

Data oddania: _____

Ocena: _____

Dariusz Jędrzejczak 173095 173095@edu.p.lodz.pl

Zadanie 1: Piętnastka

1. Wstęp

Ten dokument zawiera tylko najistotniejsze dane i wykresy. Dane testowe, pośrednie oraz skrypty użyte do generowania i obróbki danych są dostępne na życzenie.

Pomoc do programu użytego do generowania i rozwiązywania układanek jest dostępna po uruchomieniu go bez parametrów.

2. Opis rozwiązania

Do testów użyto 1140 różnych układanek (z czego pierwsze 860 to wszystkie możliwe stany planszy na głębokościach 0-8, a pozostałe 280 stanowią stany na głębokościach 9-22, po 20 układanek dla każdej głębokości).

Wszystkie układanki poddano próbie rozwiązania przez 6 różnych algorytmów przeszukiwania grafów (DFS, zmodyfikowany algortm DFS, ID-DFS, BFS, A*, zmodyfikowany algortm A*). Modyfikacja DFS polegała na użyciu heurystyki odległości Hamminga do wyboru następnego stanu. Modyfikacja A* polegała na braniu pod uwagę tylko heurystyki przy nadawaniu priorytetów kolejnym stanom. Zmodyfikowany algorytm DFS (na wykresach DFS hamming) okazał się skuteczniejszy. A* z modyfikacją (na wykresach A* hack) był zdecydowanie najszybszy ze wszystkich analizowanych algorytmów. Rozwiązał wszystkie układanki, nie zawsze znajdując jednak optymalną ścieżkę. Co ciekawe, w pewnych zakresach głębokości (0-5 i 9-15) działał niemal identycznie jak A*, znajdując zawsze optymalne ścieżki (Rysunek 3).

Każdy z algorytmów został przetestowany używając co najmniej czterech różnych ustalonych kolejności przeszukiwania sąsiednich stanów. Dodatkowo

każdy algorytm został uruchomiony co najmniej 4 razy z losową kolejnością przeszukiwania sąsiadów w każdym węźle grafu stanów. Z danych wynika, że w ogólności ta kolejność nie ma znaczenia dla wyników, zatem dane ze wszystkich testów z różnymi kolejnościami zostały połączone na wykresach. W każdym z testów mierzono liczbę ruchów użytych do rozwiązania, liczbę wygenerowanych i odwiedzonych stanów, czas obliczeń w sekundach oraz zajętość pamięci fizycznej w MB.

Każdy algorytm przerywał rozwiązywanie układanki po osiągnięciu określonej ilości iteracji. Dla DFS było to 1500000 iteracji, dla pozostałych algorytmów 500000. Nierozwiązanie 20 układanek z rzędu powodowało zakończenie dalszego przetwarzania dla danego wariantu algorytmu.

Dla każdego algorytmu na podstawie danych wyznaczono, dopasowując je metodą najmniejszych kwadratów do funkcji a^x , współczynnik rozgałęziania a . Wszystkie współczynniki i powiązane funkcje zostały przedstawione na wykresach. Niektóre z tych współczynników są nieco zaniżone, gdyż algorytmy, które nie rozwiązały wszystkich przypadków testowych wygenerowały dane z przynajmniej dwudziestoma punktami, w których rozwiązanie nie zoszło osiągnięte i wszystkie mierzone wartości osiągnęły maksimum (jest to widoczne na wykresach). Wyrzucenie tych danych również zaburzyłoby wyniki. W obu tych przypadkach różnica jest nieznacząca dla porównywania wartości współczynników między sobą.

Algorytm A* był testowany z użyciem 3 heurystyk: odległości Hamminga[1] (na wykresach A* hamming), odległości Manhattan[2] (A* manhattan) oraz heurystyki Gaschniga (N-MaxSwap)[3] (A* nmax). Wszystkie te heurystyki są optymistyczne, zatem gwarantują znalezienie optymalnej ścieżki, co potwierdzają zebrane dane. Tylko heurystyka oparta na odległości Hamminga poradziła sobie ze wszystkimi przypadkami testowymi. Pozostałe heurystyki zaczęły przekraczać wyznaczony limit iteracji od głębokości 18, rozwiązując kilka przypadkowych układanek na tej głębokości i kończąc działanie na głębokości 19 (po osiągnięciu limitu 20 nieudanych prób z rzędu).

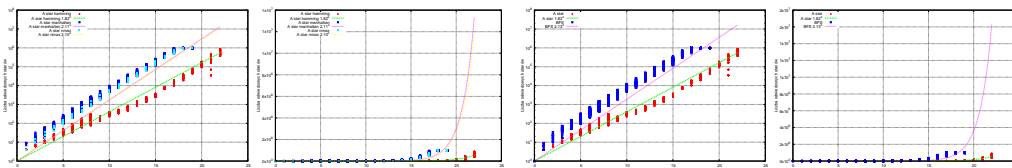
Wygenerowanie wszystkich danych zajęło ok. 22 godziny obliczeń.

3. Wyniki

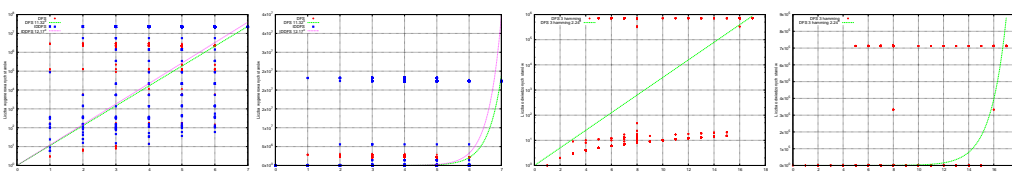
Algorytmy uporządkowane w kolejności od najmniejszego współczynnika rozgałęziania a , obliczonego na podstawie liczby odwiedzonych stanów (dla IDDFS wygenerowanych):

Nazwa algorytmu	współczynnik rozgałęziania
A* hack hamming	1.57
A* hamming	1.82
A* nmax	2.10
A* manhattan	2.11
BFS	2.13
DFS hamming	2.24
DFS	11.32
IDDFS	12.17

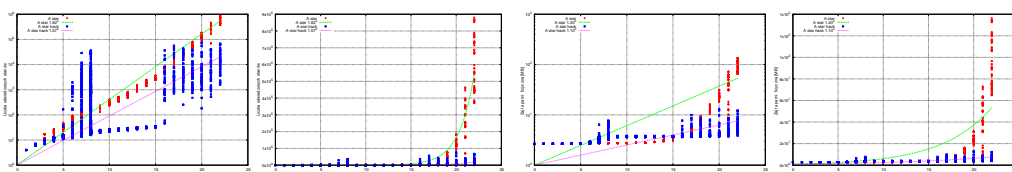
4. Wykresy



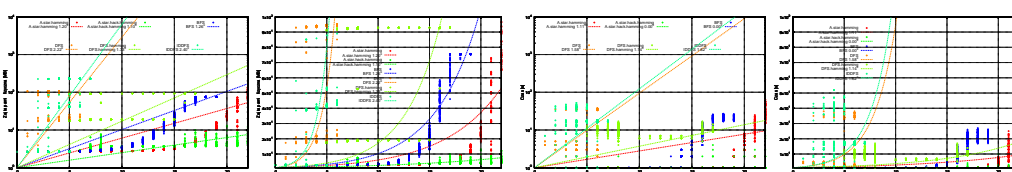
Rysunek 1. Porównanie heurystyk A* i porównanie A* (Hamming) z BFS



Rysunek 2. Porównanie DFS i IDDFS oraz DFS z heurystyką; pierwsze 2 wykresy przedstawiają liczbę wygenerowanych stanów (dla IDDFS odwiedzone stany nie były liczone)



Rysunek 3. Porównanie A* i jego zmodyfikowanej wersji (obie używają odległości Hamminga)



Rysunek 4. Porównanie wszystkich algorytmów; dla algorytmów BFS i A* hack współczynnik a wynosi 0 na wykresie czasu; wynika to z błędnego dopasowania;

Literatura

- [1] <http://heuristicswiki.wikispaces.com/Hamming+distance+%28Heuristic%29>
- [2] <http://heuristicswiki.wikispaces.com/Manhattan+Distance>
- [3] <http://heuristicswiki.wikispaces.com/N-MaxSwap>