŻNYCH SCHEMATÓW CHŁODZENIA**

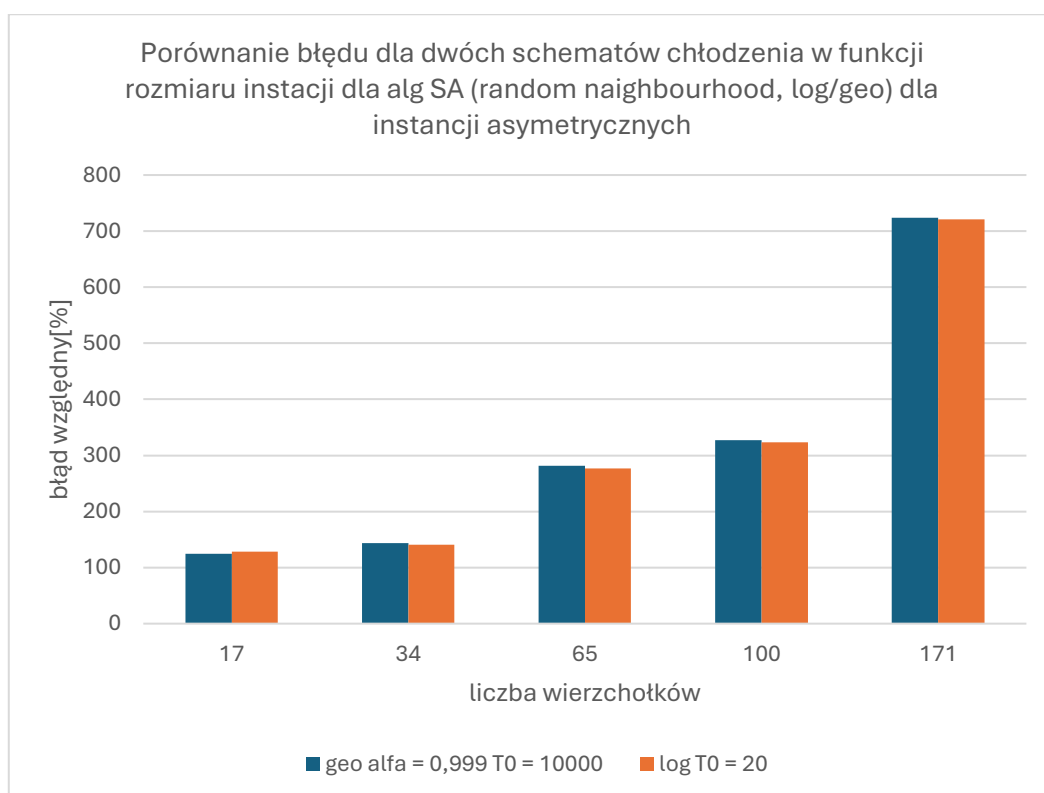
	geo alfa = 0,999 T0 = 10000	log T0 = 20
V	czas[ms]	
17	188,9	213,23
34	296,67	381,4
65	263,43	262
100	386,85	351,09
171	560,7	437,3



**WYKRES 15 PRZEDSTAWIA ZALEŻNOŚĆ CZASU WYKONANIA OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI ASYMETRYCZNYCH DLA RÓŻNYCH SCHEMATÓW CHŁODZENIA**

**TABELA 9 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU WZGLĘDNEGO W % OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI ASYMETRYCZNYCH DLA RÓŻNYCH SCHEMATÓW CHŁODZENIA**

	błąd względny[%]	
V	geo alfa = 0,999 T0 = 10000	log T0 = 20
17	124,16	128,83
34	143,97	140,45
65	281,98	276,37
100	327,1	322,93
171	723,93	721,1

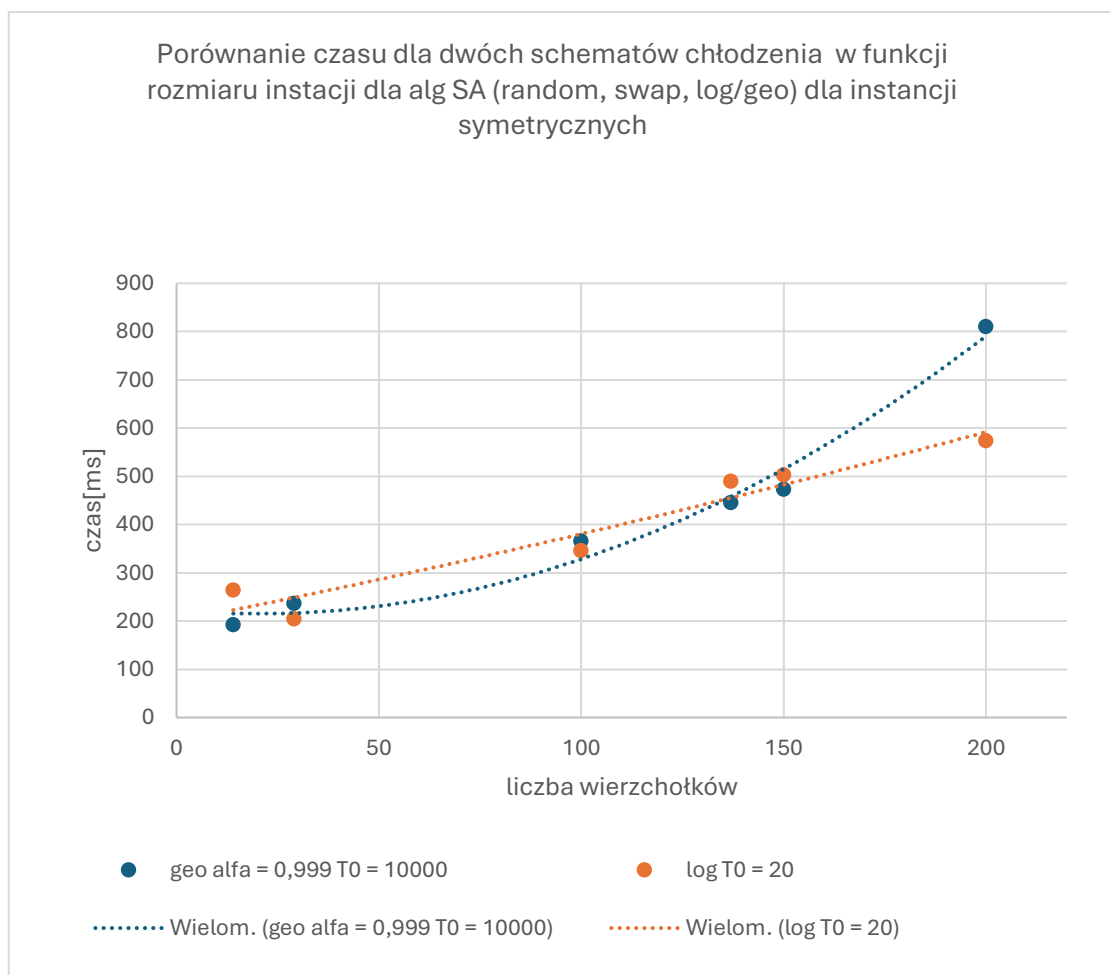


**WYKRES 16 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU WZGLĘDNEGO W % OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI ASYMETRYCZNYCH DLA RÓŻNYCH SCHEMATÓW CHŁODZENIA**



**TABELA 10 PRZEDSTAWIA ZALEŻNOŚĆ CZASU WYKONANIA OD WIELKOŚCI GRAFU  
ALGORYTMU SA DLA INSTANCJI SYMETRYCZNYCH DLA RÓŻNYCH SCHEMATÓW CHŁODZENIA**

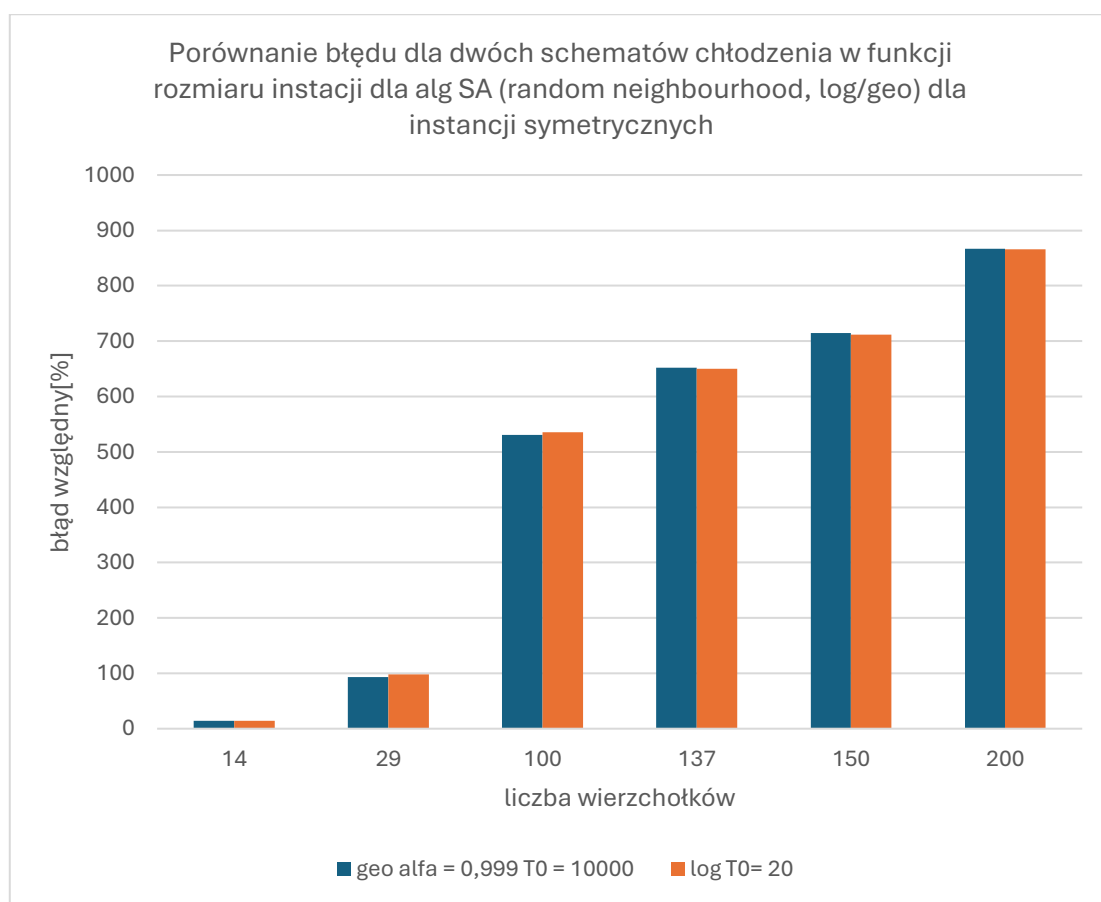
	czas[ms]	
V	geo alfa = 0,999 T0 = 10000	log T0 = 20
14	192,45	264
29	236,7	205,03
100	366,05	346,53
137	445,6	489,32
150	473,49	502,83
200	810,01	573,48



**WYKRES 17 PRZEDSTAWIA ZALEŻNOŚĆ CZASU WYKONANIA OD WIELKOŚCI GRAFU  
ALGORYTMU SA DLA INSTANCJI SYMETRYCZNYCH DLA RÓŻNYCH SCHEMATÓW CHŁODZENIA**

**TABELA 11 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU WZGLĘDNEGO W % OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI SYMETRYCZNYCH DLA RÓŻNYCH SCHEMATÓW CHŁODZENIA**

V	błąd względny[%]	
	geo alfa = 0,999 T0 = 10000	log T0= 20
14	13,85	14,41
29	92,93	97,88
100	530,81	535,6
137	651,77	649,94
150	714,35	712,09
200	867,25	865,58

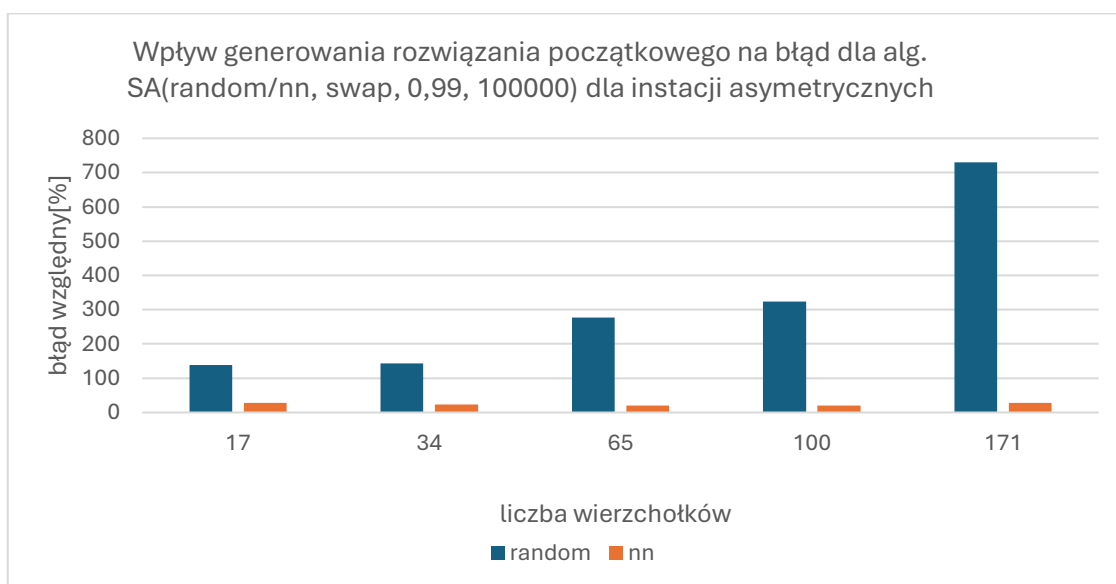


**WYKRES 18 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU WZGLĘDNEGO W % OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI SYMETRYCZNYCH DLA RÓŻNYCH SCHEMATÓW CHŁODZENIA**

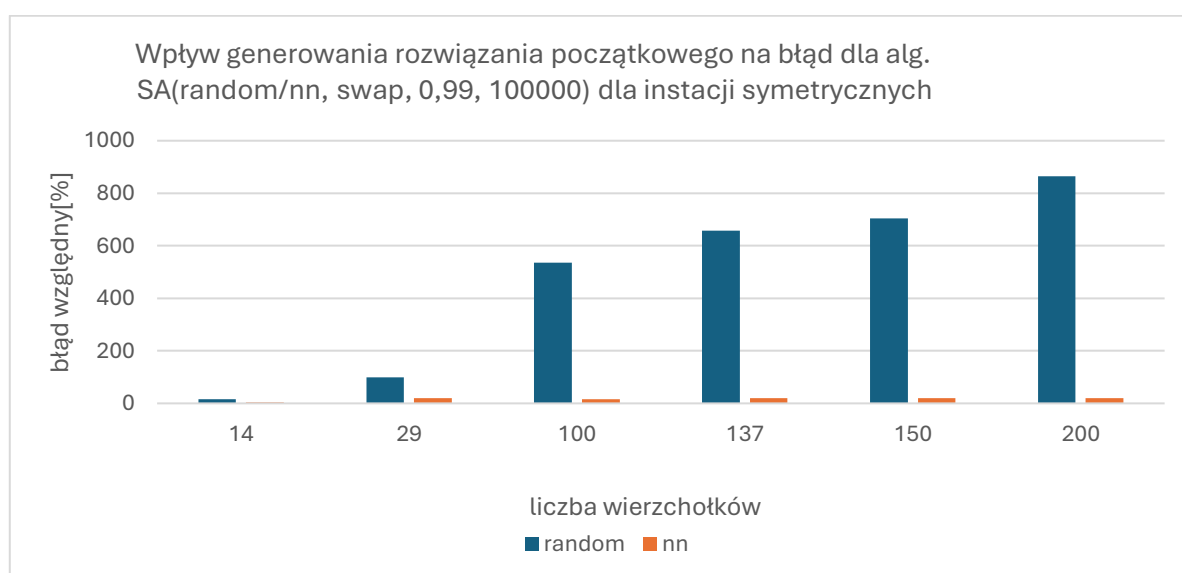
**TABELA 12.1 I 12.2 PRZEDSTAWIA WPLYW ROZWIĄZANIA POCZĄTKOWEGO NA BŁĄD OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI ODPOWIEDNIO ASYMETRYCZNYCH I SYMETRYCZNYCH**

	random	nn
V	błąd[%]	
17	138,7	27,27
34	143,58	23,64
65	276,85	19,74
100	323,48	19,56
171	729,92	28,2

	random	nn
V	błąd[%]	
14	16,55	2,65
29	99,91	20,19
100	535,09	16,05
137	658,41	20,83
150	705,03	19,86
200	865,35	19,82



**WYKRES 19 PRZEDSTAWIA WPLYW ROZWIĄZANIA POCZĄTKOWEGO NA BŁĄD OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI ASYMETRYCZNYCH**

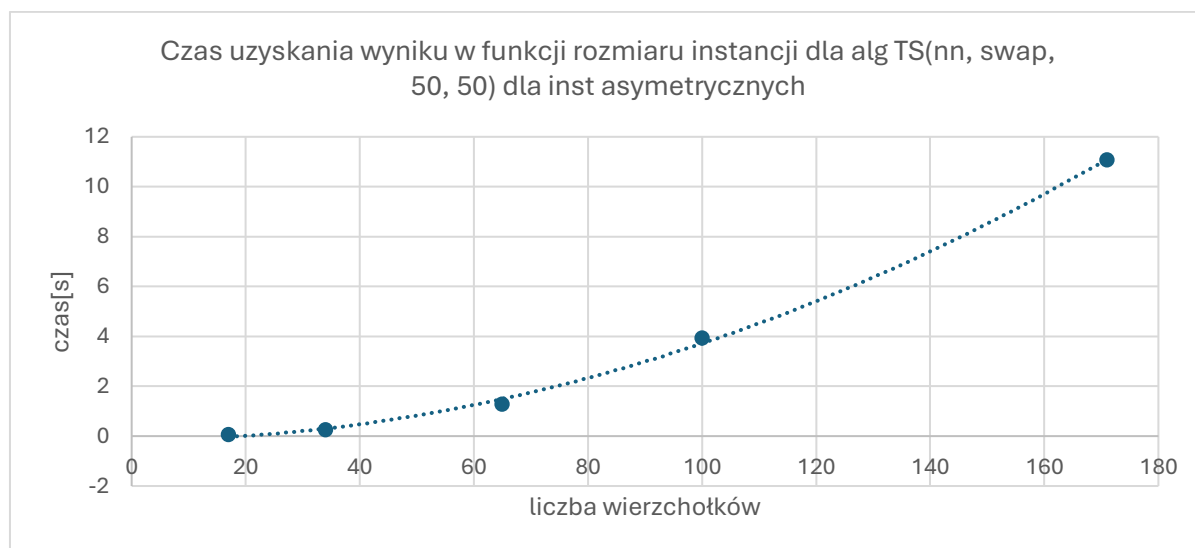


**WYKRES 20 PRZEDSTAWIA WPLYW ROZWIĄZANIA POCZĄTKOWEGO NA BŁĄD OD WIELKOŚCI GRAFU ALGORYTMU SA DLA INSTANCJI SYMETRYCZNYCH**

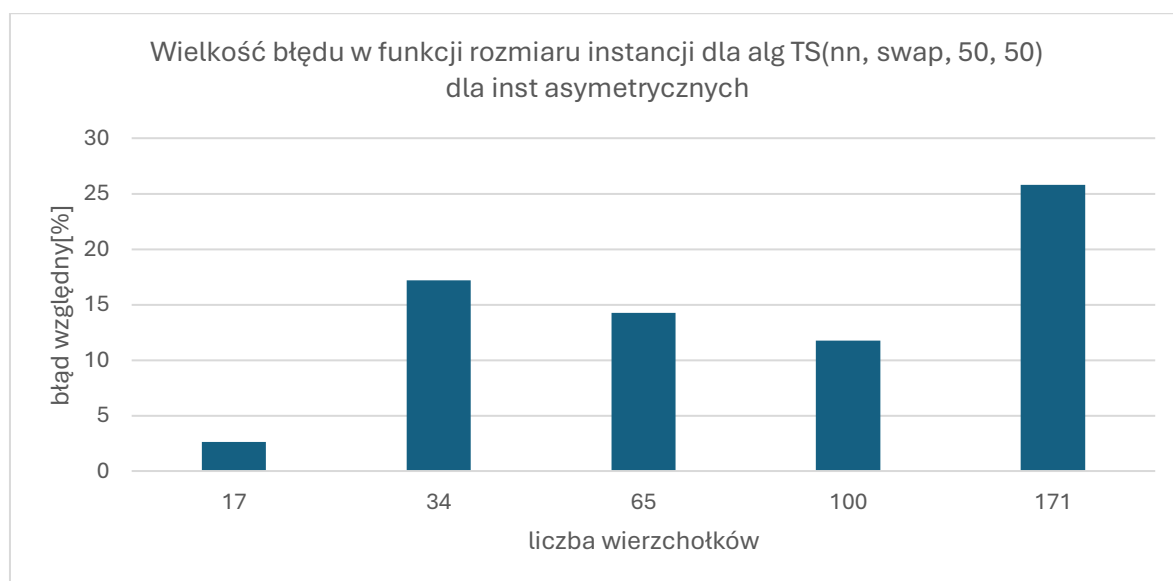
## ALGORYTM TABU SEARCH – TABELE I WYKRESY:

**TABELA 13 PRZEDSTAWIA CZAS I BŁĄD WYKONANIA ALGORYTMU TABU SEARCH(NN, SWAP, 50, 50) ODPowiednio dla instancji asymetrycznych**

V	czas[s]	błąd[%]
17	0,06	2,6
34	0,25	17,19
65	1,27	14,25
100	3,92	11,78
171	11,06	25,77



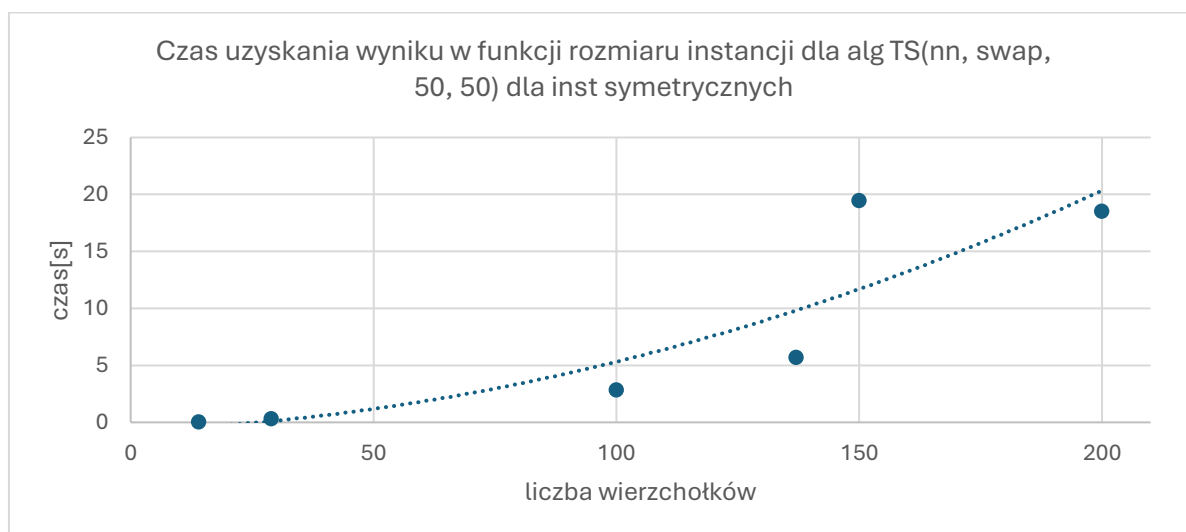
**WYKRES 21 PRZEDSTAWIA ZALEŻNOŚĆ CZASU OD WIELKOŚCI GRAFU ALGORYTMU TS DLA INSTANCJI ASYMETRYCZNYCH**



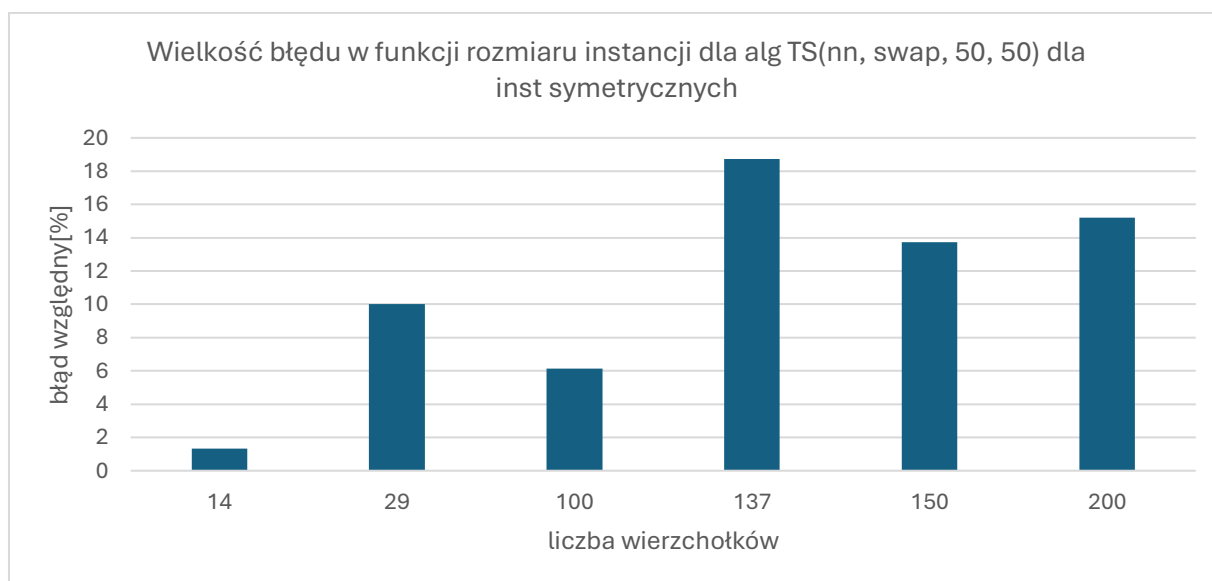
**WYKRES 22 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU WZGLĘDNEGO W % OD WIELKOŚCI GRAFU ALGORYTMU TS DLA INSTANCJI ASYMETRYCZNYCH**

**TABELA 14 PRZEDSTAWIA CZAS I BŁĄD WYKONANIA ALGORYTMU TABU SEARCH(NN, SWAP, 50, 50) ODPowiednio dla instancji symetrycznych**

V	czas[s]	błąd[%]
14	0,04	1,32
29	0,33	10,00
100	2,86	6,12
137	5,68	18,72
150	19,44	13,74
200	18,53	15,19



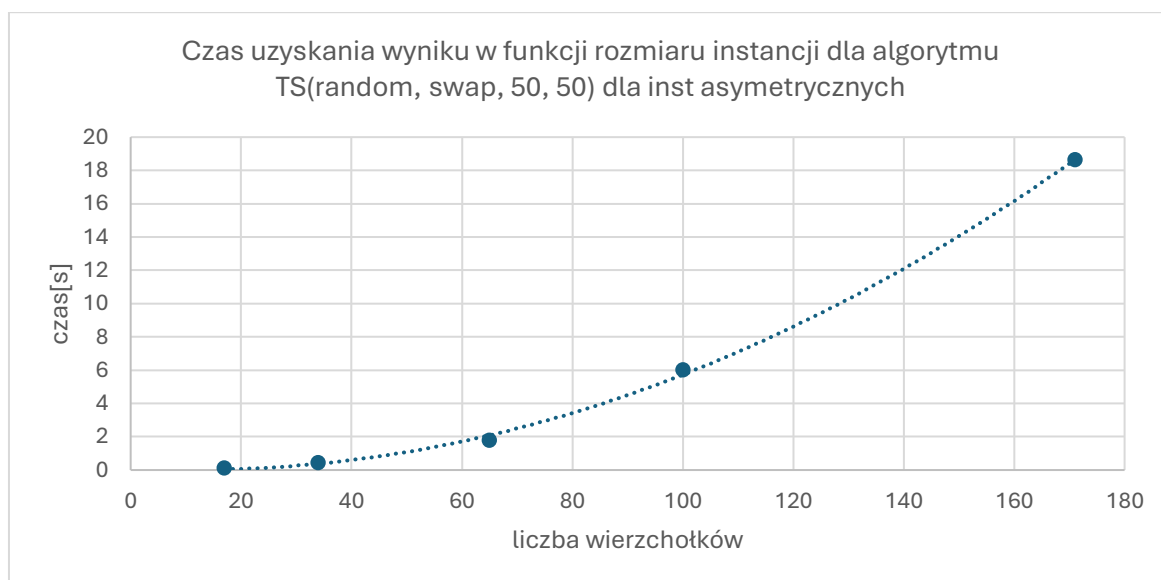
**WYKRES 23 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU WZGLĘDNEGO W % OD WIELKOŚCI GRAFU ALGORYTMU TS DLA INSTANCJI SYMETRYCZNYCH**



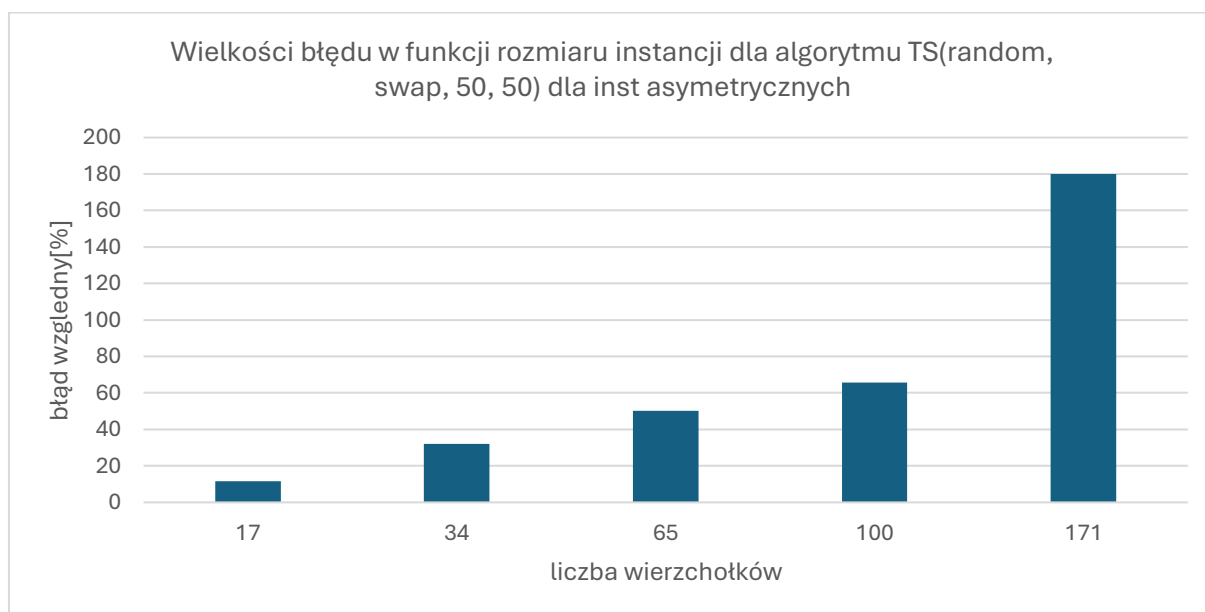
**WYKRES 24 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU WZGLĘDNEGO W % OD WIELKOŚCI GRAFU ALGORYTMU TS DLA INSTANCJI SYMETRYCZNYCH**

**TABELA 15 PRZEDSTAWIA CZAS I BŁĄD WYKONANIA ALGORYTMU TABU SEARCH(RANDOM, SWAP, 50, 50) DLA INSTANCJI ASYMETRYCZNYCH**

V	czas[s]	błąd[%]
17	0,10	11,69
34	0,42	31,94
65	1,77	50,18
100	5,99	65,56
171	18,61	180



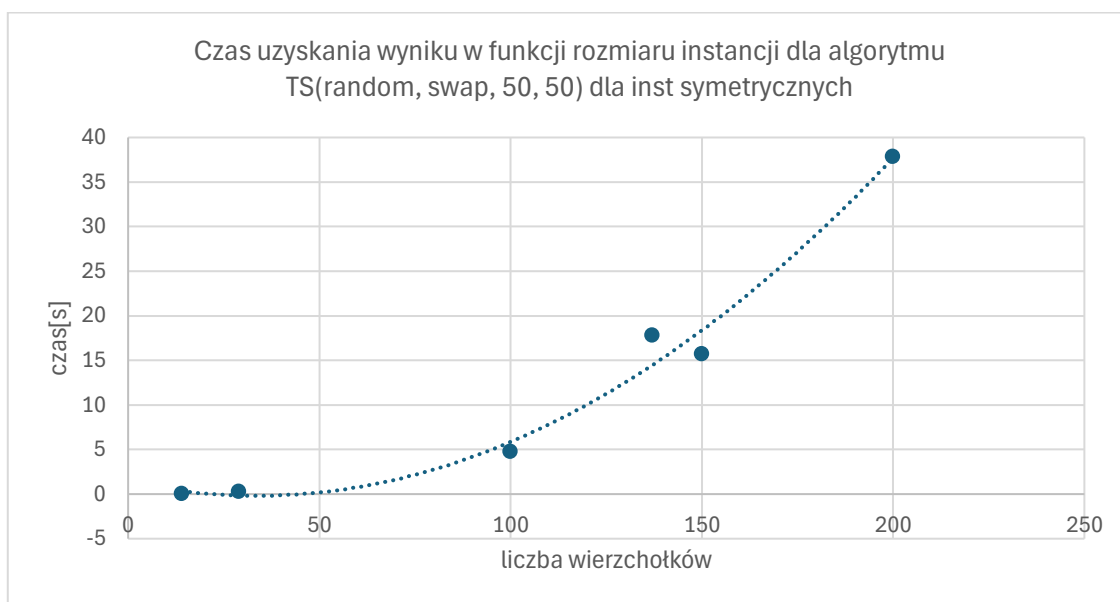
**WYKRES 25 PRZEDSTAWIA ZALEŻNOŚĆ CZASU OD WIELKOŚCI GRAFU ALGORYTMU TS DLA INSTANCJI ASYMETRYCZNYCH**



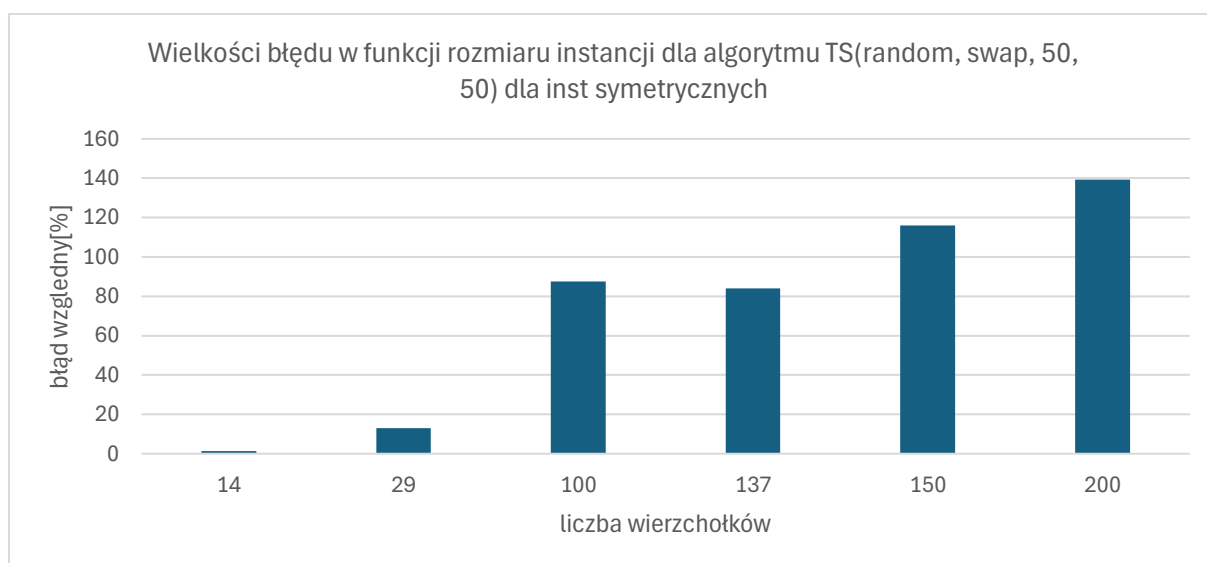
**WYKRES 26 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU WZGLĘDNEGO W % OD WIELKOŚCI GRAFU ALGORYTMU TS DLA INSTANCJI ASYMETRYCZNYCH**

**TABELA 16 PRZEDSTAWIA CZAS I BŁĄD WYKONANIA ALGORYTMU TABU SEARCH(RANDOM, SWAP, 50, 50) ODPOWIEDNIO DLA INSTANCJI SYMETRYCZNYCH**

V	czas[s]	błąd[%]
14	0,05	1,32
29	0,32	13,03
100	4,77	87,6
137	17,83	83,94
150	15,72	115,87
200	37,87	139,36



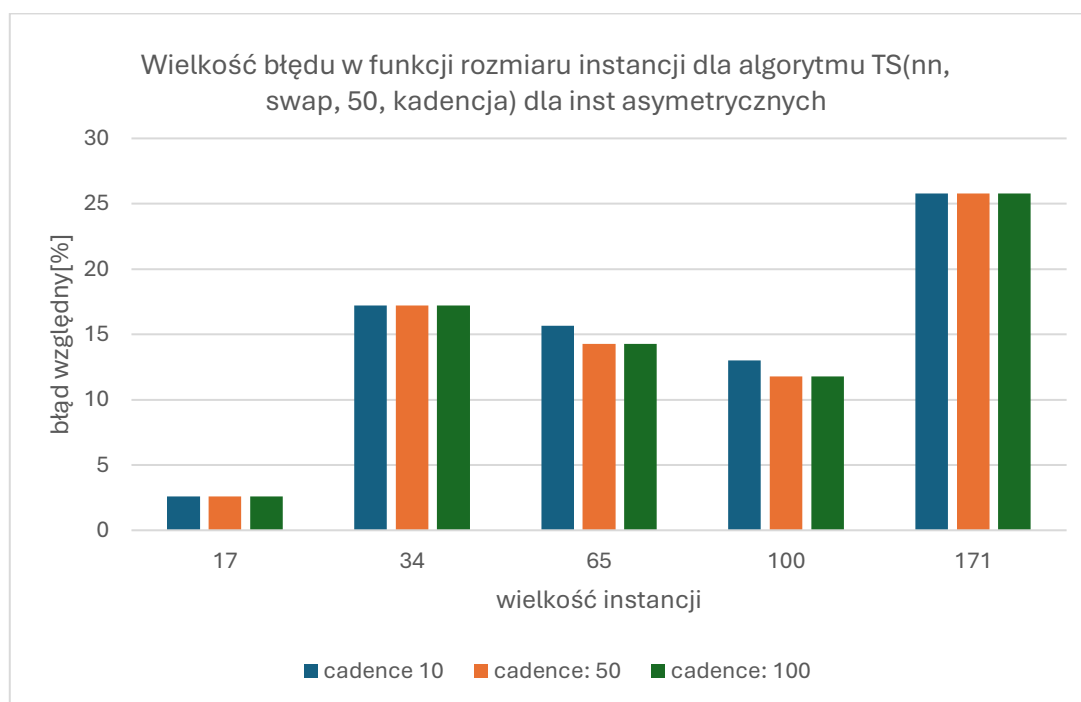
**WYKRES 27 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU WZGLĘDNEGO W % OD WIELKOŚCI GRAFU ALGORYTMU TS DLA INSTANCJI SYMETRYCZNYCH**



**WYKRES 28 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU WZGLĘDNEGO W % OD WIELKOŚCI GRAFU ALGORYTMU TS DLA INSTANCJI SYMETRYCZNYCH**

**TABELA 17 PRZEDSTAWIA BŁĄD WYKONANIA ALGORYTMU TABU SEARCH(NN, SWAP, 50, KADENCJA) DLA RÓŻNYCH KADENCJI DLA INSTANCJI ASYMETRYCZNYCH**

	cadence 10	cadence: 50	cadence: 100
V	błąd[%]		
17	2,6	2,6	2,6
34	17,19	17,19	17,19
65	15,66	14,25	14,25
100	13	11,78	11,78
171	25,77	25,77	25,77

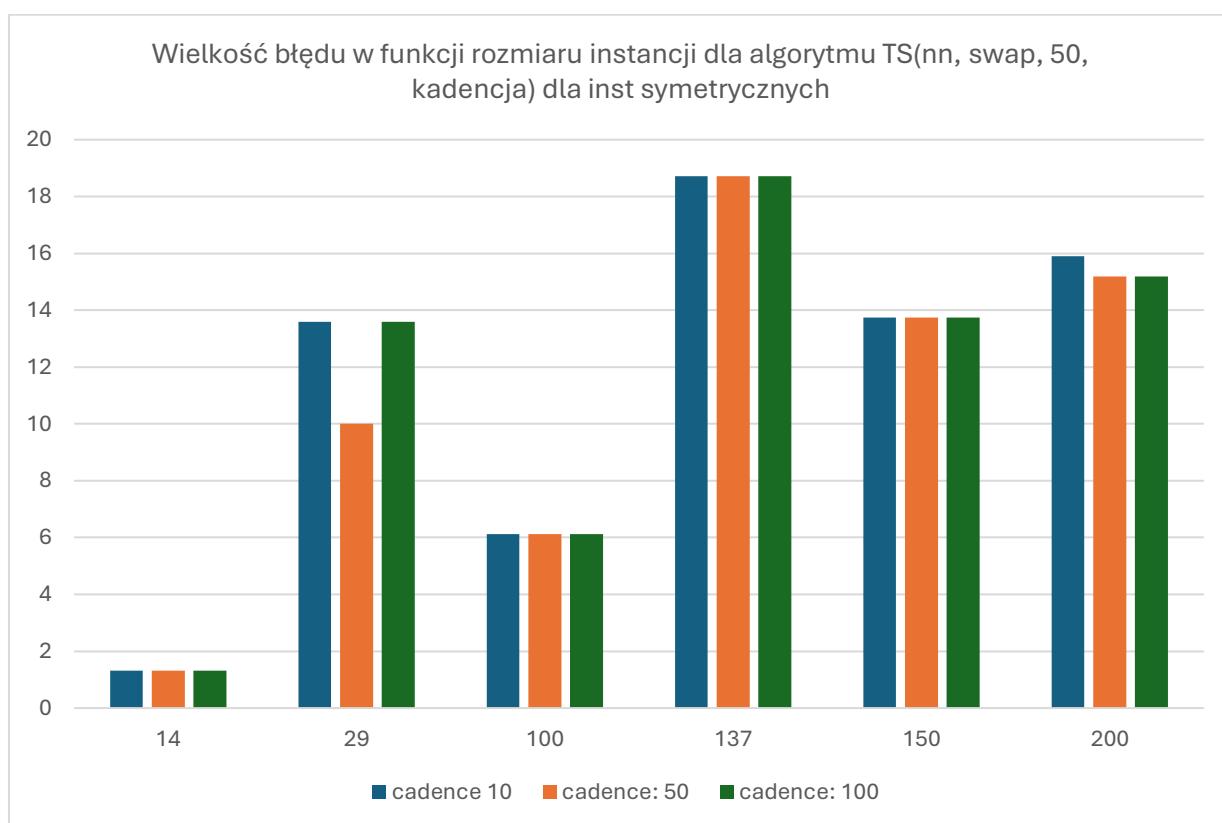


**WYKRES 29 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU WZGLĘDNEGO W % OD WIELKOŚCI GRAFU ALGORYTMU TS DLA RÓŻNYCH KADENCJI DLA INSTANCJI ASYMETRYCZNYCH**



**TABELA 18 PRZEDSTAWIA BŁĄD WYKONANIA ALGORYTMU TABU SEARCH(NN, SWAP, 50, KADENCJA) DLA RÓŻNYCH KADENCJI DLA INSTANCJI SYMETRYCZNYCH**

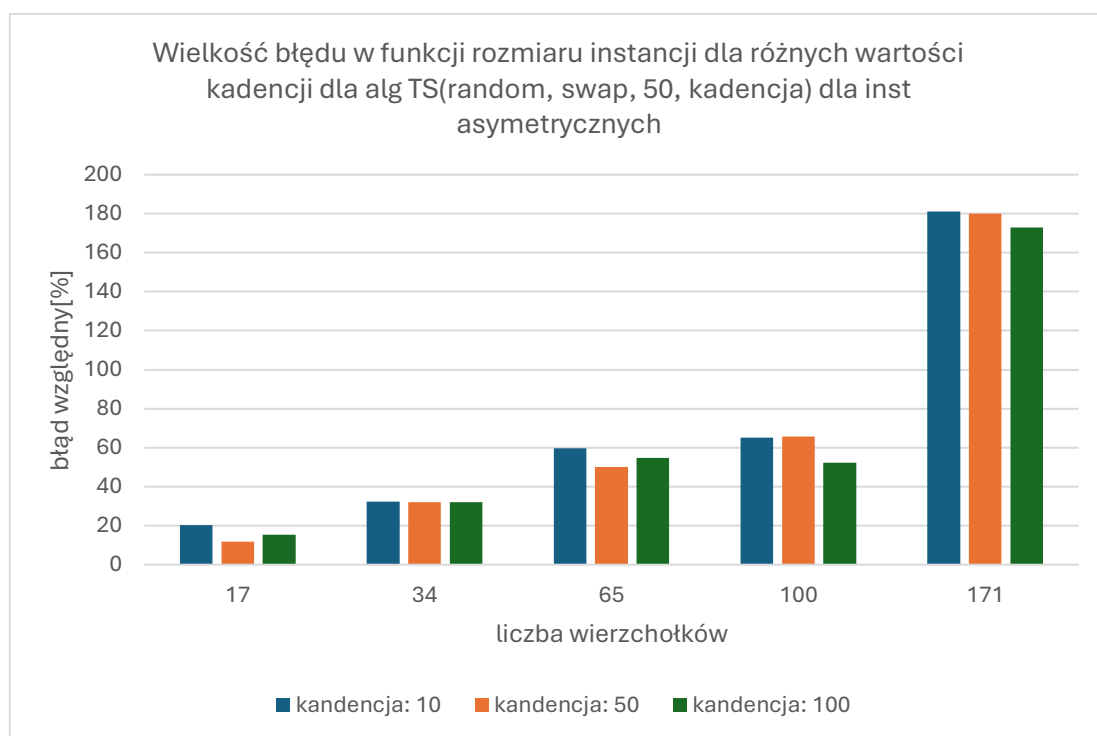
	cadence 10	cadence: 50	cadence: 100
V	błąd[%]		
14	1,32	1,32	1,32
29	13,6	10	13,6
100	6,12	6,12	6,12
137	18,72	18,72	18,72
150	13,74	13,74	13,74
200	15,89	15,19	15,19



**WYKRES 30 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU WZGLĘDNEGO W % OD WIELKOŚCI GRAFU ALGORYTMU TS DLA RÓŻNYCH KADENCJI DLA INSTANCJI SYMETRYCZNYCH**

**TABELA 19 PRZEDSTAWIA BŁĄD WYKONANIA ALGORYTMU TABU SEARCH(RANDOM, SWAP, 50, KADENCJA) DLA RÓŻNYCH KADENCJI DLA INSTANCJI ASYMETRYCZNYCH**

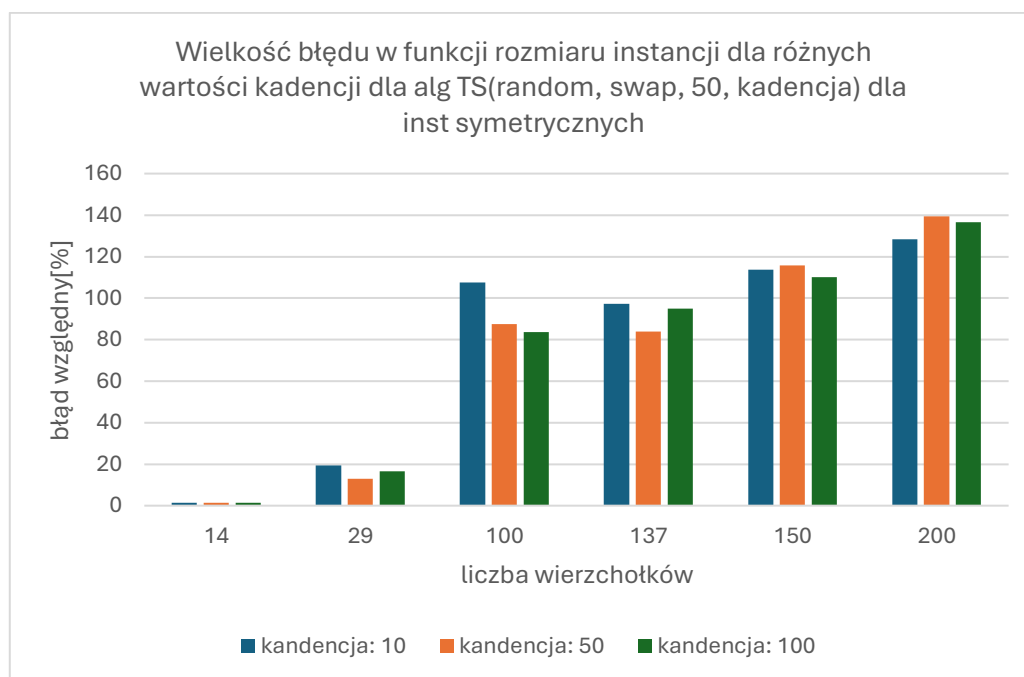
	kadencja: 10	kadencja: 50	kadencja: 100
V	błąd[%]		
17	20,26	11,69	15,32
34	32,46	31,94	31,96
65	59,63	50,18	54,75
100	65,07	65,56	52,42
171	181,23	180	172,81



**WYKRES 31 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU WZGLĘDNEGO W % OD WIELKOŚCI GRAFU ALGORYTMU TS DLA RÓŻNYCH KADENCJI DLA INSTANCJI ASYMETRYCZNYCH**

**TABELA 20 PRZEDSTAWIA BŁĄD WYKONANIA ALGORYTMU TABU SEARCH(RANDOM, SWAP, 50, KADENCJA) DLA RÓŻNYCH KADENCJI DLA INSTANCJI SYMETRYCZNYCH**

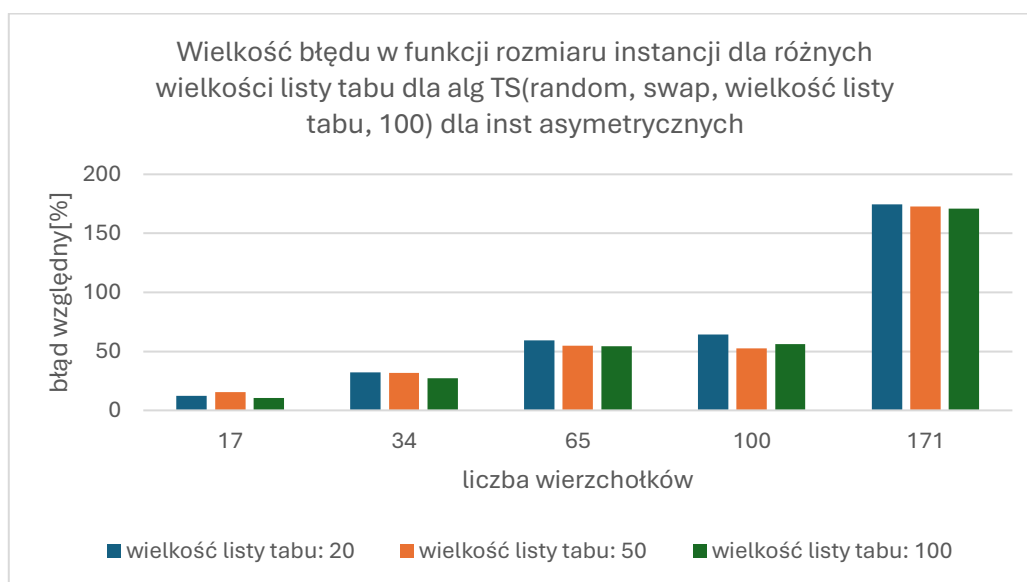
	kadencja: 10	kadencja: 50	kadencja: 100
V	błąd[%]		
14	1,32	1,32	1,32
29	19,43	13,03	16,47
100	107,56	87,6	83,69
137	97,15	83,94	95,05
150	113,82	115,87	110,19
200	128,35	139,36	136,53



**WYKRES 32 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU WZGLĘDNEGO W % OD WIELKOŚCI GRAFU ALGORYTMU TS DLA RÓŻNYCH KADENCJI DLA INSTANCJI SYMETRYCZNYCH**

**TABELA 21 PRZEDSTAWIA BŁĄD WYKONANIA ALGORYTMU TABU SEARCH(RANDOM, SWAP, WIELKOŚĆ LISTY TABU, 100) DLA RÓŻNYCH WIEŁOŚCI LISTY TABU DLA INSTANCJI ASYMETRYCZNYCH**

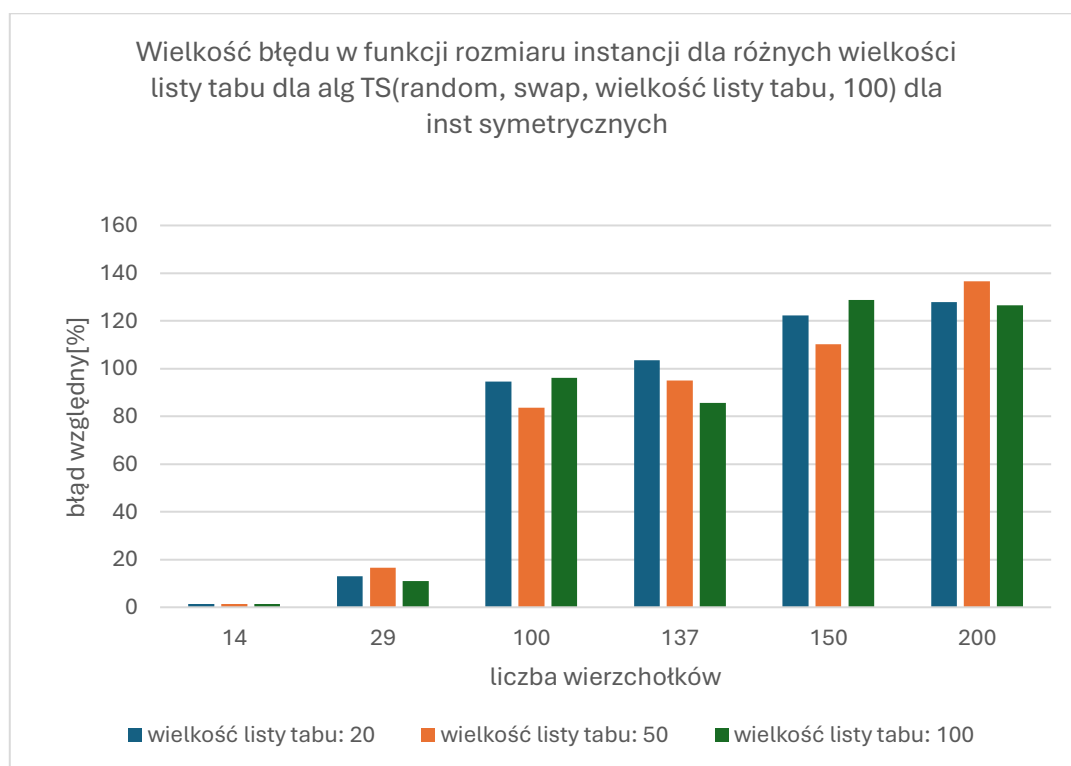
	wielkość listy tabu: 20	wielkość listy tabu: 50	wielkość listy tabu: 100
V	błąd[%]		
17	12,47	15,32	10,39
34	32,18	31,96	27,12
65	59,25	54,75	54,5
100	64,08	52,42	56,13
171	174,57	172,81	170,86



**WYKRES 33 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU WZGLĘDNEGO W % OD WIELKOŚCI GRAFU ALGORYTMU TS DLA RÓŻNYCH WIELKOŚCI LISTY TABU DLA INSTANCJI ASYMETRYCZNYCH**

**TABELA 21 PRZEDSTAWIA BŁĄD WYKONANIA ALGORYTMU TABU SEARCH(RANDOM, SWAP, WIELKOŚĆ LISTY TABU, 100) DLA RÓŻNYCH WIEŁOŚCI LISTY TABU DLA INSTANCJI SYMETRYCZNYCH**

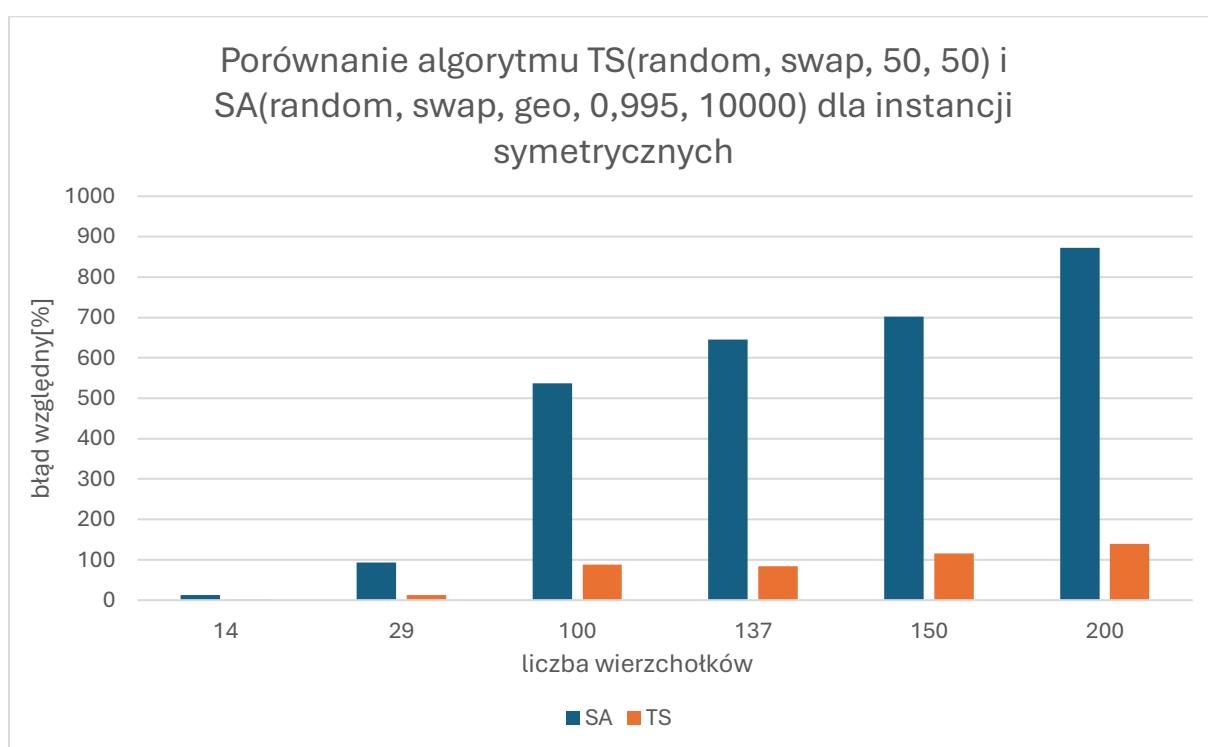
	wielkość listy tabu: 20	wielkość listy tabu: 50	wielkość listy tabu: 100
V	błąd[%]		
14	1,32	1,32	1,32
29	12,97	16,47	11,04
100	94,64	83,69	96,24
137	103,46	95,05	85,64
150	122,25	110,19	128,85
200	127,9	136,53	126,53



**WYKRES 34 PRZEDSTAWIA ZALEŻNOŚĆ BŁĘDU WZGLĘDNEGO W % OD WIELKOŚCI GRAFU ALGORYTMU TS DLA RÓŻNYCH WIELKOŚCI LISTY TABU DLA INSTANCJI SYMETRYCZNYCH**

**TABELA 22 PRZEDSTAWIA PORÓWNANIE BŁĘDU WYKONANIA ALGORYTMU TABU SEARCH(RANDOM, SWAP, 50, 50) I SA(RANDOM, SWAP, 0,995, 100000) DLA INSTANCJI SYMETRYCZNYCH**

Algorytm	SA	TS
V	błąd[%]	
14	12,55	1,32
29	93,34	13,03
100	536,78	87,6
137	645,72	83,94
150	701,77	115,87
200	872,04	139,36

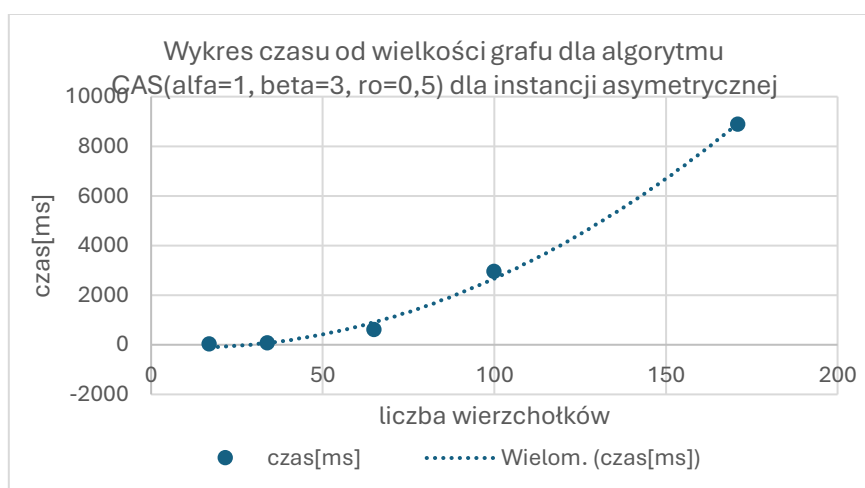


**WYKRES 35 PRZEDSTAWIA PORÓWNANIE BŁĘDU WYKONANIA ALGORYTMU TABU SEARCH(RANDOM, SWAP, 50, 50) I SA (RANDOM, SWAP, 0,995, 100000) DLA INSTANCJI SYMETRYCZNYCH**

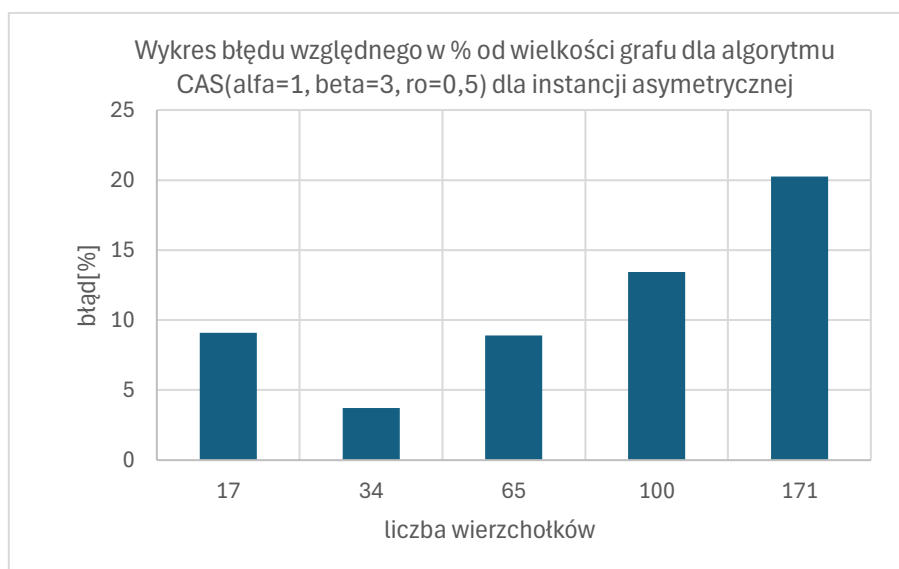
## ALGORYTMY MRÓWKOWE (CAS, DAS, QAS) – TABELKI I WYKRESY

**TABELA 23 PRZEDSTAWIA CZAS ORAZ BŁĄD DLA ALGORYTMU ACOCAS( $\alpha = 1$ ,  $\beta = 3$ ,  $\rho = 0,5$ ) DLA INST ASYMETRYCZNEJ**

V	czas[ms]	błąd[%]
17	12,07	9,09
34	54,55	3,73
65	603,4	8,91
100	2941,33	13,42
171	8866,77	20,24



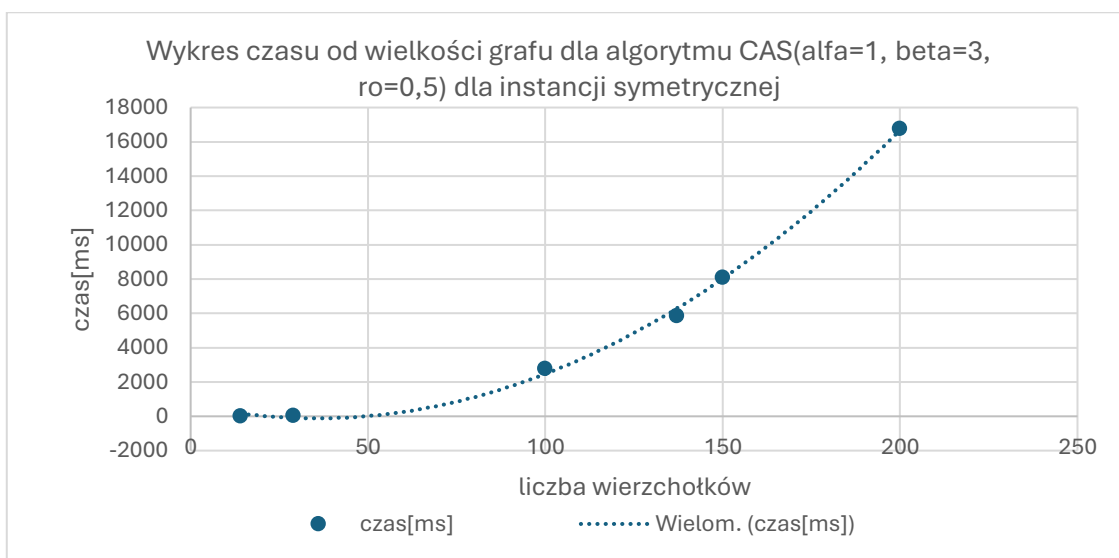
**WYKRES 36 PRZEDSTAWIA CZAS WYKONANIA W ZALEŻNOŚCI OD WIELKOŚCI GRAFU DLA ALGORYTMU ACOCAS DLA INSTANCJI ASYMETRYCZNEJ**



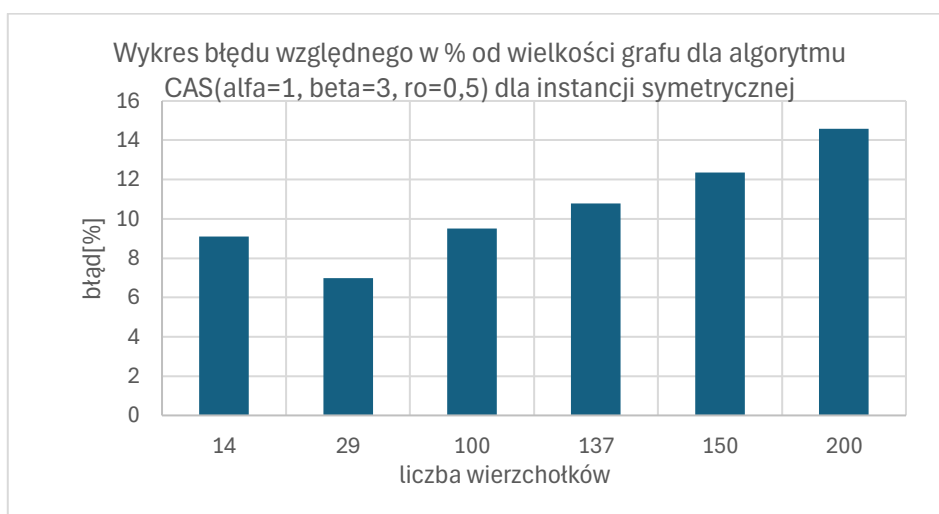
**WYKRES 37 PRZEDSTAWIA BŁĄD W ZALEŻNOŚCI OD WIELKOŚCI GRAFU DLA ALGORYTMU ACOCAS DLA INSTANCJI ASYMETRYCZNEJ**

**TABELA 24 PRZEDSTAWIA CZAS ORAZ BŁĄD DLA ALGORYTMU ACOCAS( $\alpha = 1$ ,  $\beta = 3$ ,  $\rho = 0,5$ ) DLA INST SYMETRYCZNEJ**

V	czas[ms]	błąd[%]
14	2,6	9,09
29	50,91	6,99
100	2771,72	9,52
137	5844,3	10,78
150	8090,62	12,37
200	16769,37	14,57



**WYKRES 38 PRZEDSTAWIA CZAS WYKONANIA W ZALEŻNOŚCI OD WIELKOŚCI GRAFU DLA ALGORYTMU ACOCAS DLA INSTANCJI SYMETRYCZNEJ**

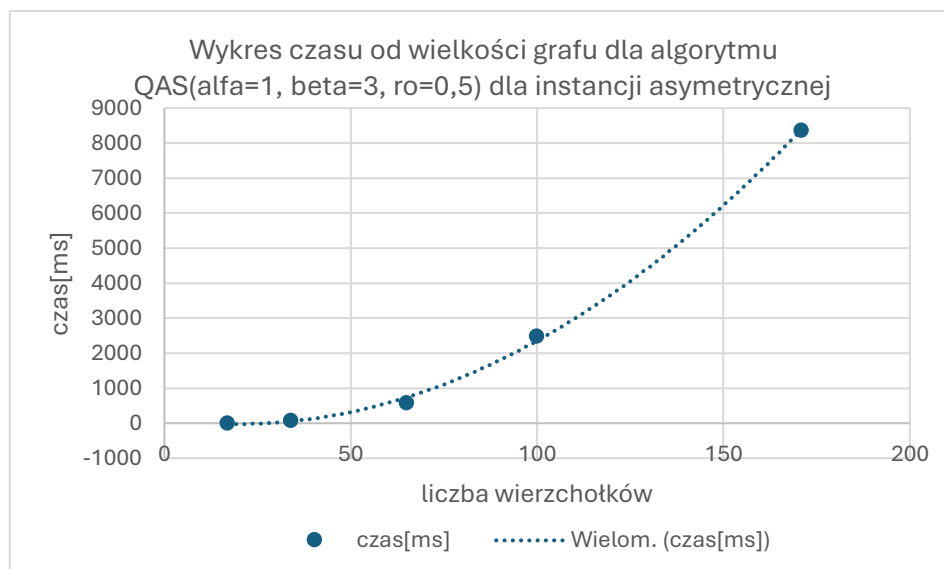


**WYKRES 39 PRZEDSTAWIA BŁĄD W ZALEŻNOŚCI OD WIELKOŚCI GRAFU DLA ALGORYTMU ACOCAS DLA INSTANCJI SYMETRYCZNEJ**

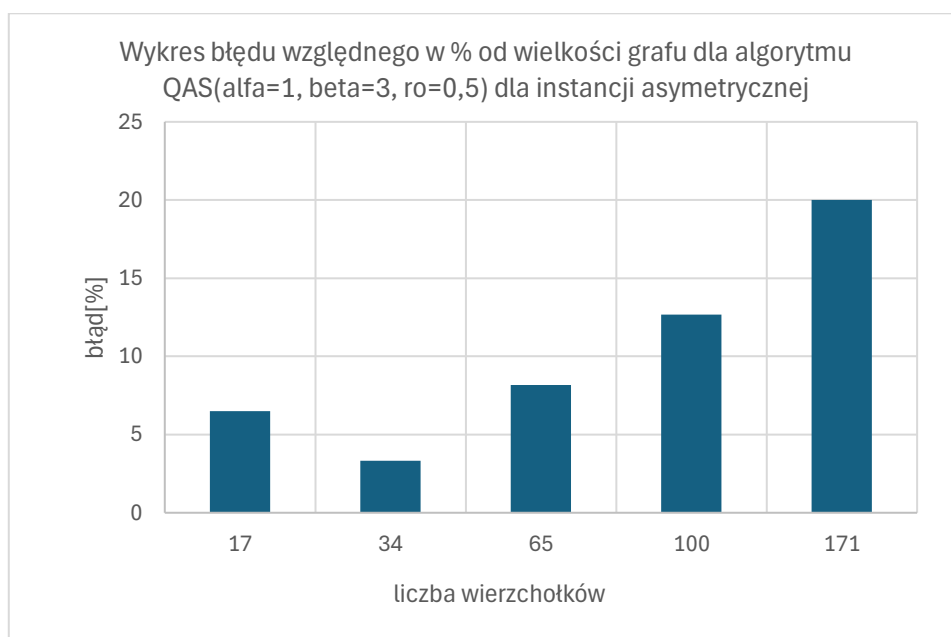


**TABELA 25 PRZEDSTAWIA CZAS ORAZ BŁĄD DLA ALGORYTMU ACOQAS( $\alpha = 1$ ,  $\beta = 3$ ,  $\rho = 0,5$ ) DLA INST ASYMETRYCZNEJ**

V	czas[ms]	błąd[%]
17	9,77	6,49
34	76,08	3,34
65	579,54	8,17
100	2484,16	12,67
171	8369,44	19,99



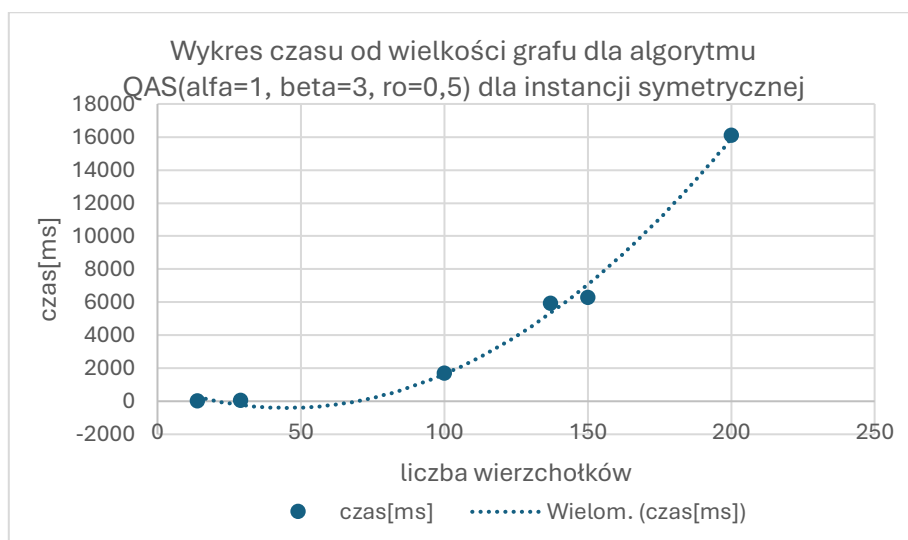
**WYKRES 40 PRZEDSTAWIA CZAS WYKONANIA W ZALEŻNOŚCI OD WIELKOŚCI GRAFU DLA ALGORYTMU ACOQAS DLA INSTANCJI ASYMETRYCZNEJ**



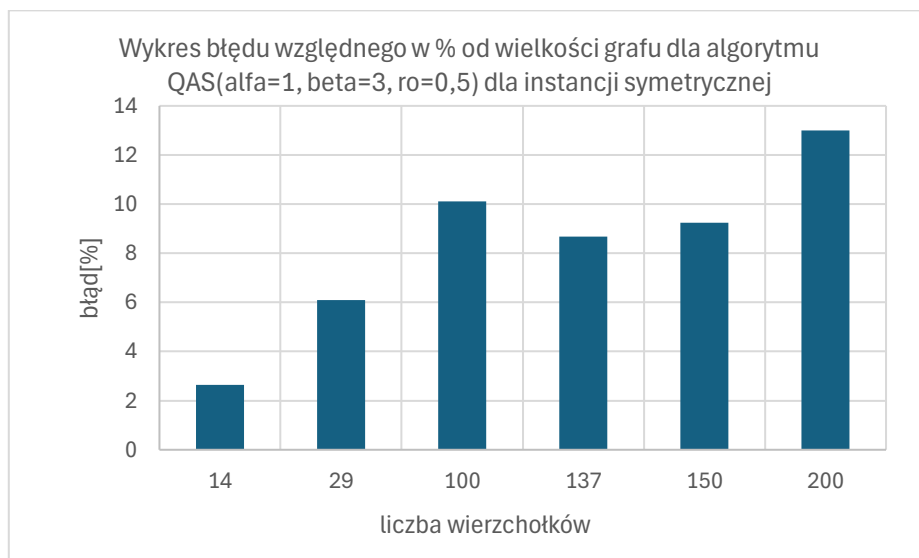
**WYKRES 41 PRZEDSTAWIA BŁĄD W ZALEŻNOŚCI OD WIELKOŚCI GRAFU DLA ALGORYTMU ACOQAS DLA INSTANCJI ASYMETRYCZNEJ**

**TABELA 26 PRZEDSTAWIA CZAS ORAZ BŁĄD DLA ALGORYTMU ACOQAS( $\alpha = 1$ ,  $\beta = 3$ ,  $\rho = 0,5$ ) DLA INST SYMETRYCZNEJ**

V	czas[ms]	błąd[%]
14	4,73	2,65
29	48,78	6,09
100	1678,32	10,12
137	5920,75	8,68
150	6255,6	9,25
200	16071,04	13



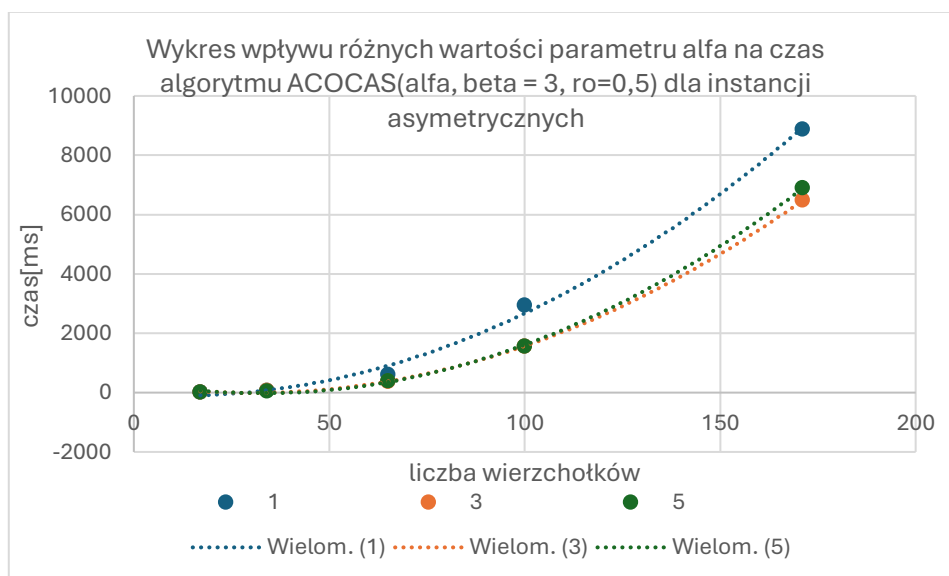
**WYKRES 42 PRZEDSTAWIA CZAS WYKONANIA W ZALEŻNOŚCI OD WIELKOŚCI GRAFU DLA ALGORYTMU ACOQAS DLA INSTANCJI SYMETRYCZNEJ**



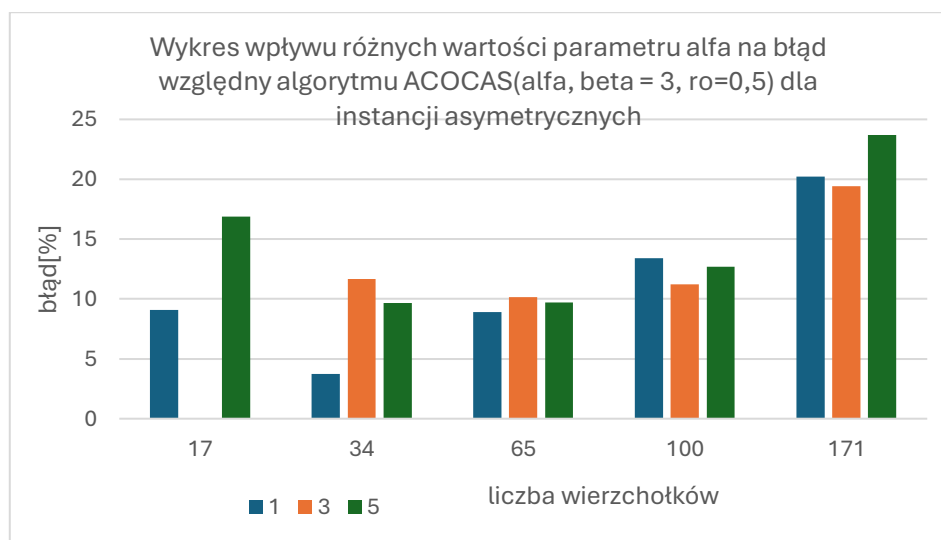
**WYKRES 43 PRZEDSTAWIA BŁĄD W ZALEŻNOŚCI OD WIELKOŚCI GRAFU DLA ALGORYTMU ACOQAS DLA INSTANCJI SYMETRYCZNEJ**

**TABELA 27 PRZEDSTAWIA CZAS I BŁĄD WZGLĘDNY DLA ALGORYTMU ACOCAS( $\alpha$ ,  $\beta$  = 3,  $\rho$  = 0,5) DLA INSTANCJI ASYMETRYCZNEJ (RÓŻNE  $\alpha$  Fy)**

	alfa					
	1	3	5	1	3	5
V	czas[ms]			błąd[%]		
17	12,07	5,74	9,14	9,09	0	16,88
34	54,55	55,92	49,9	3,73	11,66	9,64
65	603,4	371,62	387,7	8,91	10,17	9,7
100	2941,33	1550,64	1554,8	13,42	11,21	12,67
171	8866,77	6471,5	6885,66	20,24	19,43	23,69



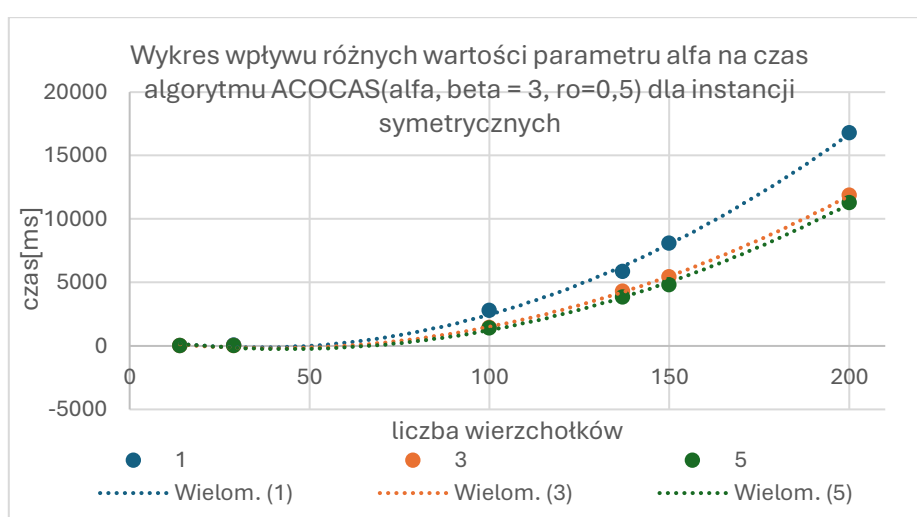
**WYKRES 44 PRZEDSTAWIA CZAS W ZALEŻNOŚCI OD WIELKOŚCI GRAFU DLA ALGORYTMU ACOCAS DLA INSTANCJI ASYMETRYCZNEJ**



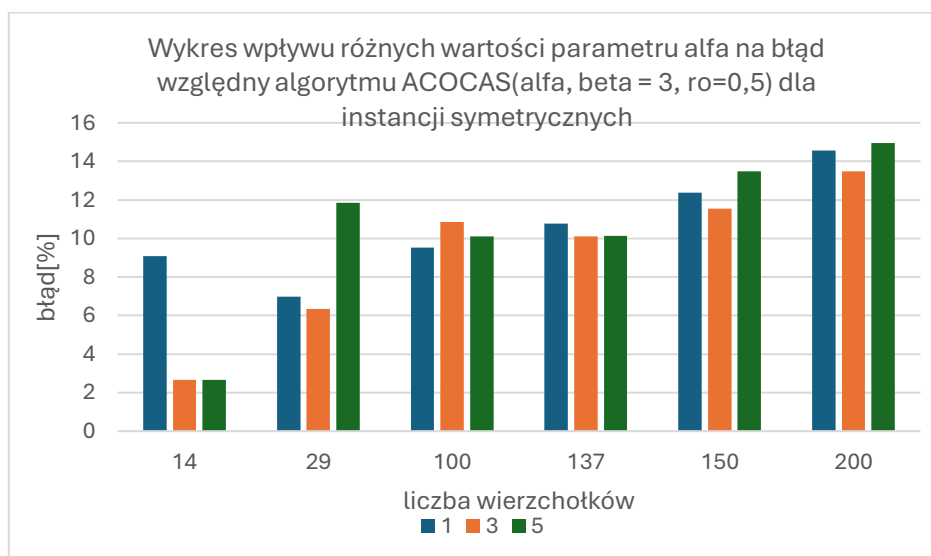
**WYKRES 45 PRZEDSTAWIA BŁĄD W ZALEŻNOŚCI OD WIELKOŚCI GRAFU DLA ALGORYTMU ACOCAS DLA INSTANCJI ASYMETRYCZNEJ**

**TABELA 28 PRZEDSTAWIA CZAS I BŁĄD WZGLĘDNY DLA ALGORYTMU ACOCAS( $\alpha$ ,  $\beta$  = 3,  $\rho$  = 0,5) DLA INST SYMETRYCZNEJ (RÓŻNE ALFY)**

	alfa					
	1	3	5	1	3	5
V	czas[ms]			błąd[%]		
14	2,6	4,76	3,99	9,09	2,65	2,65
29	50,91	33,64	28	6,99	6,34	11,86
100	2771,72	1464,56	1390,85	9,52	10,85	10,11
137	5844,3	4306,26	3820,22	10,78	10,1	10,14
150	8090,62	5431,19	4800,77	12,37	11,56	13,49
200	16769,37	11849,17	11273,6	14,57	13,48	14,95



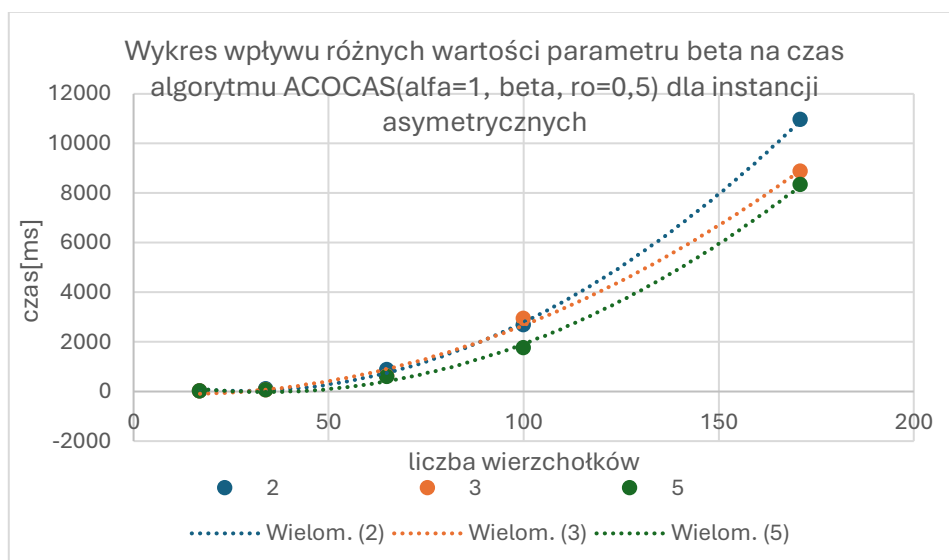
**WYKRES 46 PRZEDSTAWIA CZAS W ZALEŻNOŚCI OD WIELKOŚCI GRAFU DLA ALGORYTMU ACOCAS DLA INSTANCJI SYMETRYCZNEJ**



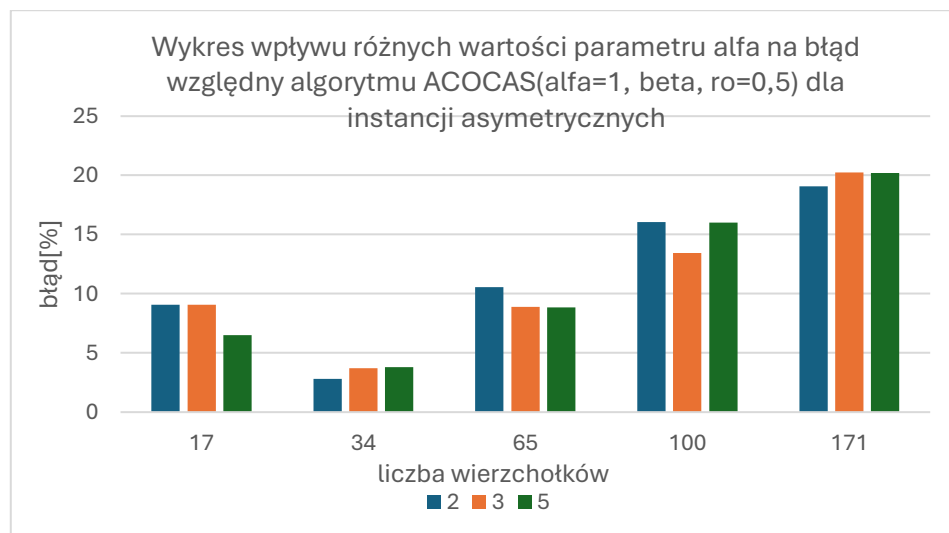
**WYKRES 47 PRZEDSTAWIA BŁĄD W ZALEŻNOŚCI OD WIELKOŚCI GRAFU DLA ALGORYTMU ACOCAS DLA INSTANCJI SYMETRYCZNEJ**

**TABELA 29 PRZEDSTAWIA CZAS I BŁĄD WZGLĘDNY DLA ALGORYTMU ACOCAS( $\alpha=1$ ,  $\beta$ ,  $\rho=0,5$ ) DLA INSTANCJI ASYMETRYCZNEJ (RÓŻNE  $\beta$ Y)**

	beta					
	2	3	5	2	3	5
V	czas[ms]			błąd[%]		
17	13,4	12,07	9,13	9,09	9,09	6,49
34	100,95	54,55	51,83	2,8	3,73	3,79
65	861,4	603,4	595,32	10,57	8,91	8,84
100	2678,67	2941,33	1739	16,04	13,42	15,99
171	10944,67	8866,77	8331,49	19,08	20,24	20,22



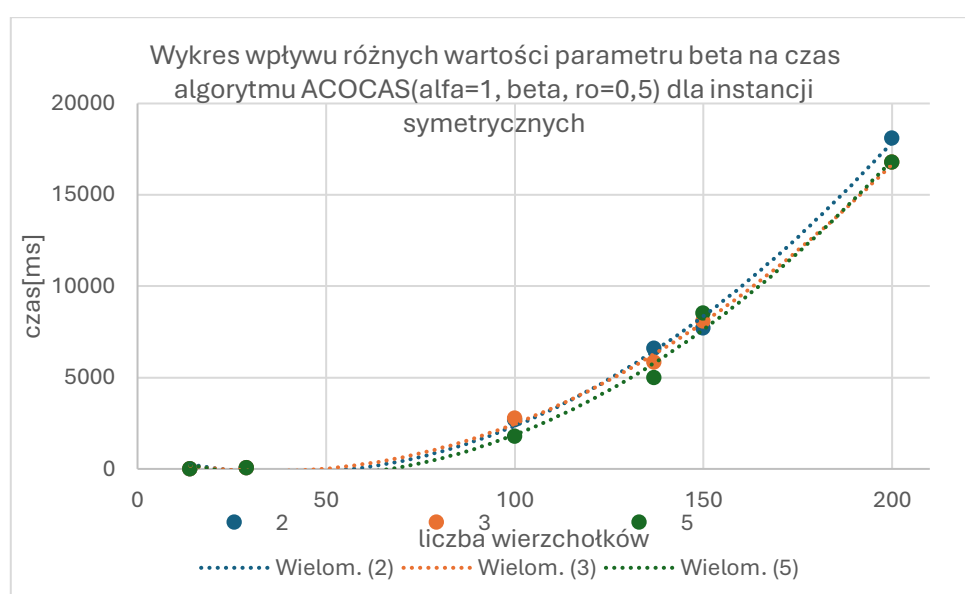
**WYKRES 48 PRZEDSTAWIA CZAS W ZALEŻNOŚCI OD WIELKOŚCI GRAFU DLA ALGORYTMU ACOCAS DLA INSTANCJI ASYMETRYCZNEJ**



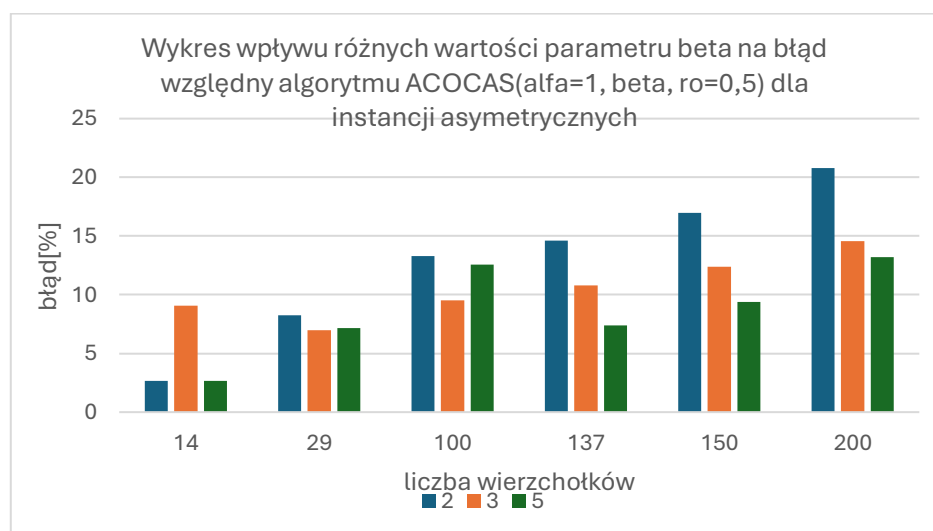
**WYKRES 49 PRZEDSTAWIA BŁĄD W ZALEŻNOŚCI OD WIELKOŚCI GRAFU DLA ALGORYTMU ACOCAS DLA INSTANCJI ASYMETRYCZNEJ**

**TABELA 30 PRZEDSTAWIA CZAS I BŁĄD WZGLĘDNY DLA ALGORYTMU ACOCAS( $\alpha=1$ ,  $\beta$ ,  $\rho=0,5$ ) DLA INST SYMETRYCZNEJ (RÓŻNE  $\beta$ ETY)**

	beta			beta		
	2	3	5	2	3	5
V	czas[ms]			błąd[%]		
14	6,03	2,6	3,45	2,65	9,09	2,65
29	58,26	50,91	43,17	8,27	6,99	7,14
100	2684,21	2771,72	1787,33	13,3	9,52	12,57
137	6597,4	5844,3	4987,69	14,61	10,78	7,38
150	7710,31	8090,62	8534,31	16,97	12,37	9,4
200	18104,81	16769,37	16782,09	20,77	14,57	13,22



**WYKRES 50 PRZEDSTAWIA CZAS W ZALEŻNOŚCI OD WIELKOŚCI GRAFU DLA ALGORYTMU ACOCAS DLA INSTANCJI SYMETRYCZNEJ**



**WYKRES 51 PRZEDSTAWIA BŁĄD W ZALEŻNOŚCI OD WIELKOŚCI GRAFU DLA ALGORYTMU ACOCAS DLA INSTANCJI SYMETRYCZNEJ (RÓŻNE  $\beta$ ETY)**

## PORÓWNANIE ALGORYTMÓW Z ZADANIA 3 I 4:

**TABELA 31 PORÓWNUJĄCA CZASY ALGORYTMÓW: BF, RANDOM, SA, TS, ACOCAS, ACOQAS DLA INSTANCJI  $V \leq 13$  DLA INSTANCJI SYMETRYCZNYCH**

Porównanie czasu alg: random, BF, ACO, TS, SA						
symetryczne	BF	random	TS	SA	ACOCAS	ACOQAS
V	czas[s]					
5	0	0	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	0	0
8	0	0	0	0	0	0
9	0	0	0	0	0	0
10	0	0	0	0	0	0
11	4	4	0	0	0	0
12	41	51	0	0	0	0
13	500	552	0	0	0	0

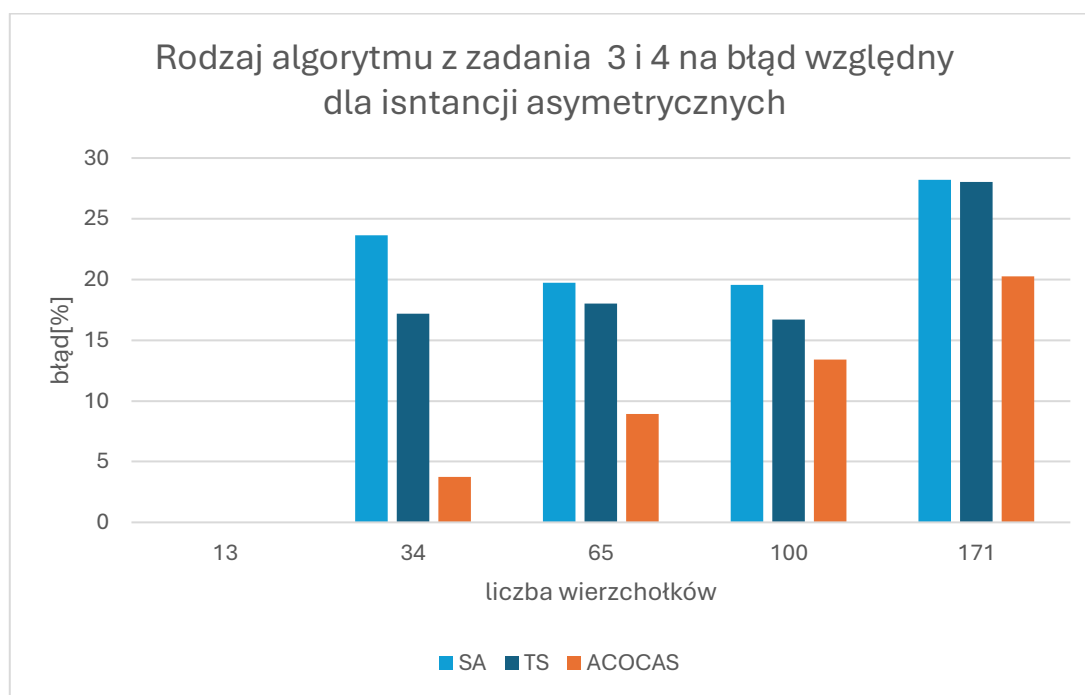
**TABELA 32 PORÓWNUJĄCA BŁĘDY WZGLĘDNE ALGORYTMÓW: BF, RANDOM, SA, TS, ACOCAS, ACOQAS DLA INSTANCJI  $V \leq 13$  DLA INSTANCJI SYMETRYCZNYCH**

Porównanie błędu alg: random, BF, ACO, TS, SA						
symetryczne	BF	random	TS	SA	ACOCAS	ACOQAS
V	błąd względny[%]					
5	0	0	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	0	0
8	0	0	0	0	0	0
9	0	0	0	0	0	10
10	0	0	0	0	0	0
11	0	0	3	3	1	2
12	0	0	0	0	2	0
13	0	0	2	2	1	1

## PORÓWNANIE ALGORYTMU MRÓWKOWEGO DO SA I TS:

TABELA 33 PORÓWNUJĄCA BŁĘDY ALGORYTMÓW: ACOCAS, SA, I TS

	asymetryczne		
V	SA	TS	ACOCAS
	błąd[%]		
13	0	0	0
34	23,64	17,19	3,73
65	19,74	18	8,91
100	19,56	16,72	13,42
171	28,2	28,02	20,24



WYKRES 52 PRZEDSTAWIA BŁĄD W ZALEŻNOŚCI OD WIELKOŚCI GRAFU DLA ALGORYTMÓW ACOCAS, TS I SA.



## Punkt 5 Analiza wyników oraz odniesienie do postawionych hipotez badawczych

1. W algorytmach w zadaniu 3 generowanie początkowego rozwiązania (nn i random) ma znaczący wpływ na błąd rozwiązania uzyskanego przez algorytm.

**Ad.1** Na podstawie tabel nr 12.1 i 12.2 oraz wykresów 19 i 20 można śmiało stwierdzić, że algorytm nn jako początkowe rozwiązanie w alg TS i SA przynosił znacząco lepsze wyniki pod względem błędu niż random. Należy wspomnieć, że im większa instancja, tym ten błąd jest większy (szczególnie dla randoma – jako początkowego rozwiązania).

2. W algorytmie symulowanego wyżarzania dobór odpowiednich parametrów metody ma kluczowy wpływ na jakość i czas otrzymania rozwiązania.

**Ad.2** Na podstawie tabeli nr 11 i wykresu nr 18 można zauważyć minimalny aczkolwiek lepsze wyniki na schematu chłodzenia logarytmicznego w porównaniu z geometrycznym. Czas wykonania dla obydwu z nich są podobne. Wpływ wysokości temperatury początkowej na ma jednoznaczny wpływ na wynik algorytmu – błąd, patrząc chociażby na tabele nr 5 i wykres nr 10 tzn., że można było dobrać większe wartości temperatury początkowej do testów, co mogłoby poskutkować lepszymi rezultatami.

3. W algorytmie przeszukiwania z zakazami dobór odpowiednich parametrów metody ma kluczowy wpływ na jakość i czas otrzymania rozwiązania.

**Ad.3** Pomimo, że na moich wykresach i tabelach nie widać znaczących różnic w zależności od zmienności parametrów tej metody, to wykres nr 29(asym) ukazuje, że słupki z kadencją = 100 są niższe niż te chociażby z kadencją 10 podobna sprawa ma się w wykresie nr 31 gdzie początkowe rozwiązanie jest generowane metodą random, słupki (te gdzie są największe instancje) są niższe dla większej kadencji. Taki stan rzeczy jest widoczny w instancjach symetrycznych aczkolwiek nie aż tak bardzo.

Rozmiar wielkości listy tabu również ma wpływ na jakość rozwiązania świadczy o tym tabela nr 21 i wykres 33(asym) – największy rozmiar listy tabu (100) ma najlepsze wyniki. W grafach symetrycznych - wykres 34 jest podobnie

4. Algorytmy mrówkowe osiągają mniejsze błędy niż algorytmy symulowanego wyżarzania i przeszukiwania z zakazami.

**Ad.4** Tabela nr 33 i wykres 52 pokazują, że algorytm mrówkowy osiąga najlepsze rezultaty ze wszystkich testowanych algorytmów w zadaniu 3 i 4. Czasami różnica procentowa wynosi

5. W algorytmach mrówkowych parametry alfa (odpowiadająca za znaczenie śladu feromonowego) i beta (preferowanie przez mrówki krawędzi w zależności od jej długości) powinny zostać odpowiednio dobrane, tak aby zachować balans i algorytm przynosił jak najlepsze rezultaty.

**Ad.5** Dla instancji asymetrycznych parametr alfa o wartości 3 miał najmniejszy czas wykonania(wykres 44) oraz również zazwyczaj najlepsze koszty wykres 45 (najniższe błędy). Dla instancji symetrycznej było podobnie (wykresy 46 i 47).

Zaś po przetestowaniu parametru beta dla asymetrycznej instancji najlepsza jego wartość to była 3 pod względem błędu(wykr.49 i wyk.48), lecz to nie zawsze, zaś najniższe czasy były dla wartości beta = 5. W przypadku instancji symetrycznej najniższe czasy osiągał parametr beta = 5 (wykres 51), a czasy dla wszystkich parametrów beta były zbliżone (wykres 50).

6. Algorytmy testowane w zadaniu 3 i 4 osiągają lepsze wyniki czasowe i są w stanie otrzymywać stosunkowo małe błędy rozwiązania w stosunku do algorytmów testowanych w zadaniu 1 i 2 (BF, Random, DFS, BFS, Lowest Cost).

**Ad.6** Doskonałym porównaniem okazuje się tabelka nr 31 i 32, które pokazują zbiorcze porównanie BF, random-a, ACO, TS, SA pod względem czasu i błędu. Czasy dla algorytmów z zadania nr 4 są niemalże równe 0 przy praktycznie zerowym błędzie, gdzie dla przykładu BF dla 13 wierzchołków wykonywał się 500s, podobnie random. Wzrost dla algorytmu brute-force jest wykładniczy (ogromny wzrost w ostatnich 3 wierszach ta belki nr 31 dla b-f i randoma)

## Punkt 6 – Bibliografia

1. <http://jaroslaw.mierzwa.staff.iiar.pwr.wroc.pl/>
2. dr inż. Tomasz Kapłon „Projektowanie efektywnych algorytmów” Wykład, Politechnika Wrocławska, semestr zimowy 2024/2025
3. [https://en.wikipedia.org/wiki/Ant\\_colony\\_optimization\\_algorithms](https://en.wikipedia.org/wiki/Ant_colony_optimization_algorithms)
4. <http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/>
5. [https://en.wikipedia.org/wiki/Travelling\\_salesman\\_problem](https://en.wikipedia.org/wiki/Travelling_salesman_problem)