

PEA Projekt 1

Implementacja i analiza efektywności algorytmu Tabu Search oraz Symulowanego Wyżarzania

Mateusz Muzioł, 259223

7 grudnia 2022

Spis treści

1	Wstęp teoretyczny	2
2	Opis algorytmów	2
2.1	Algorytm Przeszukiwań z Zakazami (Tabu Search)	2
2.2	Algorytm Symulowanego Wyżarzania (Simulated Annealing)	3
3	Plan eksperymentu	4
4	Wyniki eksperymentów	5
4.1	Tabu Search	5
4.2	Simulated Annealing	8
5	Wnioski	11

1 Wstęp teoretyczny

Celem drugiego zadania projektowego była implementacja oraz analiza efektywności algorytmów opartych o przeszukiwanie lokalne. Zadanyimi algorytmami było przeszukiwanie z zakazami (Tabu Search) oraz symulowane wyżarzanie (Simulated Annealing). Są to algorytmy heurystyczne, a więc pozwalające znaleźć dobre rozwiązanie problemu przy akceptowalnych nakładach obliczeniowych, bez gwarancji optymalności rozwiązania oraz bez określenia jego jakości, czyli odległości od optimum globalnego. Cechą wspólną obu z implementowanych algorytmów jest to iż stanowią one rozwinięcie przeszukiwania lokalnego. Podczas wykonywania, każdy z algorytmów przeszukuje pewne obszary całej przestrzeni rozwiązań, zwane dalej sąsiedztwami danego rozwiązania. Na początku wybierane jest losowe rozwiązanie, którego sąsiedztwo będzie przeglądane w dalszych krokach algorytmu. Sąsiednim rozwiązaniem nazywane jest rozwiązanie różniące się w określony sposób od rozwiązania początkowego. Zbiór wszystkich możliwych do uzyskania rozwiązań sąsiednich dla danego rozwiązania początkowego jest nazywany sąsiedztwem tego rozwiązania. Badane algorytmy różni sposób przeglądania sąsiedztwa aż do osiągnięcia kryterium stopu.

2 Opis algorytmów

2.1 Algorytm Przeszukiwań z Zakazami (Tabu Search)

Główną ideą przeszukiwania z zakazami jest wybór takiego sąsiada danego rozwiązania startowego, dla którego wartość funkcji oceny ruchu (określenie opłacalności przyjęcia rozwiązania sąsiedniego jako bieżącego) jest jak najwyższa. W przypadku problemu komiwojażera, funkcją tą jest różnica długości ścieżki rozwiązania bieżącego i sąsiedniego. Konsekwencją tego jest konieczność każdorazowego przeszukiwania całego sąsiedztwa danego losowego rozwiązania celem znalezienia najlepszego sąsiada, stąd podejście to zwane jest stromym (Steepy), w związku z czym istnieje możliwość, żeby w kolejnej iteracji algorytmu mógł zostać wykonany ten sam ruch oferujący poprawienie rozwiązania, jednak tylko w obszarze lokalnym, prowadząc do „utknięcia” algorytmu w tym obszarze i ograniczenia eksploracji przestrzeni wszystkich rozwiązań (zmniejszenia dywersyfikacji), a co za tym idzie, obszarów, w których może znajdować się optimum globalne. Efekt ten może zostać złagodzony dzięki zastosowaniu Listy Tabu, czyli listy wykonanych ruchów, zabronionych w określonej ilości iteracji (kadencji) przeglądu sąsiedztwa. Możliwym jest, żeby ruch znajdujący się na liście tabu prowadził do rozwiązania lepszego niż najlepsze dotychczasowe (jest ono zapamiętywane). Spełnia ono wtedy tzw. kryterium Aspiracji i może zostać wykorzystane pomimo tego, że znajduje się na liście tabu.

2.2 Algorytm Symulowanego Wyżarzania (Simulated Annealing)

Drugim zaimplementowanym algorytmem było symulowane wyżarzanie (Simulated Annealing). Podobnie jak w przypadku Tabu Search, jego działanie polega na badaniu różnych obszarów przestrzeni rozwiązań, jednak zastosowana została w nim inna strategia wyboru rozwiązania sąsiedniego. W odróżnieniu od Tabu Search, zamiast wyboru najlepszego rozwiązania znajdującego się w sąsiedztwie, jako rozwiązanie bieżące w danym pokoleniu przyjmowany jest każdy ruch prowadzący do poprawy rozwiązania bieżącego, stąd podejście to nazywane jest zachłannym (Greedy). Ponadto, możliwym jest przyjęcie rozwiązania gorszego, ponieważ w jego sąsiedztwie może znajdować się rozwiązanie o wiele lepsze, a być może globalnie optymalne. Jest to realizowane dzięki zastosowaniu mechanizmu przypominającego proces wyżarzania metalu.

Działanie algorytmu symulowanego wyżarzania polega na stopniowym ograniczaniu przyjmowania gorszych rozwiązań podczas badania sąsiedztwa danego rozwiązania początkowego. Stanowi to analogię do zmniejszającej się wraz z temperaturą ruchliwości cząsteczek wyżarzanego metalu. Podczas drugiego przeglądania sąsiedztwa gorsze rozwiązanie przyjmowane jest z pewnym prawdopodobieństwem.

Parametrem algorytmu, który ma wpływ na to prawdopodobieństwo jest parametr przeniesiony bezpośrednio z podstaw termodynamicznych algorytmu, czyli temperatura - im wyższa, tym prawdopodobieństwo wyboru gorszego rozwiązania jest większe (większa dywersyfikacja). Im niższa, tym algorytm jest bardziej zbliżony w działaniu do typowych metod iteracyjnych (większa intensyfikacja). Na początku działania algorytmu temperatura jest wysoka, dzięki czemu algorytm może bardzo często zmieniać konfigurację rozwiązania, niejednokrotnie wybierając rozwiązanie gorsze. Wraz z kolejnymi iteracjami algorytmu temperatura spada i coraz częściej wybierane są rozwiązania lepsze. Pod koniec pracy algorytmu temperatura jest na tyle niska, że prawdopodobieństwo wyboru gorszego rozwiązania jest bliskie zeru. Algorytm zachowuje się wówczas jak typowy algorytm iteracyjny i stara się maksymalnie ulepszyć rozwiązanie.

3 Plan eksperymentu

Efektywność zaimplementowanych algorytmów została zbadana poprzez obliczenie błędów względnych oraz czasów wykonania dla trzech instancji asymetrycznego problemu komiwojażera. Dla każdej z nich kryterium stopu, określone jako dopuszczalny czas wykonywania algorytmu, był inny i dla pliku `ftv47.atsp` wynosił 2 minuty, `ftv170.atsp` 4 minuty, a dla `rbg403.atsp` 6 minut. Dla obydwu algorytmów określony został taki sam rodzaj sąsiedztwa powstającego poprzez zamianę miejscami dwóch wierzchołków w rozwiązaniu początkowym.

W przypadku algorytmu przeszukiwania z zakazami (Tabu Search) błąd względny i czas znalezienia optymalnego rozwiązania sprawdzane były w zależności od kryterium stopu, czyli dopuszczalnego czasu wykonywania algorytmu, wynoszącego kolejno 1, 2, 3, 4, 5 oraz 6 minut.

W przypadku algorytmu symulowanego wyżarzania (Simulated Annealing) błąd względny i czas znalezienia optymalnego rozwiązania sprawdzane były w zależności od współczynnika obniżania temperatury wynoszącego kolejno 0.8, 0.85, 0.9, 0.95, 0.99, 0.999, 0.9999 oraz 0.99999. W tym wypadku kryterium stopu zostało ustawione jako stałe i inne, najbardziej optymalne dla każdej instancji problemu. W implementacji algorytmu należało również określić funkcję obliczającą temperaturę początkową w zależności od danych wejściowych. Działa ona poprzez 10 000-krotne spróbkowanie przestrzeni rozwiązań, czyli wygenerowanie losowego rozwiązania oraz jego sąsiada i obliczenie różnicy w jakości tychże rozwiązań. Na podstawie próbek określana jest średnia wartość różnicy, a temperatura początkowa obliczana jest ze wzoru:

$$T = \frac{-\delta}{\log(dT)}$$

gdzie:

δ - obliczona średnia różnica w jakości uzyskanych próbek

dT - współczynnik obniżania temperatury

4 Wyniki eksperymentów

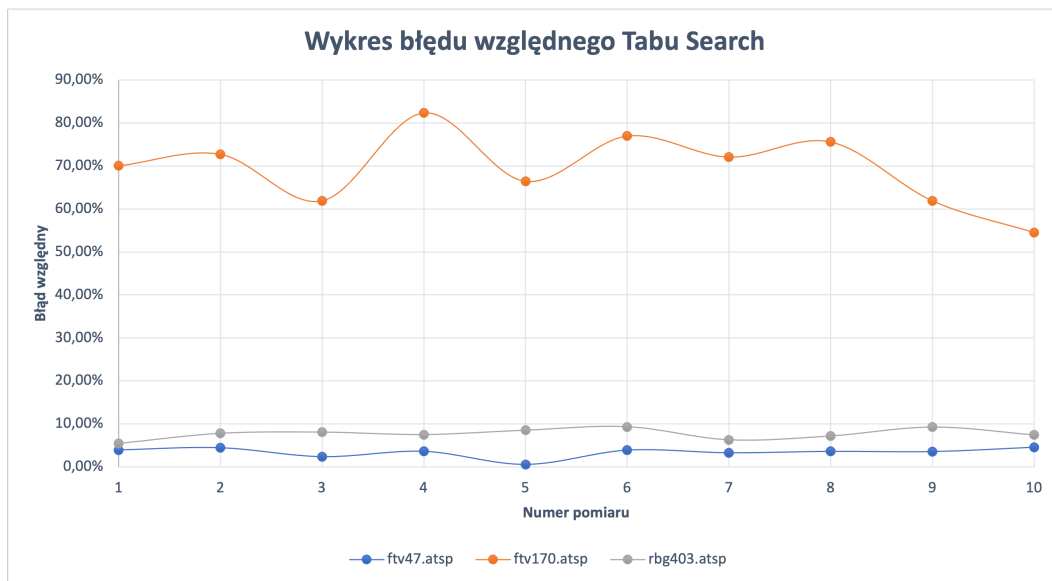
4.1 Tabu Search

L.p.	Plik								
	ftv47.atsp (1776)			ftv170.atsp (2755)			rbg403.atsp (2465)		
	Rozwiązanie [długość ścieżki]	Błąd [%]	Czas [s]	Rozwiązanie [długość ścieżki]	Błąd [%]	Czas [s]	Rozwiązanie [długość ścieżki]	Błąd [%]	Czas [s]
1	1846	3,94%	93,6709	4685	70,05%	227,988	2599	5,44%	346,29
2	1855	4,45%	119,829	4759	72,74%	160,098	2657	7,79%	342,108
3	1818	2,36%	38,872	4461	61,92%	177,814	2664	8,07%	327,506
4	1840	3,60%	47,1734	5024	82,36%	232,89	2650	7,51%	318,256
5	1786	0,56%	65,3281	4586	66,46%	205,501	2675	8,52%	315,01
6	1845	3,89%	101,697	4876	76,99%	238,258	2695	9,33%	274,333
7	1834	3,27%	69,3421	4741	72,09%	233,832	2620	6,29%	327,198
8	1840	3,60%	108,746	4838	75,61%	238,472	2642	7,18%	259,387
9	1839	3,55%	81,5228	4461	61,92%	232,042	2693	9,25%	359,686
10	1857	4,56%	49,3761	4258	54,56%	187,434	2648	7,42%	336,232
Avg.	1836	3,38%	77,5557	4668	69,47%	213,433	2654	7,68%	320,601

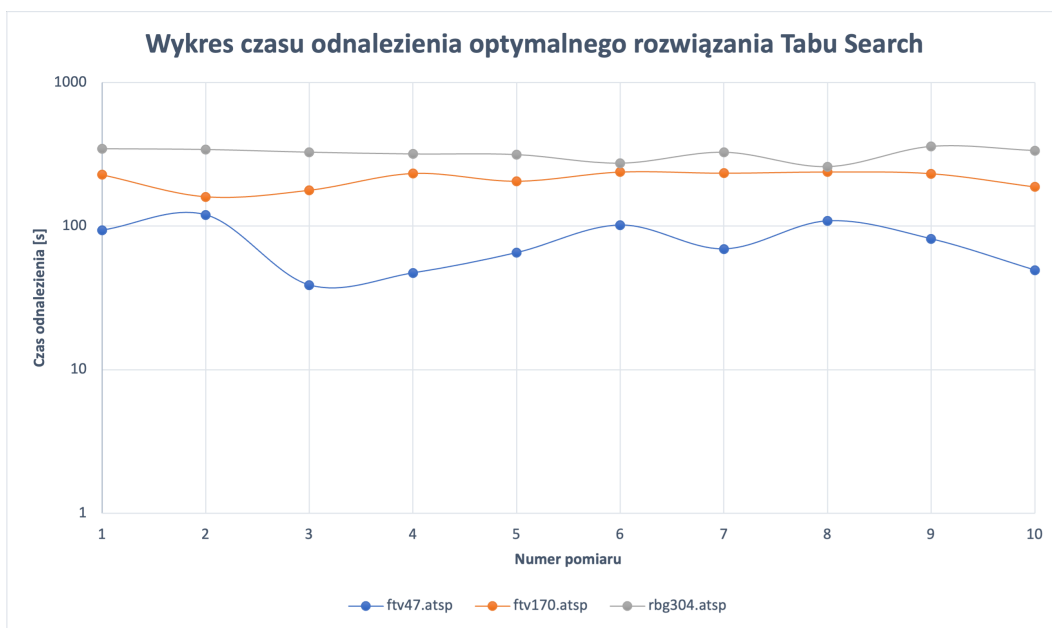
Rysunek 1: Zestawienie błędów względnych i czasów odnalezienia optymalnego rozwiązania dla każdego pomiaru

Czas [s]	Plik								
	ftv47.atsp (1776)			ftv170.atsp (2755)			rbg403.atsp (2465)		
	Rozwiązanie [długość ścieżki]	Błąd [%]	Czas [s]	Rozwiązanie [długość ścieżki]	Błąd [%]	Czas [s]	Rozwiązanie [długość ścieżki]	Błąd [%]	Czas [s]
60	1922	8,22%	24,2222	6196	124,90%	52,8564	2910	18,05%	59,8905
120	1876	5,63%	13,4023	4745	72,23%	114,866	2679	8,68%	112,813
180	1828	2,93%	92,1445	5575	102,36%	171,069	2670	8,32%	127,05
240	1852	4,28%	31,7854	4440	61,16%	223,263	2735	10,95%	200,476
300	1894	6,64%	262,739	4564	65,66%	291,363	2622	6,37%	281,824
360	1857	4,56%	172,587	4845	75,86%	324,296	2616	6,13%	359,047

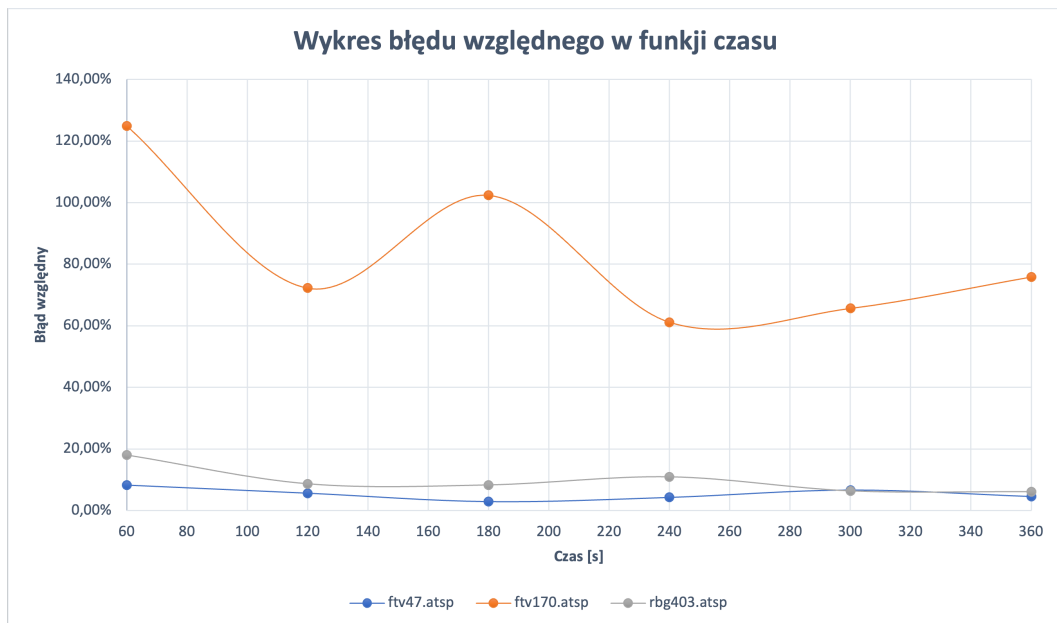
Rysunek 2: Zestawienie błędów względnych i czasu względem kryterium stopu



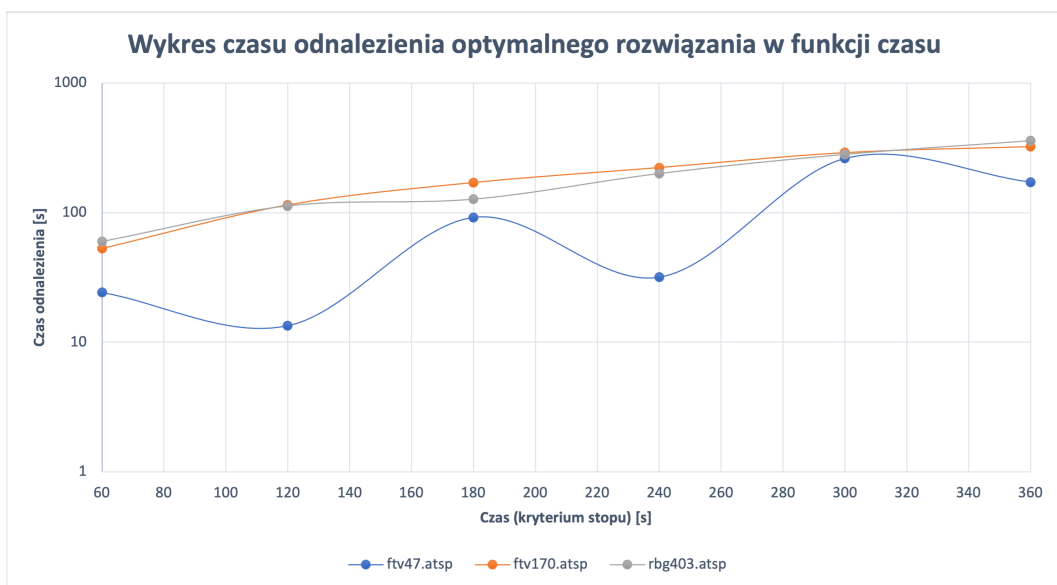
Rysunek 3: Zestawienie błędów względnych w formie wykresu



Rysunek 4: Zestawienie czasów odnalezienia optymalnego rozwiązania w formie wykresu



Rysunek 5: Wykres błędów względnych w zależności od kryterium stopu



Rysunek 6: Wykres czasu odnalezienia w zależności od kryterium stopu

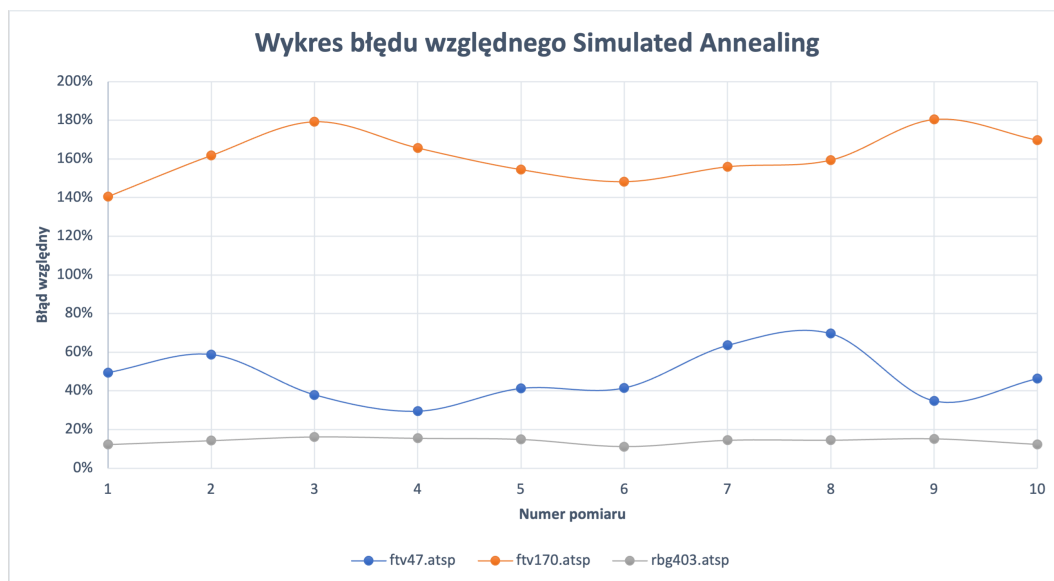
4.2 Simulated Annealing

L.p.	Plik											
	ftv47.atstp (1776)				ftv170.atstp (2755)				rbg403.atstp (2465)			
	Rozwiązanie [długość ścieżki]	Błąd [%]	Czas [s]	Tk	Rozwiązanie [długość ścieżki]	Błąd [%]	Czas [s]	Tk	Rozwiązanie [długość ścieżki]	Błąd [%]	Czas [s]	Tk
1	2655	49%	0,005421	0,426898	6625	140%	0,224345	2820,39	2766	12%	1,24476	260,031
2	2821	59%	0,007258	0,439965	7212	162%	0,155937	1354,18	2816	14%	1,15656	123,603
3	2450	38%	0,008538	1,09005	7694	179%	0,202295	4661,75	2864	16%	1,24631	64,3157
4	2299	29%	0,008389	19,8662	7319	166%	0,299664	4090,78	2846	15%	1,05562	8,36106
5	2509	41%	0,002396	10,5471	7010	154%	0,200118	1741	2832	15%	0,986583	9,15261
6	2513	41%	0,008255	230,742	6838	148%	0,224677	811,097	2740	11%	1,06493	7,26364
7	2905	64%	0,004559	97,39	7051	156%	0,221223	2377,43	2821	14%	1,04831	5,10959
8	3013	70%	0,00547	2,27027	7146	159%	0,172498	1639,11	2822	14%	1,10874	10,4301
9	2394	35%	0,00741	38,9544	7726	180%	0,35928	60,0604	2840	15%	1,23212	198,233
10	2598	46%	0,006872	1,72597	7431	170%	0,168464	1983,99	2768	12%	1,19619	158,909
Avg.	2615	47%	0,006457	40,34529	7205	162%	0,22285	2153,979	2811	14%	1,134012	84,54087

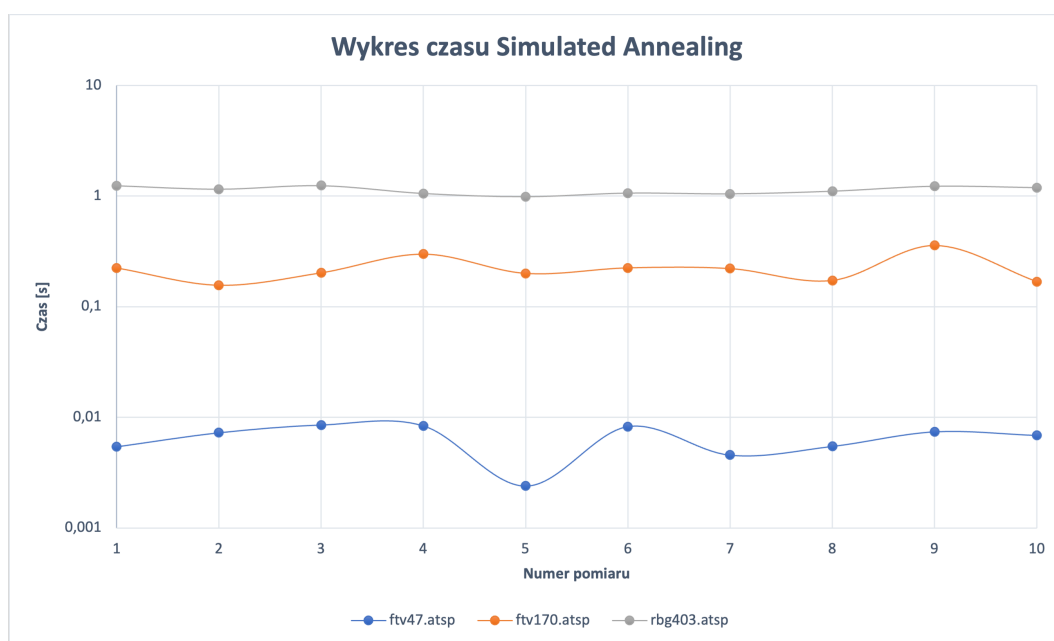
Rysunek 7: Zestawienie błędów względnych, czasów odnalezienia optymalnego rozwiązania oraz temperatur końcowych dla każdego pomiaru

Współczynnik temperatury	Plik											
	ftv47.atstp (1776)				ftv170.atstp (2755)				rbg403.atstp (2465)			
	Rozwiązanie [długość ścieżki]	Błąd [%]	Czas [s]	Tk	Rozwiązanie [długość ścieżki]	Błąd [%]	Czas [s]	Tk	Rozwiązanie [długość ścieżki]	Błąd [%]	Czas [s]	Tk
0,80	3426	93%	0,000829	590,763	19746	617%	0,000477	0,853217	6904	180%	0,000836	49,0114
0,85	2991	68%	0,001358	0,176817	18295	564%	0,000723	7,10331	7145	190%	0,000831	28,1839
0,90	2857	61%	0,002611	1627,84	16868	512%	0,001227	2601,72	6837	177%	0,001436	599,636
0,95	2690	51%	0,00322	0,158953	10520	282%	0,00927	673,652	5433	120%	0,0042	1392,99
0,99	2582	45%	0,003887	3,15102	7493	172%	0,211756	49,3381	2803	14%	1,15478	15,1281
0,999	2747	55%	0,005941	2,75521	7142	159%	0,164954	1,49768	2796	13%	2,27879	0,531671
0,9999	2731	54%	0,004369	4474,66	6610	140%	0,296807	0,140825	2791	13%	2,76526	0,81549
0,99999	2546	43%	0,004414	0,969905	7395	168%	0,17719	1267,28	2754	12%	2,08758	660,913
Avg.	2821	59%	0,003329	837,5594	11758	327%	0,107801	575,1981	4682	90%	1,036714	343,4012

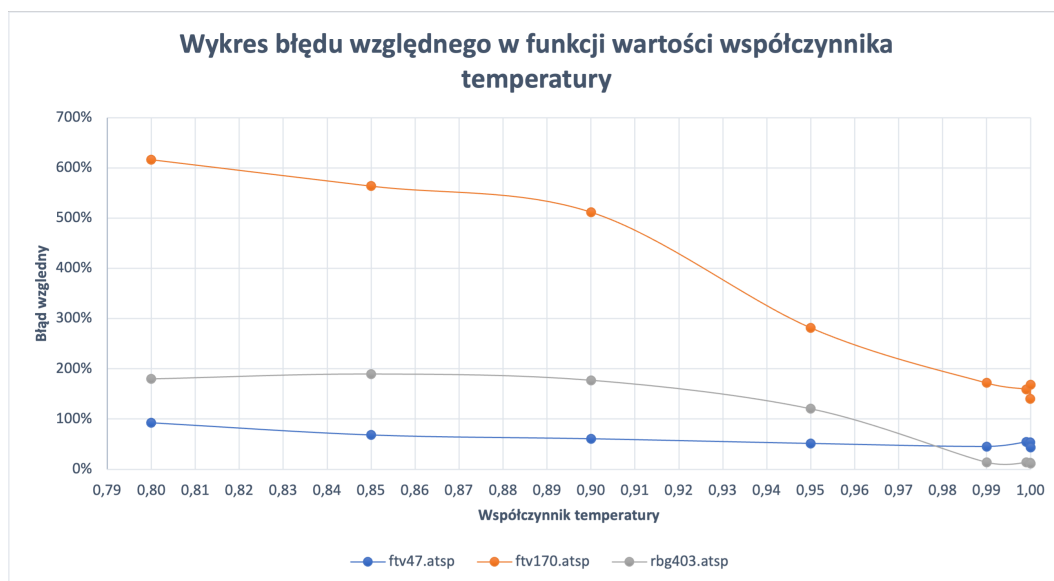
Rysunek 8: Zestawienie błędów względnych, czasu odnalezienia oraz temperatury końcowej w zależności od współczynnika temperatury



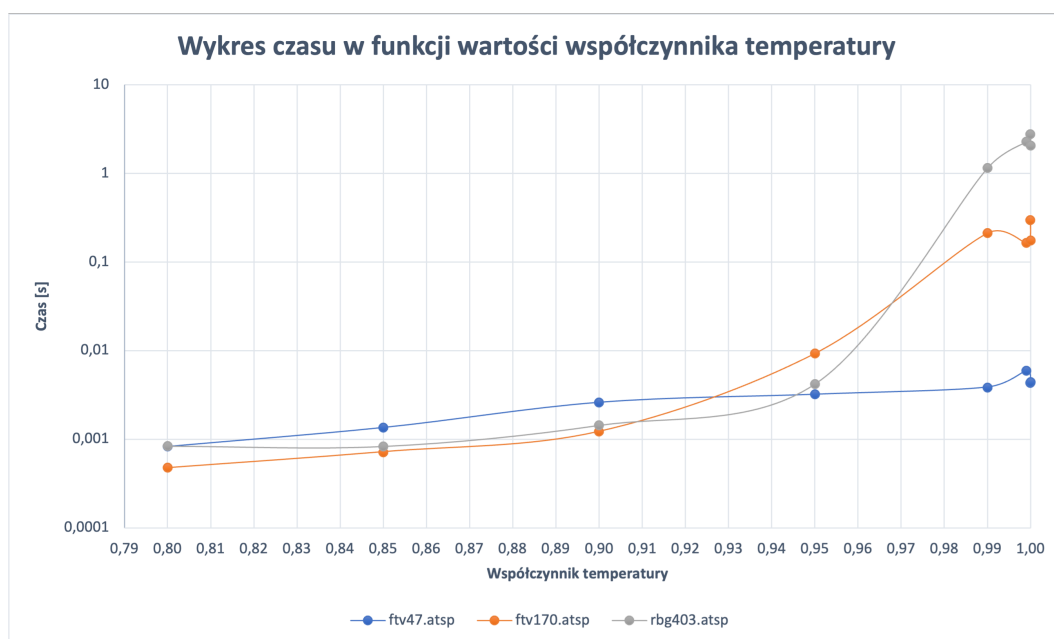
Rysunek 9: Zestawienie błędów względnych w formie wykresu



Rysunek 10: Zestawienie czasów odnalezienia optymalnego rozwiązania w formie wykresu



Rysunek 11: Wykres błędów względnych w zależności od współczynnika temperatury



Rysunek 12: Wykres czasów odnalezienia w zależności od kryterium stopu

5 Wnioski

Głównymi czynnikami wpływającymi na jakość wygenerowanych rozwiązań jest czas, przez który badane algorytmy mogą działać (kryterium stopu) oraz dodatkowo, w przypadku algorytmu symulowanego wyżarzania, współczynnik zmiany temperatury. Algorytm symulowanego wyżarzania w porównaniu do algorytmu przeszukiwania z zakazami wykazał się wielokrotnie niższym czasem znalezienia najbardziej optymalnego rozwiązania, lecz zostało to okupione o wiele większym błędem względnym. W przypadku dużej instancji problemu komiwojażera zawartej w pliku `ftv170.atsp`, oba algorytmy cechował wysoki i nieregularny błąd względny generowanych rozwiązań (ponad 60%). Badając efektywność algorytmu symulowanego wyżarzania w zależności od współczynnika zmiany temperatury wyraźnie widać, że im jest on większy (ale nie większy od 1.0), tym błąd względny generowanych rozwiązań jest niższy, a czas ich znalezienia rośnie. Może być to spowodowane tym, że przy większym współczynniku temperatura zmniejsza swoją wartość szybciej, czyli istnieje coraz większe prawdopodobieństwo znalezienia optymalnego rozwiązania, lecz przeszukanie przestrzeni rozwiązań trwa dłużej.