

AiSD — zadanie czwarte

Gliwiński Jarosław Marek
Kruczyński Konrad Marek
Grupa dziekańska I5

5 czerwca 2008

1 Wstęp

Zadanie czwarte polega na wyszukiwaniu cykli Hamiltona i Eulera w danym grafie. Pierwszy z wymienionych to cykl przechodzący przez wszystkie wierzchołki dokładnie jeden raz, natomiast drugi przechodzi rzeczony jeden raz przez wszystkie krawędzie.

Jako strukturę, w której przechowujemy graf, zdecydowaliśmy się wybrać listę następników, ponieważ pozwala ona na łatwe przechodzenie do kolejnych wierzchołków połączonych krawędzią z danym. Ponadto jest to łatwa w implementacji i intuicyjna w przypadku tego typu problemu struktura.

2 Pomiary

2.1 Generacja grafów

Oczywiście podstawowym zadaniem, które stało przed programem jeszcze przed właściwą jego częścią była generacja grafów, które zawierają cykl Eulera i Hamiltona oraz spełniają dodatkowy warunek podany w zadaniu, tj. cechują się czterdziesto lub siedemdziesięcioprocentowym wypełnieniem. W przypadku cyklu Eulera sprawę ułatwia poniższe twierdzenie:

Twierdzenie 2.1 *Dany graf V zawiera cykl Eulera wtedy i tylko wtedy, gdy jest on spójny i każdy jego wierzchołek ma stopień parzysty. Jeżeli graf ten jest skierowany, liczba krawędzi wchodzących i wychodzących w każdym wierzchołku musi być równa.*

Wykorzystamy je w poniższej metodzie generacji grafu. Weźmy zbiór wierzchołków pewnego grafu (na razie bez krawędzi) i połączmy je „pętlą”, tj. z ustawiamy je w jakimś porządku, następnie łączymy dany z następnym, a ostatni w tym porządku z pierwszym. Wówczas otrzymujemy graf, który zawiera cykl Hamiltona (wspomniana pętla) oraz Eulera (każdy wierzchołek ma stopień parzysty). Należy teraz tak dokładać krawędzie, aby uzyskać żądane wypełnienie i jednocześnie zachować parzysty stopień każdego wierzchołka. W tym celu wybierzmy dowolny wierzchołek z grafu i połączmy go

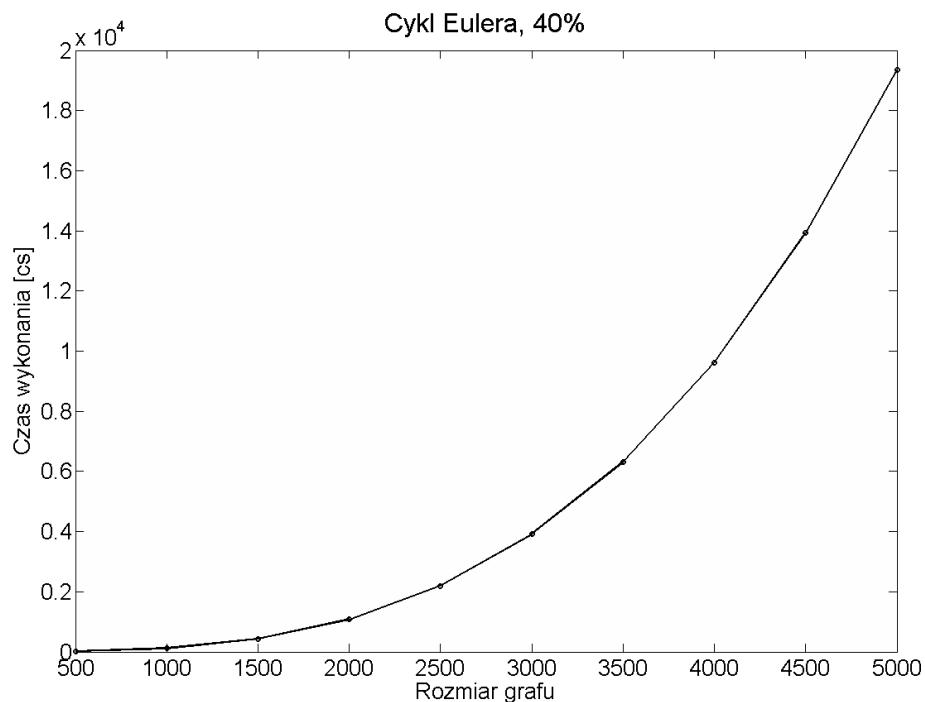
z innym losowo wybranym. Ten z kolei następnym i tak dalej, a gdy będziemy w pobliżu wymaganego wypełnienia, łączymy ostatnio wybrany z pierwszym. Jeżeli połączenie takie już istnieje, szukamy wierzchołka nie połączonego jeszcze z początkowym ani końcowym i realizujemy połączenie pośrednio przez niego. W bardzo pesymistycznym przypadku może się zdarzyć, że znalezienie wierzchołka pośredniczącego nie będzie możliwe. W czasie naszych testów nie spotkaliśmy się jednak z takim przypadkiem. Otrzymany ostatecznie graf zawiera oba cykle.

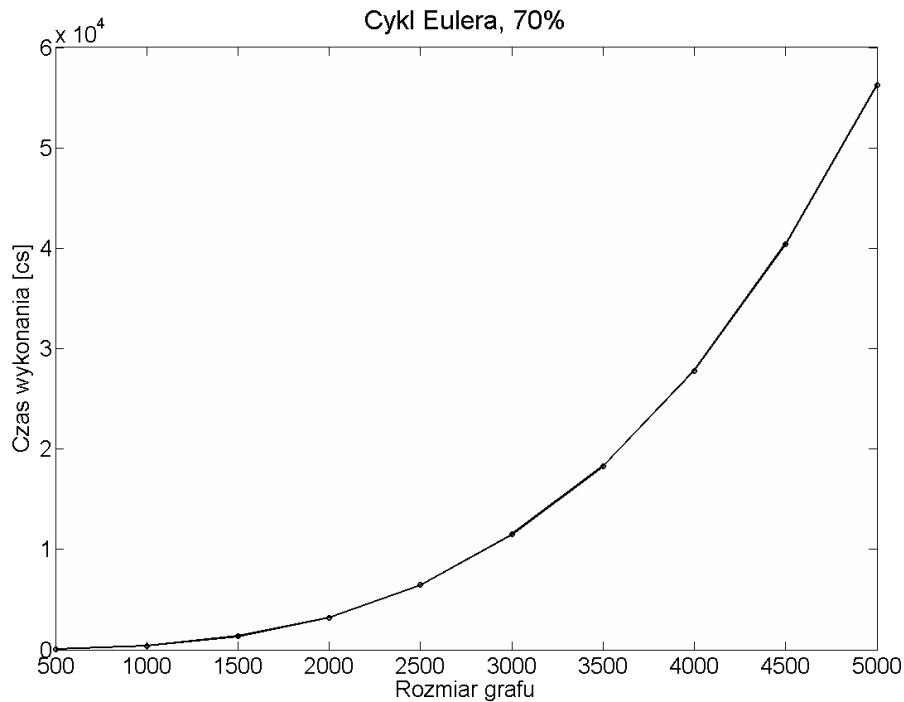
2.2 Wyszukiwanie cyklu Eulera

Algorytm stosowany w tym przypadku jest lekko zmodyfikowanym algorytmem DFS. Zaczynamy od dowolnie wybranego wierzchołka. Wiemy, iż graf jest spójny i zawiera żądany cykl. Krawędź, którą dostaniemy się do danego wierzchołka, usuwamy. Każdy odwiedzony wierzchołek odkładamy na stos. Gdy nie będziemy mogli już ruszyć się z miejsca (znajdziemy się w wierzchołku izolowanym), kończymy wykonywanie procedury DFS. Następnie zdejmujemy ze stosu wierzchołki – kolejne tworzą cykl.

Złożoność powyższego algorytmu wynosi $O(m) \approx O(n^2)$, problem znalezienia cyklu Eulera należy zatem do klasy problemów wielomianowych.

Poniższe wykresy (i tabela) zawierają dane na temat wydajności algorytmu przy różnych wypełnieniach:





Odchylenia wyników w próbie przedstawiają się:

Dane	Odchylenie względne [%]
Wypełnienie 40%	$0,45 \pm 0,0008$
Wypełnienie 70%	$0,83 \pm 0,14$

2.3 Wyszukiwanie cyklu Hamiltona

W tym przypadku zastosowano tzw. algorytm z powracaniem. Rozpoczynamy klasycznie od dowolnie wybranego wierzchołka. Przechodzimy następnie do dowolnego połączonego z nim wierzchołka i czynimy tak, aż napotkamy na cykl. Wówczas sprawdzamy jego długość. Jeżeli wynosi ona n , odwiedziliśmy wszystkie wierzchołki i praca skończona. W przeciwnym wypadku powracamy do ostatniego wierzchołka, który ma jeszcze nieodwiedzone krawędzie i szukamy kolejnego cyklu. Jeżeli w pewnym momencie powrót nie będzie możliwy, oznacza to, iż graf nie zawiera cyklu Hamiltona. Z uwagi na wybór krawędzi złożoność takiego algorytmu (pesymistyczna) wynosi aż $O(n!)$. W ogólności problem poszukiwania cyklu Hamiltona należy do klasy problemów NP-zupełnych.

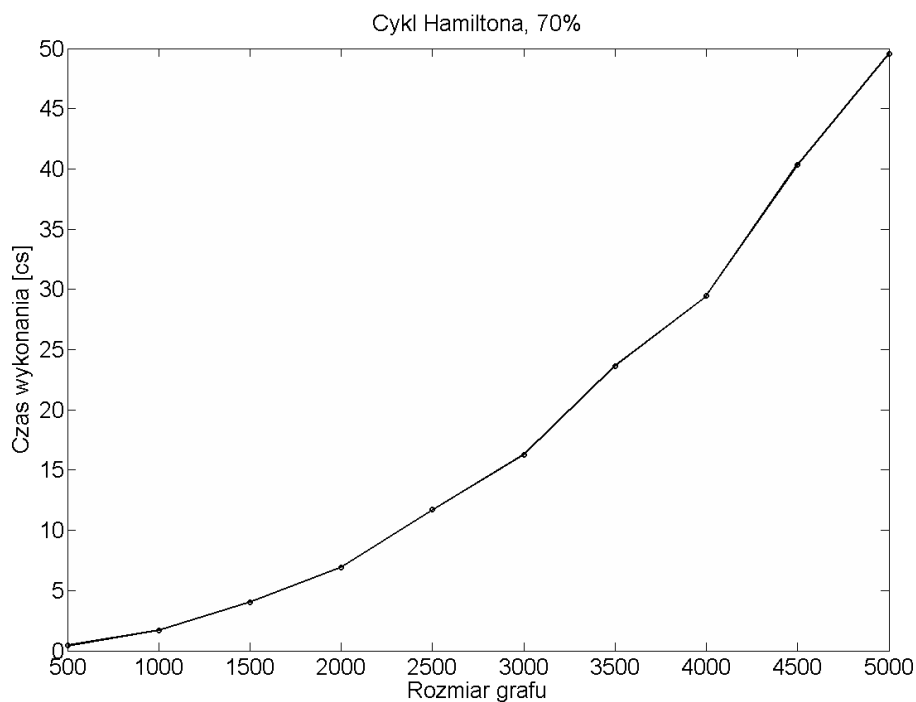
Poniżej znajdują się wyniki dla przeprowadzonych praktycznie poszukiwań. Okazuje się, że pomimo wykładniczej złożoności pesymistycznej bardzo często zdarzają się przypadki szybkiego wyszukiwania. Dla wypełnienia 40% jedynie raz na kilka poszukiwań złożoność wykraczała poza liniową, natomiast w przypadku 70% – nie udało się zaobserwować takiego przypadku ani razu! Podstawowym wnioskiem płynącym z tych obserwacji jest fakt, iż nie sposób miarodajnie określić średniej złożoności algorytmu w funkcji rozmiaru instancji, ponieważ w poszczególnych instancjach tego samego rozmiaru rozbieżność

wyników potrafi sięgać wielu rzędów wielkości. Dla przykładu, fragment zestawu prób przeprowadzonych dla grafów o rozmiarze 3500 wierzchołków:

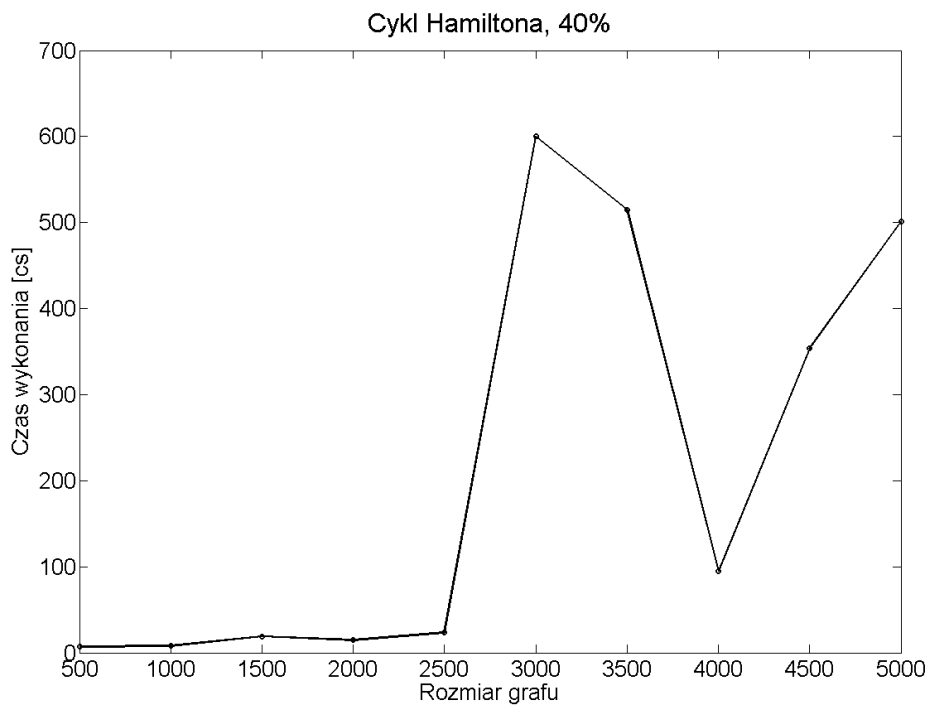
139.621
17.7591
1254.32
6671.63
21.6171
915.076
15.7578
15.8424

Wspomniane różnice są wyraźne. Należy ponadto uczciwie odnotować, że kilka razy podczas procedury testowej (w sumie 200 zaplanowanych pomiarów) trzeba było przerywać przetwarzanie ze względu na przedłużające się przetwarzanie jednego grafu, po czasie rzędu 10 minut.

Jak wspomniano, dla grafów o dużym wypełnieniu właściwie w całości występowała złożoność bliska optymistycznej. Można zaobserwować bliski liniowemu wykres odpowiadający samym niemalże instancjom o optymistycznej złożoności:



Nie można tego powiedzieć o gorzej wypełnionych grafach. Już przy badanych 40% ocena złożoności średniej na podstawie wykresu uśredniającego 20 prób na punkt pomiarowy okazuje się niemożliwa:



Odchylenia wyników w próbie przedstawiają się:

Dane	Odchylenie względne [%]
Wypełnienie 40%	$3,15 \pm 0,062$
Wypełnienie 70%	279 ± 21
Wypełnienie 70 -3 najwyższe%	77 ± 16

Widać wyraźnie, jak ogromny wpływ mają te skrajne wartości na kształt krzywej będącej wynikiem pomiaru, gdy wykreślmy alternatywną krzywą, przy konstrukcji której pominięto 3 najdłużej trwające próby w każdym punkcie:

