

Sprawozdanie 8: algorytm branch and bound

Celem ostatniego ćwiczenia była implementacja algorytmu przeszukującego graf branch and bound, wraz z jego rozszerzoną wersją, która sprawdza czy sprawdzany element był już odwiedzony. Celem zapewnienia wyników które można porównać obie wersje algorytmu pracowały każdorazowo na tym samym grafie. Celem zapewnienia spójności grafu macierz sąsiedztwa była macierzą jednostkową uzupełnioną o dodatkowe, losowe połączenia. Poniższa tabela przedstawia wyniki pomiarów poszczególnych algorytmów.

Ilość danych	B & B		B & B extended	
	Średni czas [ms]	Ilość odwiedzonych elementów	Średni czas [ms]	Ilość odwiedzonych elementów
10	0.0088	6	0.0059	6
100	0.7904	213	0.7516	205
1000	4183.41	204080	225.411	10021

Wnioski

Szybkość działania algorytmu poza rozmiarem grafu w dużej mierze zależy od ilości zawartej w nim krawędzi. Ilość ta wpływa też w dużej mierze na różnicę między podstawową a rozszerzoną wersją algorytmu. Zgodnie z oczekiwaniami wersja rozszerzona każdorazowo sprawdziła się lepiej niż podstawowa. Złożoność obliczeniowa zadanego problemu jest nie zawsze jednakowa ze względu na pewną losowość: pesymistyczna wersja wiązałaby się z rozpatrzeniem wszystkich możliwości, natomiast optymistyczna, gdzie od razu znajdziemy najlepsze połączenie prowadziłaby w zasadzie do liniowej długości. Oczywiście są to przypadki skrajne, więc zazwyczaj będziemy mieć do czynienia z przypadkiem pośrednim.