

Systemy operacyjne

26.11.2010

Zasady poprawności harmonogramu

- w każdej chwili procesor może wykonywać tylko jedno zadanie
- w każdej chwili zadanie może być obsługiwane przez co najwyżej jeden procesor
- Zadanie Z_j wykonuje się w całości w przedziale czasu $[r_j, \infty)$
- spełnione są ograniczenia kolejnościowe
- w przypadku zadań niepodzielnych każde zadanie wykonuje się nie przerywając w pewnym domknięto-otwartym przedziale czasowym, dla zadań podzielnych czasy wykonania tworzą skończoną sumę rozłącznych przedziałów

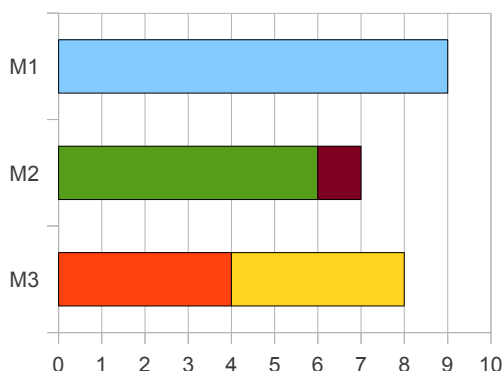
Kryteria kosztu harmonogramu

Położenie zadania Z_i w gotowym harmonogramie:

- moment zakończenia C_i (completion time)
- czas przepływu przez system (flow time) $\bar{F}_i = C_i - r_i$
- opóźnienie (lateness) $L_i = C_i - d_i$
- spóźnienie (tardiness) $T_i = \max\{C_i - d_i, 0\}$
- „znacznik spóźnienia” $U_i = w(C_i > d_i)$, a więc odpowiedź (0,1) na pytanie „czy zadanie się spóźniło”

Najczęściej stosowane kryteria kosztu harmonogramu:

- długość uszeregowania $C_{\max} = \max\{C_j; j=1, \dots, n\}$
- całkowity łączny czas zakończenia zadania $\sum C_j = \sum_{i=1}^n C_i$
- średni czas przepływu $\sum F = (\sum_{i=1}^n F_i) / n$



■ Z4 $C_{\max} = 9$

■ Z2

■ Z3

■ Z1

■ Z5

$$\sum C_j = 6 + 9 + 4 + 7 + 8 = 34$$

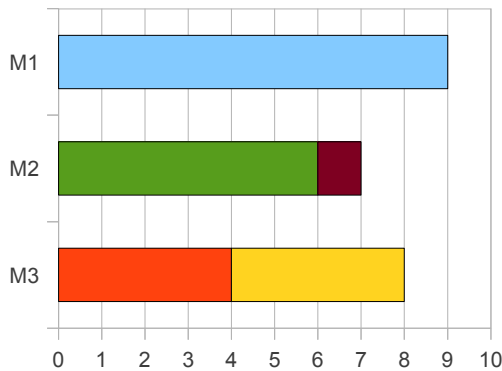
Uiszeregowanie na trzech maszynach równoległych $p_1, \dots, p_5 = 6, 9, 4, 1, 4$

- całkowity ważony czas zakończenia

$$\sum w_j C_j = \sum_{i=1}^n w_i C_i \quad w_1, \dots, w_5 = 1, 2, 3, 1, 1 \quad \sum w_j C_j = 6 + 18 + 12 + 7 + 8 = 51$$

Oparte na wymaganych terminach zakończenia:

- maksymalne opóźnienie $L_{\max} = \max\{L_j, j=1 \dots n\}$
- maksymalne spóźnienie $T_{\max} = \max\{T_j, j=1 \dots n\}$
- całkowite opóźnienie $\sum L_j = \sum_{i=1}^n L_i$
- całkowite spóźnienie $\sum T_j = \sum_{i=1}^n T_i$
- liczba spóźnionych zadań $\sum U_j = \sum_{i=1}^n U_i$
- można wprowadzić wagi zadań np. łączne całkowite spóźnienie $\sum w_j T_j = \sum_{i=1}^n w_i T_i$



Zadanie	Z1	Z2	Z3	Z4	Z5
d_i	7	7	5	5	8
L_i	-1	2	-1	2	0
T_i	0	2	0	2	0

$$L_{\max} = T_{\max} = 2$$

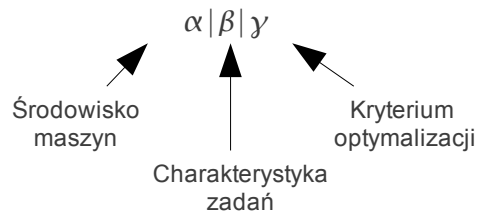
$$\sum T_i = 4, \sum L_i = 2$$

Niektóre kryteria są sobie równoważne:

$$\sum L_i = \sum C_i - \sum d_i$$

$$\bar{F} = \frac{\sum C_i}{n} - \frac{\sum r_i}{n}$$

Jak to opisać, notacja trójpolowa



α może mieć postać:

- P - procesory identyczne
- Q - procesory jednorodne
- R - procesory dowolne
- O - systemy otwarty (open shop)
- F - system przepływowy (flow shop)
- PF - permutacyjny system przepływowy
- J - system ogólny (job shop)

- po symbolu można dodać liczbę procesorów np. O4
- dla jednej maszyny piszemy liczbę 1 bez symbolu
- piszemy „-” przy braku maszyn (czynności bezstanowiskowe)

Ponadto:

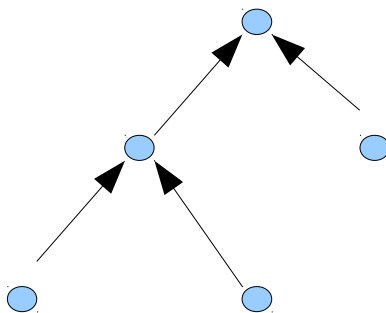
β – puste – to cechy domyślne, zadania są niepodzielne, z r_i , czasy wykonania i ewentualne wymagane terminy zakończeń d_i , dowolne

β możliwe wartości:

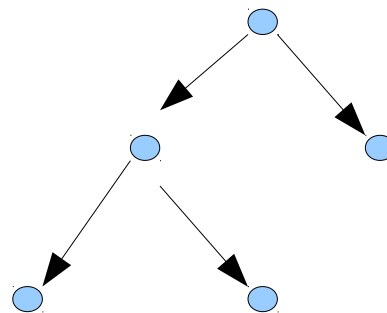
- pmtn – zadania podzielne (preemption)
- res – wymagane są dodatkowe zasoby
- prec – zadanie zależne
- r_j – występują różne wartości momentów przybycia
- $p_j=1$, lub EUT – zadanie o jednostkowym czasie wykonania
- $p_{ij} \in \{0,1\}$ lub ZUET - operacje w zadaniach są jednostkowe lub puste (procesory dedykowane)
- $C_j \leq d_i$ - istnieją wymagane i nieprzekraczalne terminy zakańczania zadań
- no-idle – procesory muszą pracować w sposób ciągły (bez okienek)
- no-wait – okienka między operacjami w zadaniach są zabronione (procesory dedykowane)

β możliwe wartości:

in-tree, out-tree, chains – różne szczególne postaci realizacji kolejnościowych



in-tree



out-tree

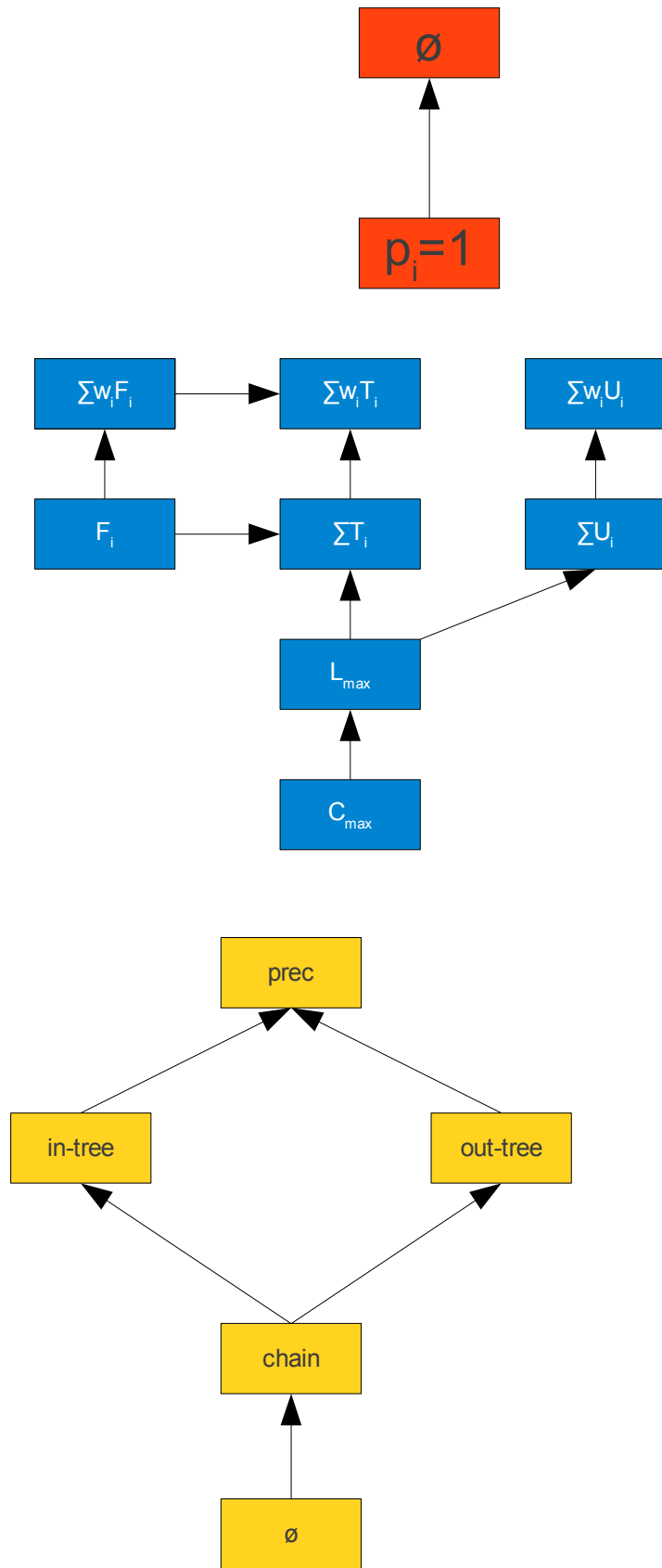
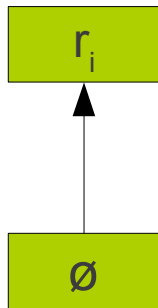
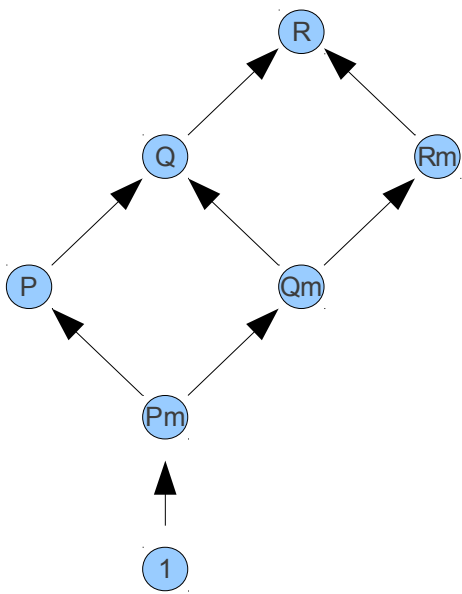
Przykłady:

P3|prec| C_{\max} - szeregowanie niepodzielnych zadań zależnych na trzech identycznych maszynach równoległych w celu zminimalizowania długości uszeregowania

R|pmtn,prec, r_i | ΣL_i - szeregowanie podzielnych zadań zależnych, z różnymi czasami przybycia i terminami zakończenia na równoległych dowolnych maszynach (liczba procesorów jest częścią danych) w celu zminimalizowania całkowitego opóźnienia

Redukcja pod-problemów ogólniejszych !! [egzamin]!!

Przykłady:



Zależność problemów szeregowania

Jeśli uwzględnimy tylko liczny maszyn 1,2,3,* to istnieje 4536 problemów z których:

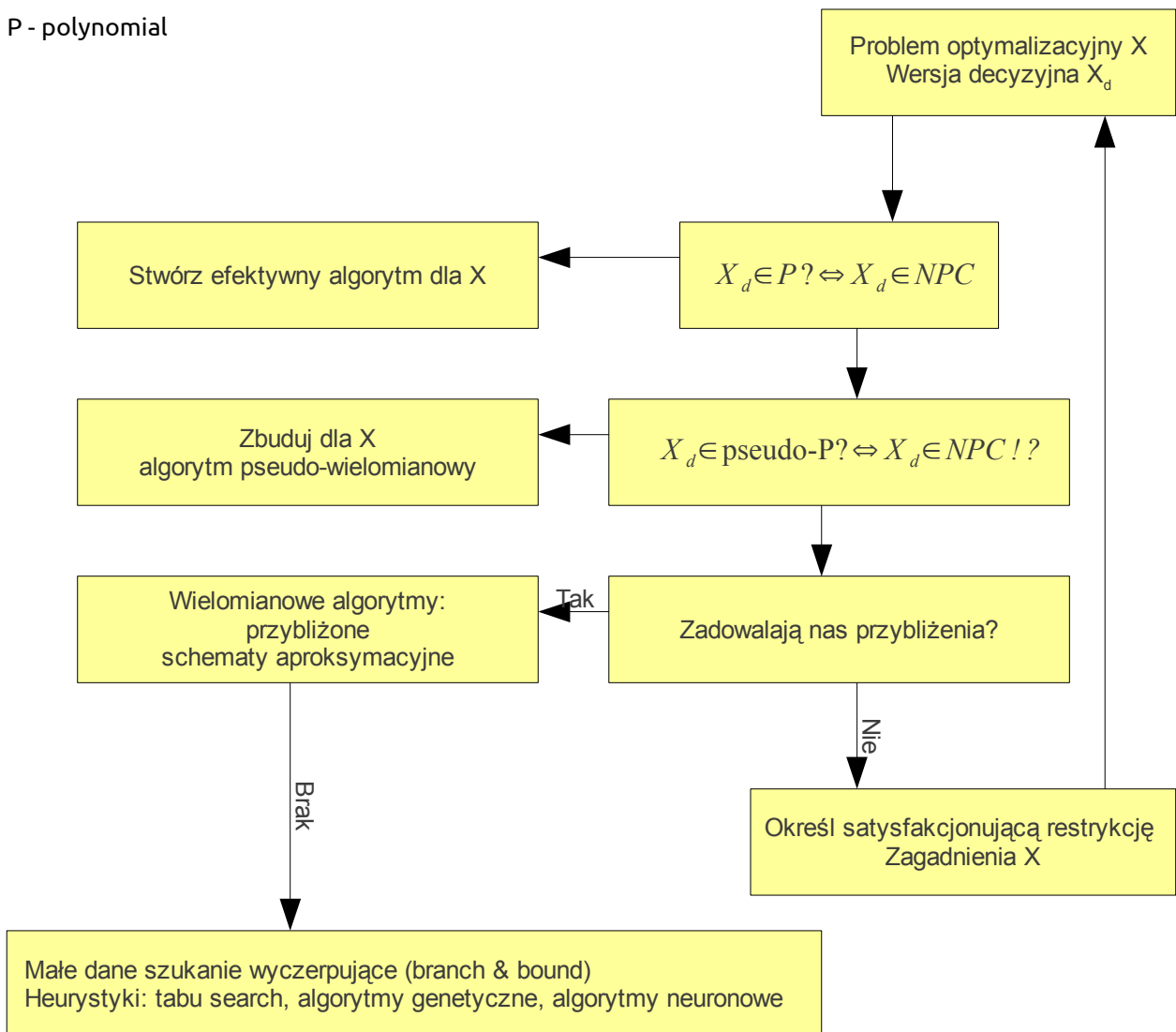
- 416 – wielomianowe
- 3817 – NP-trudne
- 303 – otwarte

Jak sobie radzić z NP-trudnością:

- wielomianowe algorytmy przybliżone o gwarantowanej dokładności względnej
- dokładne algorytmy pseudo-wielomianowe
- algorytmy dokładne, szybkie tylko w średnim przypadku
- heurystyki wyszukujące (np. tabu search, algorytmy genetyczne, algorytmy neuronowe)
- dla małych rozmiarów danych – wykładnicze przeszukiwanie wyczerpujące (np. branch & bound)

Ogólny schemat analizy zagadnień

P - polinomial



Zadania Systemów operacyjnych

- Definicja interfejsu użytkownika
- Udostępnianie interfejsu użytkownika
- Udostępnianie środowiska do wykonywania programów użytkownika
- Sterowanie urządzeniami wejścia i wyjścia
- Zarządzanie zasobami

Zarządzanie zasobami

- Przydział zasobów
- Synchronizacja dostępu do zasobów
- Ochrona i autoryzacja dostępu do zasobów
- Odzyskiwanie zasobów
- Rozliczanie – gromadzenie danych o wykorzystaniu zasobów

Przydział zasobów – realizacja żądań dostępu do zasobów w taki sposób , aby zasoby były używane zgodnie z intencją użytkowników

Synchronizacja dostępu do zasobów – strategia przydziału zasobów gwarantująca bezpieczeństwo, żywotność, brak zakleszczenia, sprawiedliwość oraz optymalność ich wykorzystania

Ochrona i autoryzacja dostępu do zasobów – dopuszczanie możliwości użytkownika zasobu tylko przez osoby uprawnione i tylko w zakresie przydzielonych im uprawnień

Odzyskiwanie zasobów – dołączanie zwolnionych zasobów do zbioru zasobów wolnych po zakończeniu ich użytkowania

Rozliczanie – rejestrowanie i udostępnianie informacji o wykorzystaniu zasobów w celach kontrolnych i optymalizacyjnych