

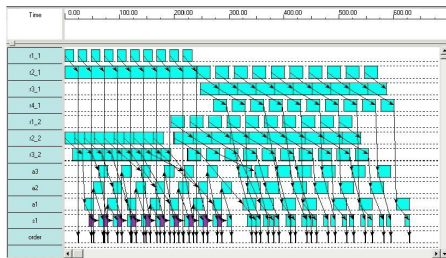
PB165 Grafy a sítě: 5. Plánování a rozvrhování

Plánování a rozvrhování

- 1 Terminologie a klasifikace
 - Úvod
 - Vlastnosti stroje
 - Omezení
 - Optimalizace
- 2 Grafová reprezentace (1.část)
 - Precedenční omezení
 - Disjunktivní grafová reprezentace

Rozvrhování a plánování (scheduling)

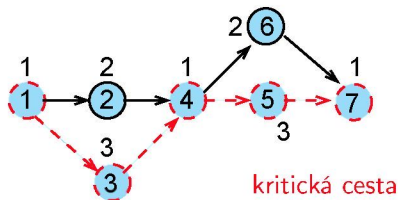
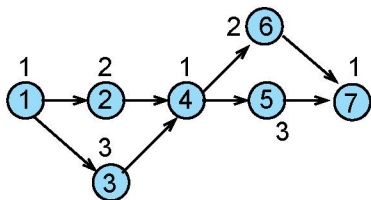
- Rozvrhování
 - optimální alokace/přirazení zdrojů v čase množině úloh
 - omezené množství zdrojů
 - maximalizace zisku za daných omezení
- Zdroj
 - kapacita
 - dostupnost v čase
 - rychlost
- Úloha
 - nejdřívejší startovní čas
 - nejpozdější koncový čas
 - doba trvání (ref. zdroj)
 - počet zdrojů
 - alternativní zdroje



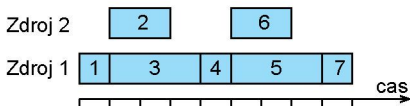
Visopt ShopFloor System

Příklad

- Rozvrhování 7 úloh na 2 zdrojích
 - doba trvání úlohy + precedenční podmínky
 - nalezení rozvrhu tak, aby se minimalizovala doba nutná na realizaci všech úloh



- Možný rozvrh
 - na kritické (nejdelší) cestě nesmí vzniknout zdržení



Úlohy, stroje

- **Stroje (zdroje, prostředky)** $i = 1, \dots, m$
- **Úlohy (aktivity)** $j = 1, \dots, n$
- (i, j) operace nebo provádění úlohy j na stroji i
- Statické parametry úlohy
 - doba trvání p_{ij}, p_j : doba provádění úlohy j na stroji i
 - termín dostupnosti j (*release date*) r_j : nejdřívejší čas, ve kterém může být úloha j prováděna
 - termín dokončení (*due date*) d_j : čas, do kdy musí být úloha j nejpozději dokončena
 - váha w_j : důležitost úlohy j relativně vzhledem k ostatním úlohám v systému
- Dynamické parametry úlohy
 - čas startu úlohy (*start time*) S_{ij}, S_j : čas, kdy začne provádění úlohy j na stroji i
 - čas konce úlohy (*completion time*) C_{ij}, C_j : čas, kdy je dokončeno provádění úlohy j na stroji i

Grahamova klasifikace

Grahamova klasifikace $\alpha|\beta|\gamma$

používá se pro popis rozvrhovacích problémů

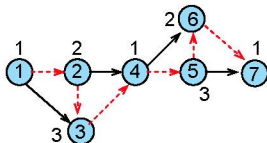
- α : charakteristiky stroje
 - popisuje způsob alokace úloh na stroje
- β : charakteristiky úloh
 - popisuje omezení aplikovaná na úlohy
- γ : optimalizační kritéria

<http://www.mathematik.uni-osnabrueck.de/research/OR/class/>

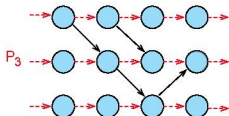
- složitost a algoritmy pro jednotlivé rozvrhovací problémy

Vlastnosti stroje

- Jeden stroj 1: $1 | \dots | \dots$



- Identické paralelní stroje P_m
 - m strojů (zapojených paralelně)
 - úloha j je jedna operace a ta může být prováděna na libovolném z m strojů

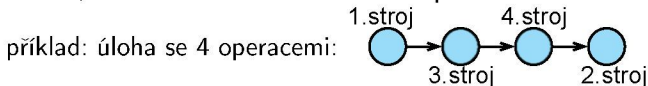


- Paralelní stroje s různou rychlostí Q_m
 - příklad: několik počítačů s různou rychlostí procesoru
- Nezávislé paralelní stroje s různou rychlostí R_m
 - stroje mají různou rychlost pro různé úlohy
 - příklad: vektorový počítač počítá vektorové úlohy rychleji než klasické PC

Multi-operační (*shop*) problémy

- Multi-operační (*shop*) problémy

- jedna úloha je prováděna postupně na několika strojích
- lze říci, že úloha se skládá z několika operací



- Job shop* J_m

- m strojů
- úloha musí být prováděna na všech strojích
- každá úloha má předem definované pořadí, ve kterém je prováděna na jednotlivých strojích

- Multi-operační problémy jsou klasické

detailně studované problémy **operačního výzkumu**

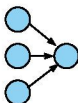
- reálné problémy mnohem komplikovanější
- metody řešení lze použít jako základ pro řešení složitějších problémů
- př. automobilová výrobní linka

Omezení

- Precedenční podmínky

prec

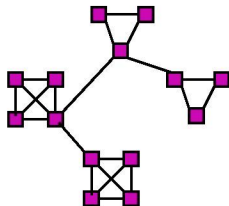
- úloha může být prováděna až po skončení další(ch) úloh



- Vhodnost stroje

 M_j

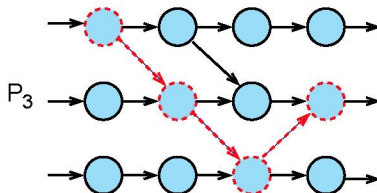
- podmnožina strojů M_j , na níž lze provádět úlohu j
- př. úloha může být prováděna pouze na těch strojích v počítačové síti, kde jsou dostupná data



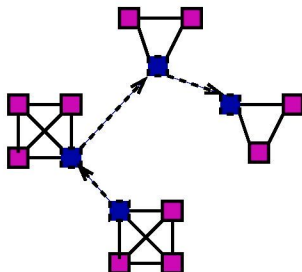
- Omezení na pracovní sílu

 W_l

- stroje mohou potřebovat operátory a úlohy lze provádět pouze pokud jsou dostupní W operátorů
- mohou existovat různé skupiny operátorů se specifickou kvalifikací W_l je počet operátorů ve skupině l
- př. hw zařízení (mikroskop) dostupné pouze pro jednu úlohu



- Směrovací (*routing*) omezení
 - udávají, na kterých strojích musí být úloha prováděna
 - vazba na směrování v počítačových sítích
 - pořadí provádění úlohy v multi-operačních problémech



- Nastavovací (*setup*) doba a cena

 $s_{ijk}, s_{jk} \quad c_{ijk}, c_{jk}$

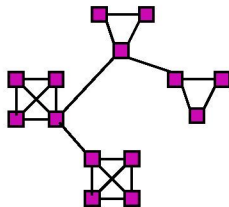
- závislé na posloupnosti provádění
- s_{ijk} čas nutný pro provádění úlohy k po úloze j na stroji i
- c_{ijk} cena nutná . . .
- s_{jk}, c_{jk} nastavovací doba a cena nezávislá na stroji
- příklad: plnění limonád do lahví

- Doprava a komunikace

 t_{jkl}, t_{kl}, t_j

- t_{jkl} doba na přepravu ze stroje k na stroj l pro úlohu j
- t_{kl} doba nezávislá na úloze
- t_j doba nezávislá na strojích
- doba nutná na přepravu mezi dvěma zařízeními
- omezení na množství a možnou dobu přepravy
- propustnost linky a vzdálenost uzlů v počítačové síti

t_{kl} dáno vzdáleností uzlů v síti/grafu:



Optimalizace: *makespan*

- Maximální čas konce úloh (*makespan*)

$$C_{\max} = \max(C_1, \dots, C_n)$$

- Minimalizace makespan často
 - maximalizuje výkon (*throughput*)
 - zajišťuje rovnoměrné zatížení strojů (*load balancing*)
- Velmi často používané kritérium

Optimalizace: termín dokončení d_j 

- Zpoždění (*lateness*) $L_j = C_j - d_j$
- Minimalizace *lateness*

$$L_{\max} = \max(L_1, \dots, L_n)$$

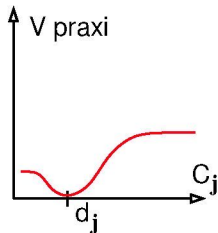
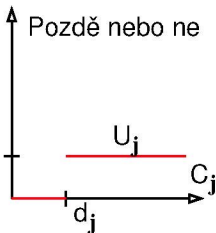
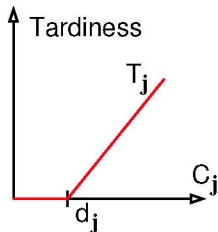
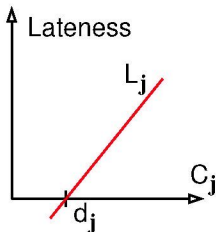
- Nezáporné zpoždění (*tardiness*) $T_j = \max(C_j - d_j, 0)$
- Minimalizace maximální *tardiness*

$$\sum_{j=1}^n T_j \quad \text{celkové zpoždění úloh}$$

- Minimalizace maximální vážené *tardiness*

$$\sum_{j=1}^n w_j T_j \quad \text{vážené zpoždění úloh}$$

Termín dokončení a grafy



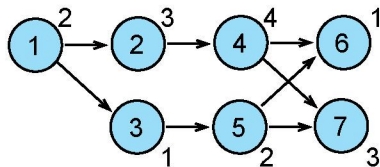
- 1 Terminologie a klasifikace
 - Úvod
 - Vlastnosti stroje
 - Omezení
 - Optimalizace

- 2 Grafová reprezentace (1.část)
 - Precedenční omezení
 - Disjunktivní grafová reprezentace

Precedenční omezení

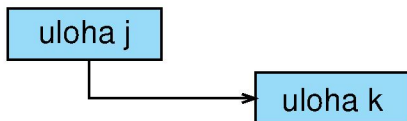
- Úloha může být prováděna až po skončení další(ch) úloh
 - úloha a před úlohou b : $a \rightarrow b : S_a + p_a \leq S_b$
- Orientovaný acyklický vrcholově ohodnocený graf**
 - uzly reprezentují úlohy
 - hrany reprezentují precedenční podmínky
 - ohodnocení vrcholu reprezentuje doba trvání
 - graf bez cyklů (pro cyklický graf neexistuje žádné řešení)

| Úloha | Doba trvání | Předchůdci |
|-------|-------------|------------|
| 1 | 2 | – |
| 2 | 3 | 1 |
| 3 | 1 | 1 |
| 4 | 4 | 2 |
| 5 | 2 | 3 |
| 6 | 1 | 4,5 |
| 7 | 3 | 4,5 |

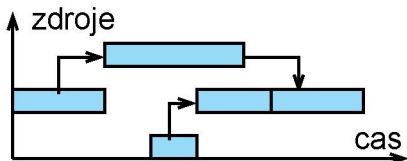


Úloha jako obdélník

- Úloha jako uzel lze převést na **úloha jako obdelník**

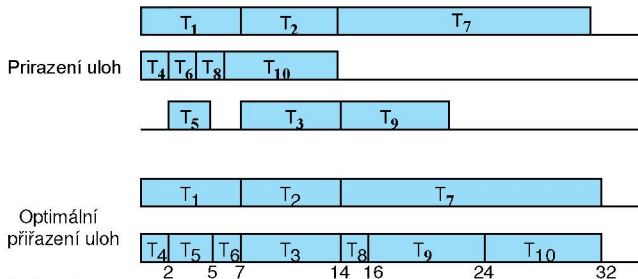
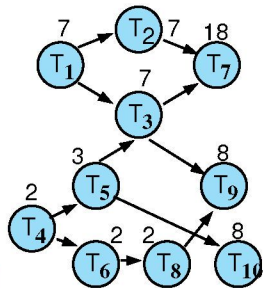


- Horizontální strany obdelníku použity jako časové osy odpovídající době provádění úlohy



Montáž kola

- 3 pracovníci
- Každá úloha má určenou dobu trvání
- Precedenční podmínky
- Nepreemptivní (úlohy nelze přerušit)



Precedenční omezení: aplikace

- Zprostředkování, instalace a testování rozsáhlého počítačového systému
 - projekt zahrnuje
 - evaluace a výběr hardware, vývoj software, nábor a školení lidí, testování a ladění systému, ...
 - precedenční vztahy
 - některé úlohy mohou být prováděny paralelně
 - úloha musí být realizována až po dokončení jiných úloh
 - cíl: minimalizovat čas na realizaci celého projektu
- Obecně: problémy plánování projektu
- Plánování workflows
 - 1 orientovaný acyklický graf pro provádění úloh na počítačové síti
 - 2 obecné rozšíření: cyklické grafy + podmínky vyhodnocení cyklů

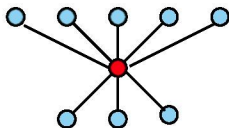
Disjunktivní grafová reprezentace a multi-operační rozvrhování

- n úloh
- m strojů
- Jedna úloha je prováděna postupně na několika strojích
- Operace (i, j) : provádění úlohy j na stroji i
- Pořadí operací úlohy je stanoveno:
 - $(i, j) \rightarrow (k, j)$ specifikuje, že úloha j má být prováděna na stroji i dříve než na stroji k
- p_{ij} : trvání operace (i, j)
- Cíl: rozvrhovat úlohy na strojích
 - bez překrytí na strojích
 - bez překrytí v rámci úlohy
 - minimalizace *makespan* C_{max}

Aplikace: automobilová montážní linka

- Rozdílné typy aut na montážní lince
 - dvou-dveřové kupé, čtyř-dveřový sedan, ...
 - rozdílné barvy
 - rozdílné vybavení: automatická vs. manuální převodovka, posuvná střech, ...

- **Kritická místa** (*bottlenecks*)

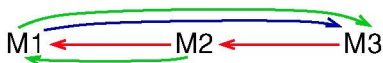


- výkon stroje ovlivňuje tempo výroby
 - např. lakování (změna barvy vyžaduje časově náročné čištění)
- Cíl
 - maximalizace výkonnosti vhodným seřazením automobilů,
 - rovnoměrná pracovní zátěž na jednotlivých výrobních místech

Příklad: *job shop* problém

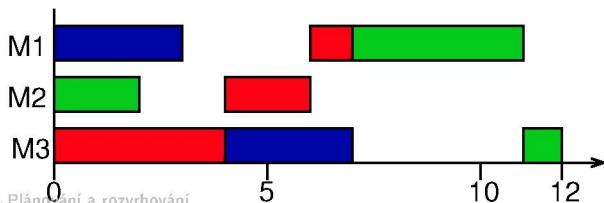
Data:

- stroje: $M1, M2, M3$
- úlohy: $J1 : (3, 1) \rightarrow (2, 1) \rightarrow (1, 1)$
 $J2 : (1, 2) \rightarrow (3, 2)$
 $J3 : (2, 3) \rightarrow (1, 3) \rightarrow (3, 3)$



- doby trvání: $p_{31} = 4, p_{21} = 2, p_{11} = 1$
 $p_{12} = 3, p_{32} = 3$
 $p_{23} = 2, p_{13} = 4, p_{33} = 1$

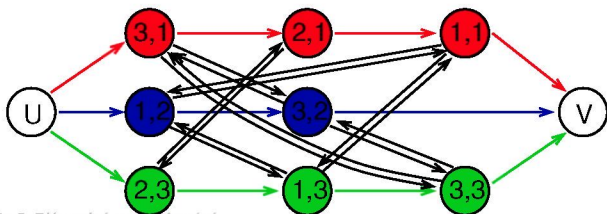
Řešení:



Disjunktivní grafová reprezentace

Graf $G = (N, A \cup B)$

- uzly odpovídají operacím $N = \{(i, j) | (i, j) \text{ je operace}\}$
- **konjunktivní hrany** A reprezentují pořadí operací úlohy
 - $(i, j) \rightarrow (k, j) \in A \iff$ operace (i, j) předchází (k, j)
- **disjunktivní hrany** B reprezentují konflikty na strojích
 - dvě operace (i, j) a (i, l) jsou spojeny dvěma opačně orientovanými hranami
- dva pomocné uzly U a V reprezentující zdroj a stok
- hrany z U ke všem prvním operacím úlohy
- hrany ze všech posledních operací úlohy do V



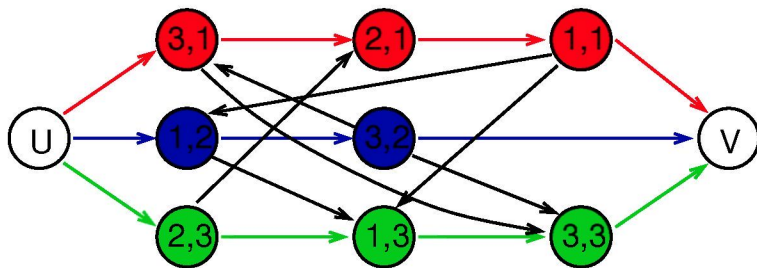
Výběr hran

Pojmy:

- Podmnožina $D \subset B$ je nazývána **výběr**, jestliže obsahuje z každého páru disjunktivních hran právě jednu
- Výběr D je **splnitelný**, jestliže výsledný orientovaný graf $G(D) = (N, A \cup D)$ je acyklický
 - jedná se o graf s konjunktivními hranami a vybranými diskjunktivními hranami

Poznámky:

- splnitelný výběr určuje posloupnost, ve které jsou operace prováděny na strojích
- každý (konzistentní) rozvrh jednoznačně určuje splnitelný výběr
- každý splnitelný výběr jednoznačně určuje (konzistentní) rozvrh

Příklad: **nesplnitelný** výběr

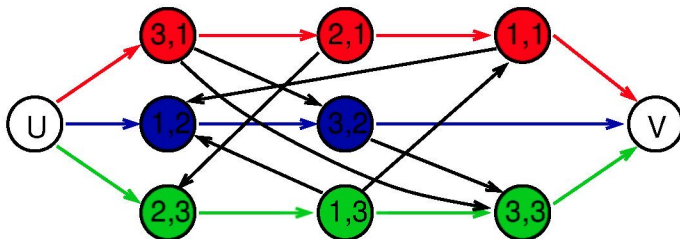
V grafu existuje v důsledku nevhodného výběru hran cyklus:

- $(1, 2) \rightarrow (3, 2)$
- $(3, 2) \rightarrow (3, 1) \rightarrow (2, 1) \rightarrow (1, 1) \rightarrow (1, 2)$

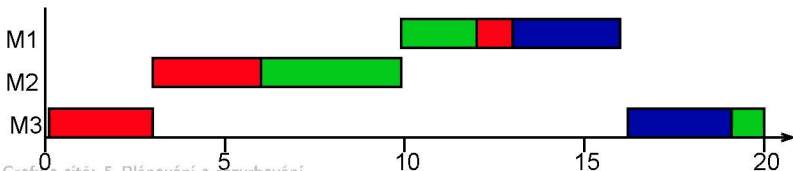
\Rightarrow nelze splnit (k tomuto výběru neexistuje rozvrh)

Příklad: splnitelný výběr

Jakým způsobem nalézt rozvrh pro daný splnitelný výběr?



Tedy: jakým způsobem lze nalézt tento odpovídající rozvrh:



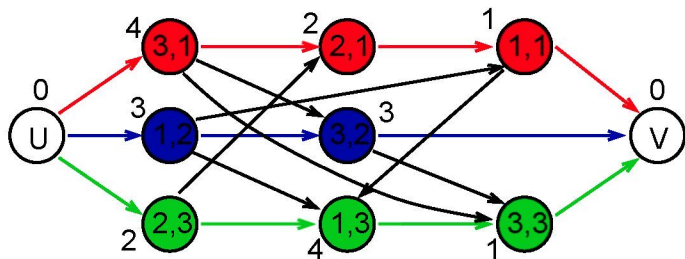
Výpočet rozvrhu pro výběr

Metoda: výpočet **nejdelších cest** z U do dalších uzlů v $G(D)$

Technický popis:

- uzly (i, j) mají **ohodnocení** p_{ij} , uzel U má ohodnocení 0
- **délka cesty** i_1, i_2, \dots, i_r : součet ohodnocení uzlů i_1, i_2, \dots, i_{r-1}
- spočítej délku l_{ij} **nejdelší cesty** z U do (i, j) a V
 - např. použitím Dijkstrova algoritmu
- zahaj operaci (i, j) v čase l_{ij}
- délka **nejdelší cesty** z U do V je rovna *makespan*
 - tato cesta je **kritická cesta**

Výpočet rozvrhu pro výběr

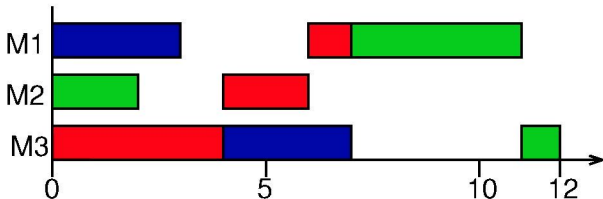


Výpočet l_{ij} :

| uzel | (3,1) | (1,2) | (2,3) | (2,1) | (3,2) | (1,1) | (1,3) | (3,3) | V |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|----|
| délka | 0 | 0 | 0 | 4 | 4 | 6 | 7 | 11 | 12 |

Příklad: výběr pro daný rozvrh

Nalezněte výběr hran pro daný rozvrh:



Konstrukce odpovídajícího výběru: vybereme disjunktivní hrany, které odpovídají uspořádání operací úlohy v rozvrhu

