Wrocław, 21 kwietnia 2020

Sterowanie **P**rocesami **D**yskretnymi - Laboratorium
Problem *FSP* - *Wprowadzenie*

prowadzący: *mgr inż. Radosław Idzikowski*

1 Wprowadzenie

Celem laboratorium jest zapoznanie się z podstawami teorii szeregowania zadań na przykładzie wielo-maszynowego problemu przepływowego (**F**low **S**hop **P**roblem)

2 Problem

Problem $FP||C_{\max}$ jest szczególnym przypadkiem problemu ogólniejszego $F||C_{\max}$, gdzie na każdej maszynie będziemy mieli taką samą kolejność wykonywania zadań. Mamy zbiór n zadań wykonywanych na maszynach:

$$\mathcal{J} = \{1, 2, \ldots, n\}, \tag{1}$$

które należy wykonać na m maszynach:

$$\mathcal{M} = \{1, 2, \ldots, m\}, \tag{2}$$

każde j-te zadanie składa się dokładnie z m operacji

$$\mathcal{O}_j = \{o_{1j}, o_{2j}, \ldots, o_{mj}\}, \tag{3}$$

ponieważ każda operacja o_{ij} z zadania j jest wykonywana nieprzerwanie na innej maszynie i. Czas wykonania *performed time*) operacji o_{ij} wynosi p_{ij}. W ramach jednej maszyny może wykonywać się naraz tylko jedno zadanie. W problemie przepływowym w ramach zadań musi być zachowany porządek technologiczny, tzn. aby mogła się zacząć wykonywać kolejna operacja najpierw musi się wykonać operacja poprzednia z tego samego zadania. Przez π oznaczymy kolejność wykonywania zadań (W permutacyjnym problemie przepływowym na każdej maszynie mamy tą samą kolejność wykonywania zadań).

W celu utworzenia harmonogramu dla zadanej permutacji pi musi utworzyć macierz S momentów rozpoczęcia operacji oraz macierz C momentów zakończenia operacji. Dla permutacji naturalnej $pi = (1, 2, \ldots, n)$, gdzie $S_{ij} = S_{i\pi(j)}$ i $C_{ij} = C_{i\pi(j)}$:

Macierz S momentów rozpoczęcia:

$$S = \begin{bmatrix} S_{11} & S_{12} & S_{13} & \ldots & S_{1n} \\ S_{21} & S_{22} & S_{23} & \ldots & S_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ S_{m1} & S_{m2} & S_{m3} & \ldots & S_{mn} \end{bmatrix} \tag{4}$$

Macierz C momentów zakończenia:

$$C = \begin{bmatrix} C_{11} & C_{12} & C_{13} & \ldots & C_{1n} \\ C_{21} & C_{22} & C_{23} & \ldots & C_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ C_{m1} & C_{m2} & C_{m3} & \ldots & C_{mn} \end{bmatrix} \tag{5}$$

Przy układaniu harmonogramu należy pamiętać o dwóch ograniczeniach. Po pierwsze, że aby mogła się zacząć wykonywać kolejna operacja $o_{i\pi(j+1)}$ na tej samem maszynie, najpierw musi się zakończyć poprzednia operacja $o_{i\pi(j)}$:

$$S_{i\pi(j+1)} \geqslant C_{i\pi(j)} \tag{6}$$

Po drugie, musi być zachowany porządek technologiczny, więc aby operacja o_{i+1j} z zadania j mogła się zacząć wykonywać najpierw musi zostać wykonana operacja $o_{i\pi(j)}$ na maszynie poprzedniej:

$$S_{i+1\pi(j)} \geqslant C_{i\pi(j)} \tag{7}$$

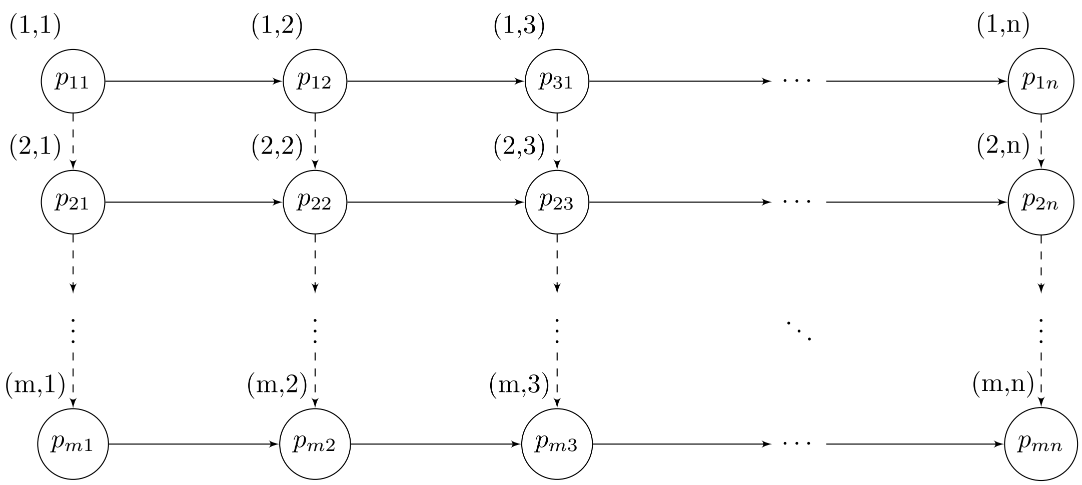
Uwzględniając oba ograniczenia możemy wyznaczyć czas rozpoczęcia operacji $o_{i\pi(j)}$ wzorem:

$$S_{i\pi(j)} = max\{C_{i-1\pi(j)}, C_{i\pi(j-1)}\}. \tag{8}$$

Rozpatrywanym kryterium optymalizacyjnym jest czas zakończenia wszystkich zadań $C_{\max}$, ponieważ wszystkie zadania muszą się zakończyć na maszynie m oraz wiadomo, które zadanie wykona się jako ostatnio to:

$$C_{\max} = C_{m,\pi(n)}. \tag{9}$$

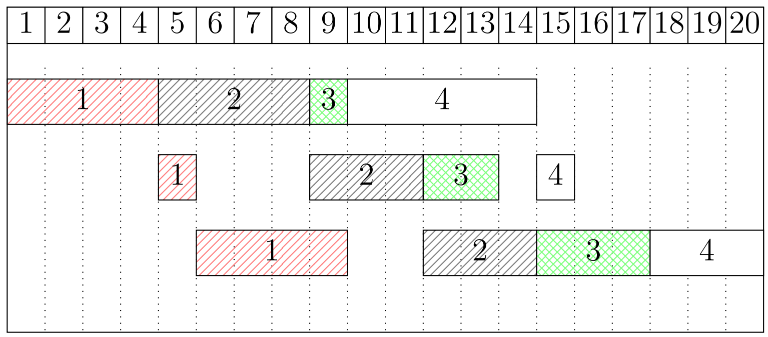
Dla problemu $FP||C_{\max}$ i dla permutacji naturalnej $pi = (1, 2, \ldots, n)$ możemy zbudować graf rozwiązania $\mathbf{G}(\pi)$



Rysunek 1: Graf $\mathbf{G}(\pi)$

3 Przykład

Na Rysunku 1 mamy graficzne rozwiązanie problemu dla danych z Tabeli 1. Czas wykonywania operacji na maszynie jest zaprezentowany w formie bloczku. Operacje w ramach jednego zadanie są oznaczone tym samym numerkiem i kolorem. Na schemacie dobrze widać, że na maszynach pojawiają się przerwy (przestoje).



Rysunek 2: Permutacyjny problem przepływowy: $n = 4$ i $m = 3$ dla $\pi = (1, 2, 3, 4)$

Tabela 1: Instancja o rozmiarze $n = 4$ i $m = 3$

zadanie	p_{1j}	p_{2j}	p_{3j}
1	4	1	4
2	4	3	3
3	1	2	3
4	5	1	3

4 Struktura pliku

1. n m
2. 1 p_{11} 2 p_{21} ... m p_{m1}
3. 1 p_{12} 2 p_{22} ... m p_{m2}
4. 1 p_{13} 2 p_{23} ... m p_{m3}

...

$n+1$ 1 p_{1n} 2 p_{2n} ... m p_{mn}

W pierwszej linii jest n – liczba zadań oraz m – liczba maszyn. W kolejnych n liniach znajdują się parametry kolejnych zadań poprzedzone indeksami maszyn.

5 Zadanie

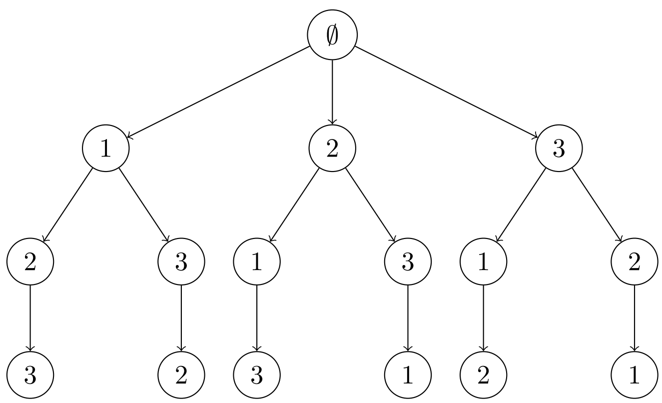
W trakcje zajęć należy:

1. Wczytać danie z pliku.
2. Napisać funkcje celu dla kryterium C_{max}.
3. Napisać algorytm przeglądu zupełnego.
4. Sprawdzić poprawność wyników dla przeglądu zupełnego.

Proszę pamiętać, że kompletne rozwiązanie to permutacja i wartość $C_{\max}$ dla niej.

6 Przegląd zupełny

Sprawdzenie wszystkich możliwych kombinacji (*Brute Force*). Dla zbioru $\{1, 2, 3\}$ mamy $n!$ kombinacji: $(1, 2, 3)$, $(1, 3, 2)$, $(2, 1, 3)$, $(2, 3, 1)$, $(3, 1, 2)$ i $(3, 2, 1)$.



Rysunek 3: Drzewa przestrzeni stanów dla $n = 3$

opracował: *Radosław Idzikowski*