Prezentacja kierunków pracy naukowej

Dariusz Dereniowski

Katedra Algorytmów i Modelowania Systemów
Politechnika Gdańska

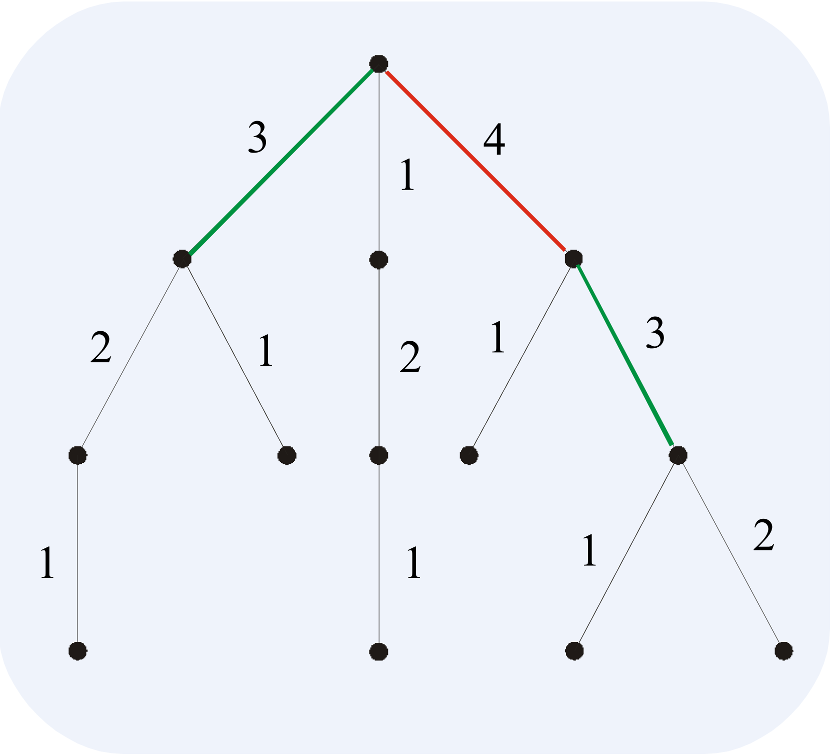
Kierunki badawcze

- Uporządkowane kolorowanie grafów
- Szeregowanie zadań w środowisku wieloprocesorowym
- Wyszukiwanie elementów w częściowych porządkach
- Przeszukiwanie grafów
- Złożoność obliczeniowa w teorii gier

Kolorowanie uporządkowane

Def. Funkcję $c: E \to \{1,\ldots,k\}$ nazywamy *uporządkowanym k-pokolorowaniem* krawędzi grafu G jeśli każda ścieżka łącząca dwie krawędzie x,y takie, że $c(x) = c(y)$ zawiera krawędź z spełniającą $c(z) > c(x)$.

Def. Najmniejsza liczba k, dla której istnieje uporządkowane k□pokolorowanie krawędzi G nazywamy *uporządkowanym indeksem chromatycznym* grafu G, oznaczanym symbolem $\chi_r'(G)$.



Zastosowanie kolorowania

Równoległe procesy montażu:

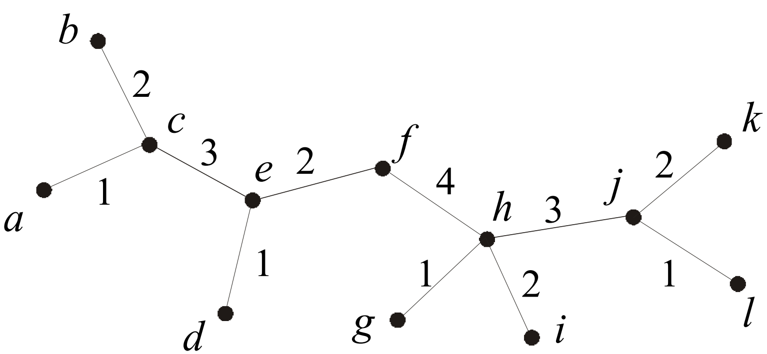
- wierzchołek grafu: reprezentuje element składowy
- krawędź grafu: występuje wówczas, gdy elementy odpowiadające wierzchołkom na jej końcach powinny zostać połączone (scalanie) podczas procesu montażu
- ograniczenie: nie można przeprowadzać jednocześnie dwóch operacji scalania jeśli współdzielą one element składowy
- cel: minimalizacja czasu (liczby równoległych tur) całego montażu

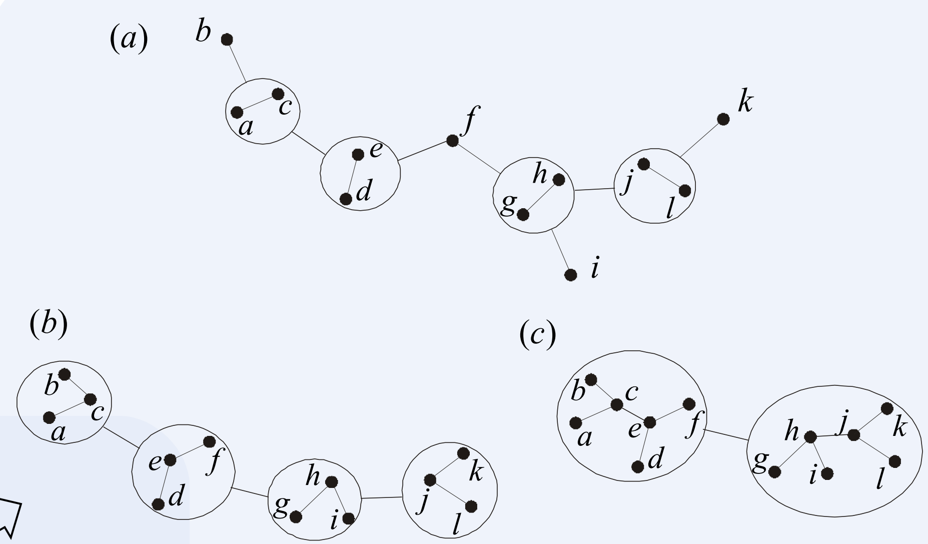
Zastosowanie kolorowania

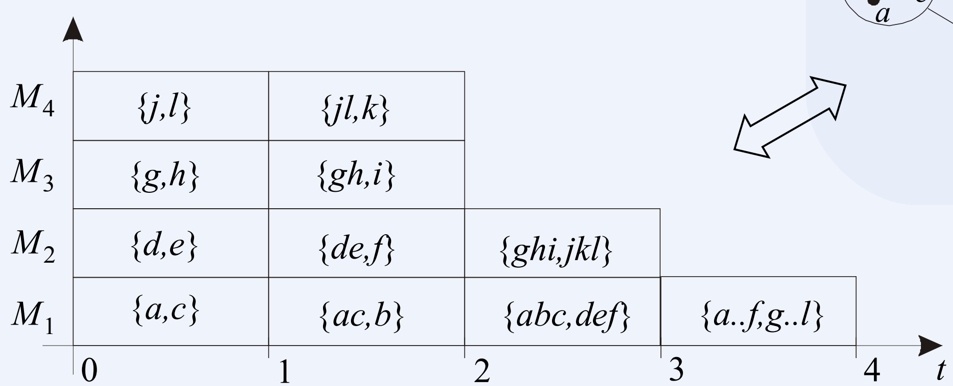
Wykorzystanie uporządkowanego pokolorowania do konstrukcji harmonogramu:

- kolor: reprezentuje numer przedziału czasowego, w którym operacja odpowiadająca danej krawędzi zostanie zrealizowana
- wymagana liczba maszyn: jest równa krotności koloru, który został użyty największą liczbę razy
- długość harmonogramu: jest równa liczbie wykorzystanych kolorów (uporządkowany indeks chromatyczny grafu)

Przykład procesu złączania







Uogólnienia problemu

Przyjęte kierunki badań nad modelem:

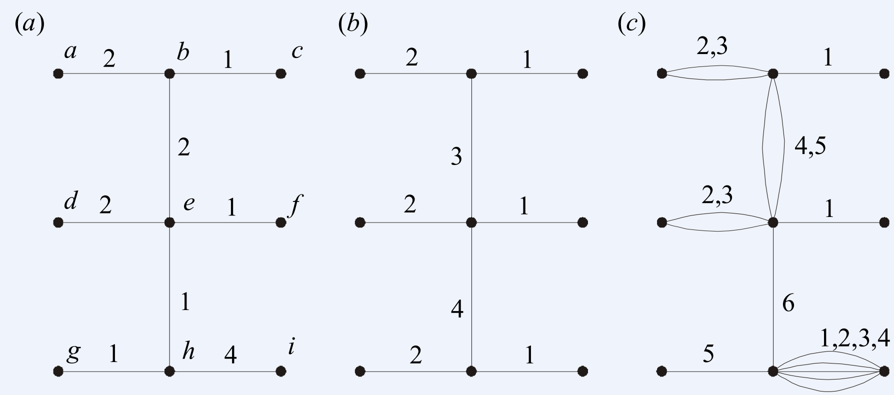
- grafy ważone: wagi krawędzi modelują różne czasy realizacji poszczególnych operacji złączania
- listy dozwolonych/zabronionych przedziałów czasowych dla poszczególnych operacji
- ograniczenie liczby dostępnych maszyn/procesorów

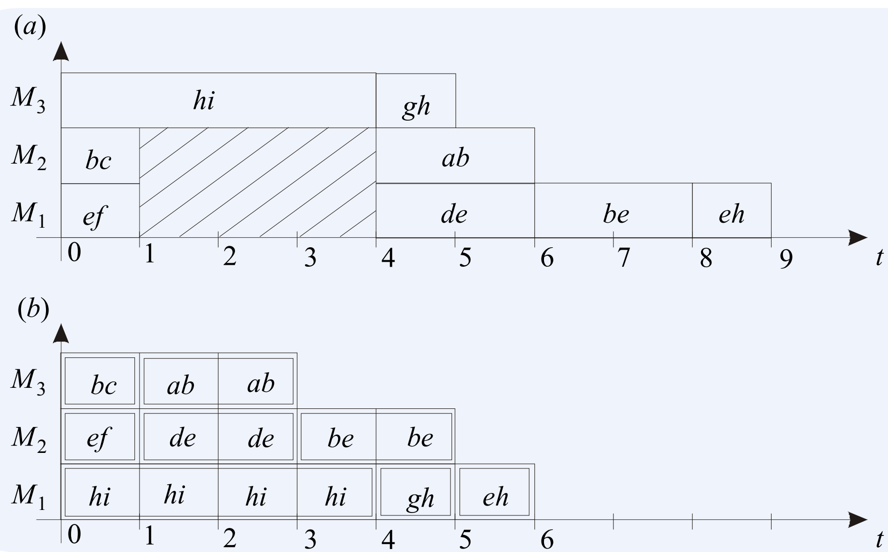
Wnioski i wyniki:

- problemy ogólniejsze stają się obliczeniowo trudne już dla szczególnych acyklicznych klas grafów
- stanowi to motywację do konstrukcji wielomianowych algorytmów przybliżonych

Przykład – grafy ważone

(a) graf ważony
(b) optymalne uporządkowane pokolorowanie odpowiedniego grafu prostego
(c) optymalne uporządkowane pokolorowanie odpowiedniego multigrafu (uwzględniamy różne czasy operacji)



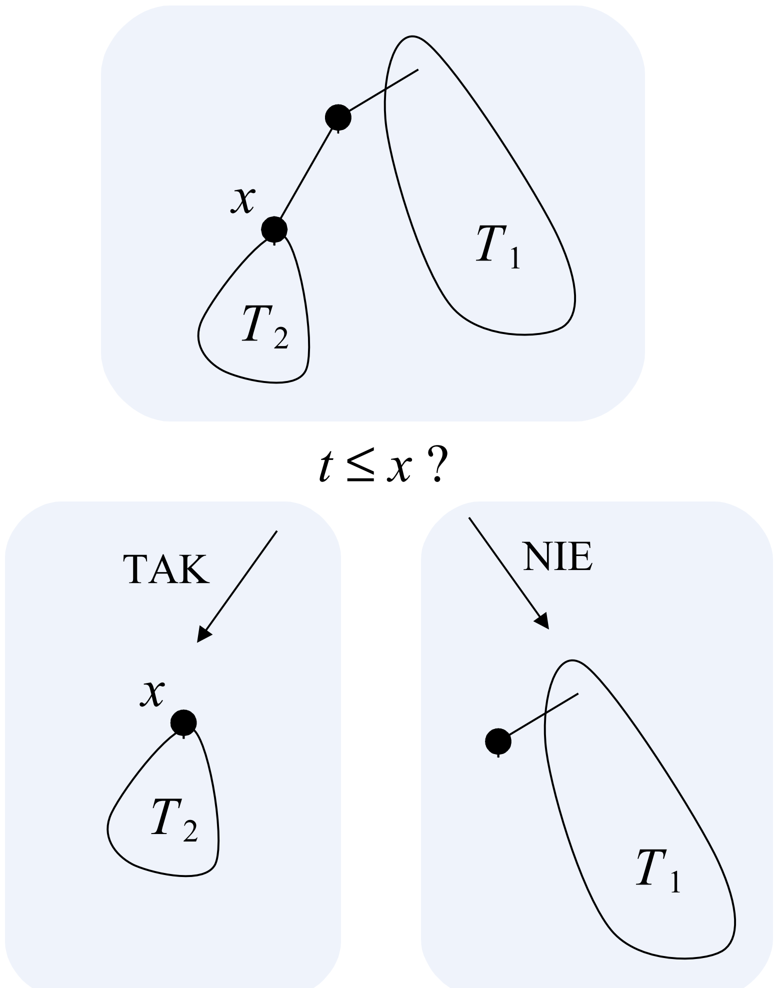


- uszeregowanie odpowiadające pokolorowaniu z rys. (*b*)
- uszeregowanie odpowiadające pokolorowaniu z rys. (*c*)

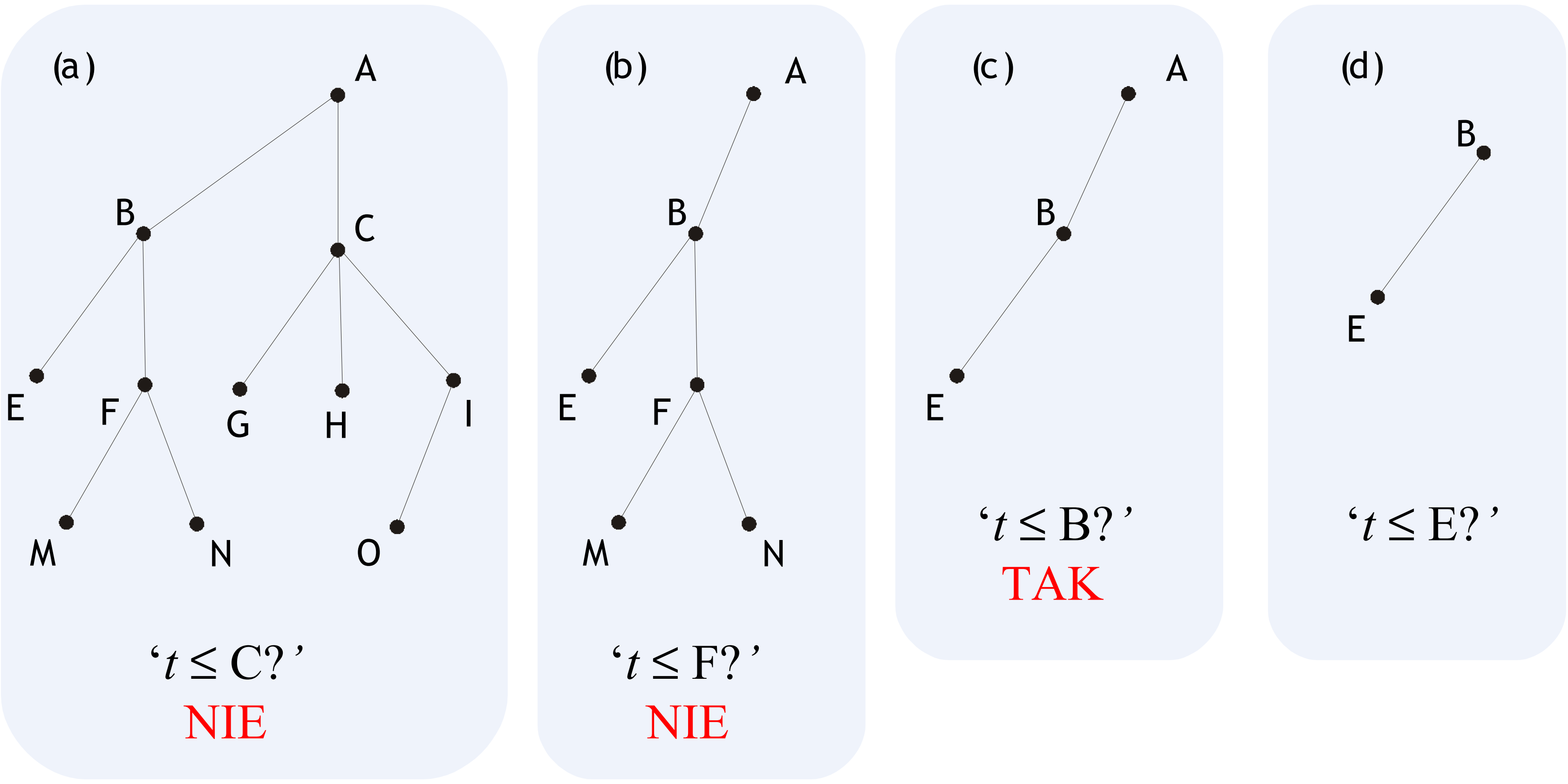
Wniosek: Korzystniejszy harmonogram dzięki uwzględnieniu czasów operacji w grafie ważonym.

Wyszukiwanie w częściowych porządkach

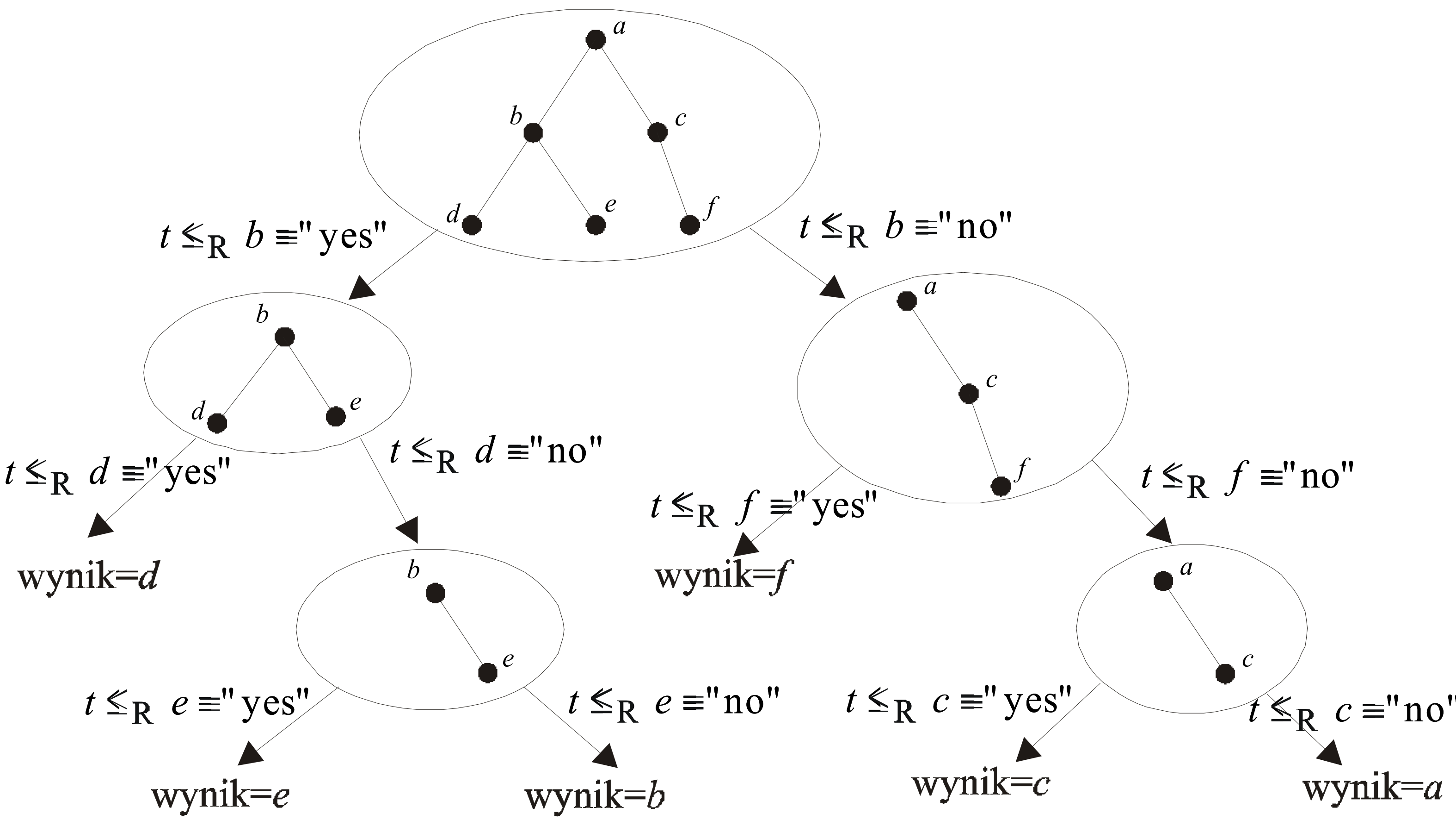
- Zakładamy, że dany jest diagram Hassego relacji (graf skierowany)
- W każdym kroku wyszukiwania wybieramy dowolny wierzchołek grafu x i wykonujemy test postaci $t \leq x$.
- Ograniczamy przestrzeń przeszukiwania w zależności od wyniku testu: jeśli odpowiedź brzmi „TAK", to kontynuujemy wyszukiwanie w poddrzewie zakorzenionym w x; w przeciwnym wypadku przeszukujemy komplementarną część grafu
- Celem jest obliczenie takiej strategii, która minimalizuje liczbę testów w najgorszym przypadku



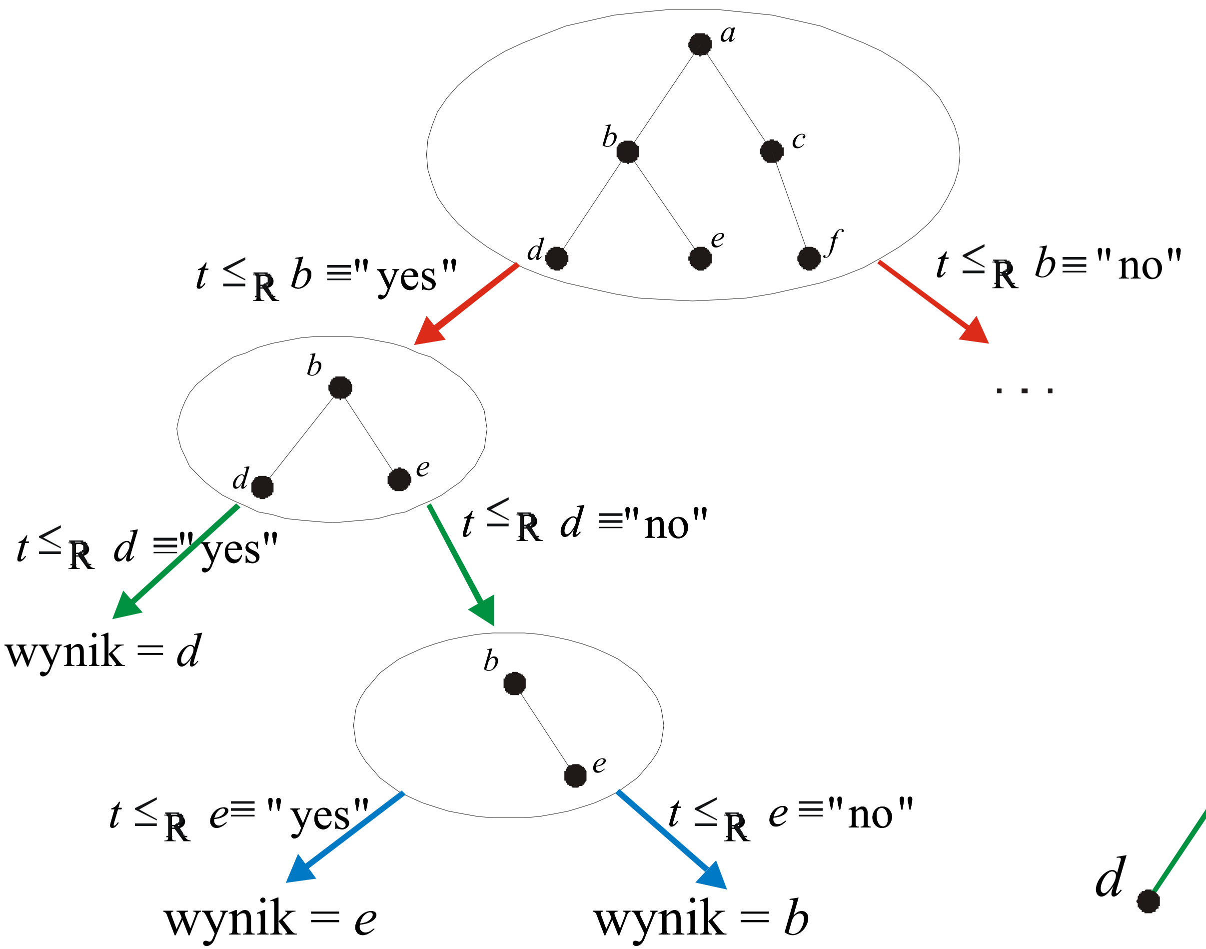
Redukcja przestrzeni

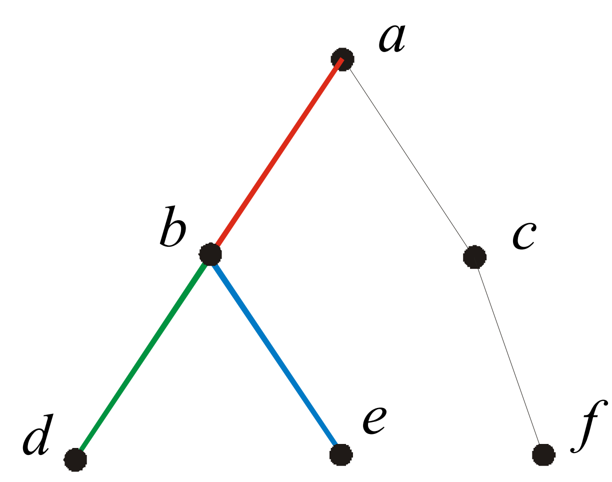


Przykład kompletnej strategii



Związek pomiędzy problemami





Uzyskane wyniki

Tw. $\chi_r'(T)$ *jest równe maksymalnej liczbie zapytań w optymalnej strategii wyszukiwania, gdzie* T *jest drzewem odpowiadającym diagramowi Hassego.*

Tw. *Jeśli diagram Hassego jest zakorzenionym drzewem, to optymalną strategię wyszukiwania można znaleźć w czasie liniowym.*

Tw. *Obliczenie optymalnej strategii dla dowolnego częściowego porządku z elementem największym jest problemem* NP-*trudnym.*

Tw. *Istnieje wielomianowy* $O(\log n/(\log\log n))$-*przybliżony algorytm dla problemu szukania strategii wyszukiwania w częściowym porządku z elementem największym. Złożoność algorytmu wynosi* $O(n^3)$.

Pozostałe wyniki

Problem:

- dla danej gry dana jest dowolna konfiguracja
- czy bieżący gracz posiada strategię wygrywającą?
- jak trudna jest odpowiedź na powyższe pytanie?
- jak trudne jest obliczenie takiej strategii?

Tw. *Gra Phutball leży w klasie problemów* PSPACE-*trudnych*.

Klasy problemów:
$P \subseteq NP \subseteq PSPACE \subseteq EXPTIME$

Tw. *Gra Node Blocking leży w klasie problemów* PSPACE-*zupełnych*.

Pozostałe kierunki badań: przeszukiwanie grafów, szeregowanie *just-in-time*.

Dziękuję za uwagę