

PB165 Grafy a sítě: Barvení grafu, transport a sítě

1 Barvení grafu

- Popis problému a jednoduché řešení
- Přiřazení místností
- Rezervační problém
- Rozvrhování operátorů

2 Transport

- Doba na nastavení
- Doba na dopravu

3 Plánování na počítačové síti

- Úvod
- Paralelní úlohy s komunikací

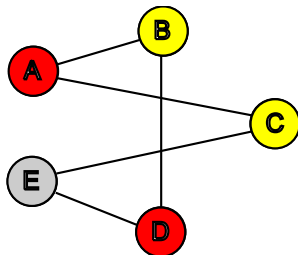
Problém barvení grafu

- Je možné obarvit vrcholy grafu s použitím n barev tak, aby žádné dva sousední vrcholy nebyly obarveny stejnou barvou?

Chromatické číslo grafu

- Minimální počet barev n nutný k obarvení grafu tak, by žádné dva sousední vrcholy nebyly obarveny stejnou barvou.

NP-úplný problém



Barvení grafu a rozvrhování

- Rezervační problémy
- Přiřazení místností
- Rozvrhování operátorů

Heuristiky pro barvení grafu se saturací

- **Stupeň uzlu**
 - počet hran spojených s uzlem
- **Úroveň saturace**
 - počet různých barev spojených s uzlem
- Intuice
 - obarví uzly s vyšším stupněm dříve
 - obarví uzly s vyšší úrovní saturace dříve

Heuristiky pro barvení grafu se saturací

- **Stupeň uzlu**
 - počet hran spojených s uzlem
- **Úroveň saturace**
 - počet různých barev spojených s uzlem
- Intuice
 - obarvi uzly s vyšším stupněm dříve
 - obarvi uzly s vyšší úrovní saturace dříve
- **Algoritmus**
 - 1 uspořádej uzly v klesajícím pořadí podle jejich stupně
 - 2 použij barvu 1 pro první uzel
 - 3 vyber neobarvený uzel s maximální úrovní saturace v případě volby z nich vyber uzel s maximálním stupněm v neobarveném podgrafu
 - 4 obarvi vybraný uzel s nejmenší možnou barvou
 - 5 jestliže jsou všechny uzly obarveny STOP jinak běž na krok 3

- **Problém přřazení místností**

- úloha = předmět s několika schůzkami týdně
- zdroj = místnost
- dva předměty nesmí být zároveň vyučovány ve stejné místnosti
- všechny schůzky předmětu musí být vyučovány ve stejné místnosti

rozvrh: přřazení místnosti každému předmětu

možné řešení:

- nalezení rozvrhu vzhledem k danému počtu místností
- nalezení rozvrhu s minimálním počtem místností

- **Přřazení místností jako barvení grafu**

- vrchol: předmět
- hrana: mezi předměty, které vyžadují stejný čas výuky
- barva vrcholu: odpovídá vybrané místnosti (zdroji)
 - sousedící vrcholy/předměty musí mít různé barvy/místnosti, protože vyžadují stejný čas

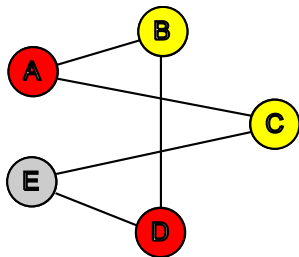
Přiřazení místností: příklad

Kolik místností je třeba k rozvrhování těchto předmětů?

předmět	A	B	C	D	E
časy	(1,4)	(1,3)	(2,4)	(3,5)	(2,5)
stupeň	2	2	2	2	2

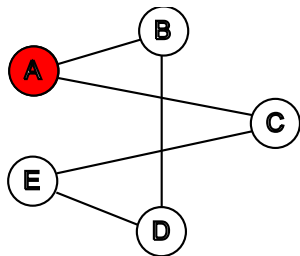
Řešení:

místnost červená žlutá žlutá červená šedá



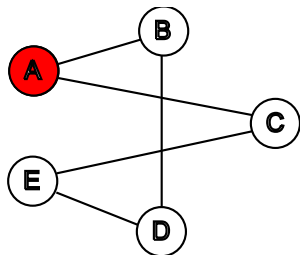
čas/předmět	A	B	C	D	E
1	+	+	-	-	-
2	-	-	+	-	+
3	-	+	-	+	-
4	+	-	+	-	-
5	-	-	-	+	+

Přřazení místností: příklad (pokračování)

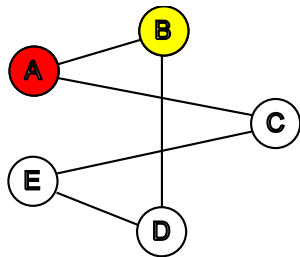


předmět	A	B	C	D	E
saturace	-	1	1	0	0
stupeň neob.	-	1	1	2	2

Přřazení místností: příklad (pokračování)

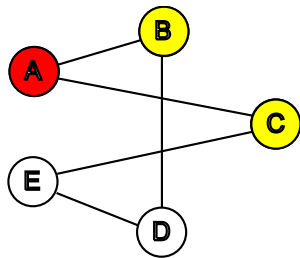


předmět	A	B	C	D	E
saturace	-	1	1	0	0
stupeň neob.	-	1	1	2	2



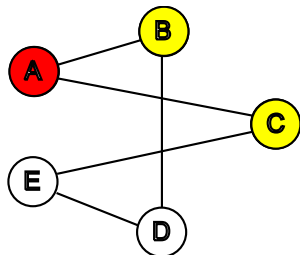
předmět	A	B	C	D	E
saturace	-	-	1	1	0
stupeň neob.	-	-	1	1	2

Přřazení místností: příklad (dokončení)

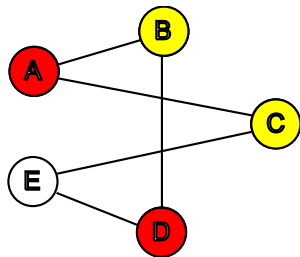


předmět	A	B	C	D	E
saturace	-	-	-	1	1
stupeň neob.	-	-	-	1	1

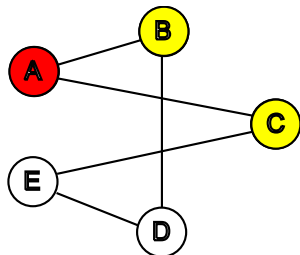
Přřazení místností: příklad (dokončení)



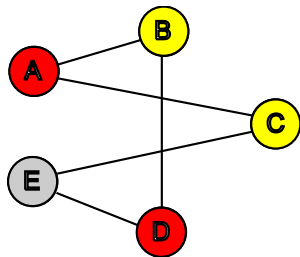
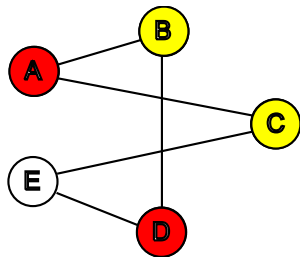
předmět	A	B	C	D	E
saturovac	-	-	-	1	1
stupeň neob.	-	-	-	1	1



Přřazení místností: příklad (dokončení)



předmět	A	B	C	D	E
saturovac	-	-	-	1	1
stupeň neob.	-	-	-	1	1



- Příklady
 - rezervace aut
 - rezervace pokojů v hotelu
 - rezervace strojů v továrně
- Určen časový interval pro každou rezervaci
 - $p_j = r_j - d_j$
 - p_j doba trvání úlohy
 - r_j termín dostupnosti
 - d_j termín dokončení
- Každá rezervace vyžaduje zdroj (auto, pokoj, stroj)
- Možné řešení
 - lze rezervace realizovat s daným počtem zdrojů?
 - kolik zdrojů je třeba ke splnění rezervací?

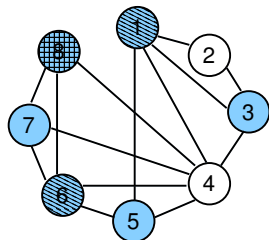
Rezervační problém jako barvení grafu

- Vrchol: rezervace
- Hrana: pokud se dvě rezervace překrývají v čase
- Barva vrcholu: odpovídá vybranému zdroji
 - sousedící vrcholy/rezervace musí mít různé barvy/zdroje, protože se překrývají v čase
 - kolik zdrojů je třeba ke splnění rezervací = chromatické číslo
 - lze rezervace realizovat s daným počtem zdrojů = existuje barvení s daným počtem barev

• Příklad:

j	1	2	3	4	5	6	7	8
r_j	0	1	1	3	4	5	6	6
d_j	5	3	4	7	6	7	9	8

Odpovídající problém barvení grafu:



Rozvrhování operátorů

- Zadáno několik různých operátorů
- Úloha potřebuje jeden nebo více specifických operátorů
- Úlohy vyžadující stejného operátora nemohou běžet zároveň
- Jednotková doba trvání úlohy
- Možné řešení:
 - rozvržení všech úloh v rámci časového horizontu
 - nalezení minimálního času (=makespan) tak, aby byly provedeny všechny úlohy
- Rozvrhování operátorů jako barvení grafu
 - vrchol: úloha
 - hrana: mezi úlohami, které potřebují stejného operátora
 - barva vrcholu: čas pro realizaci úlohy
 - sousedící úlohy/vrcholy musí mít různý čas/barvu, protože vyžadují stejného operátora
 - rozvržení všech úloh v rámci časového horizontu = existuje barvení s daným počtem barev
 - makespan = chromatické číslo grafu

Příklad: plánování schůzek

Vytvoř rozvrh pro 5 schůzek se 4 lidmi

- schůzka = úloha, člověk = operátor
- všechny schůzky trvají jednu hodinu

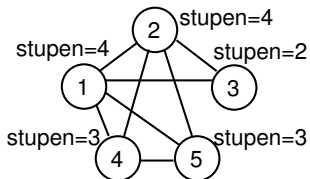
	1	2	3	4	5
Joe	1	1	0	1	1
Lisa	1	1	1	0	0
Jane	1	0	1	0	0
Larry	0	1	0	1	1

Příklad: plánování schůzek

Vytvoř rozvrh pro 5 schůzek se 4 lidmi

- schůzka = úloha, člověk = operátor
- všechny schůzky trvají jednu hodinu

	1	2	3	4	5
Joe	1	1	0	1	1
Lisa	1	1	1	0	0
Jane	1	0	1	0	0
Larry	0	1	0	1	1

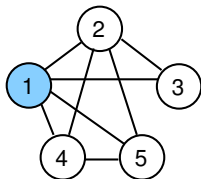
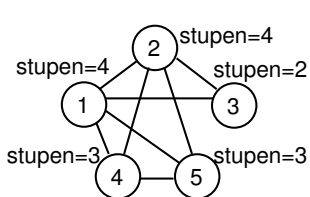


Příklad: plánování schůzek

Vytvoř rozvrh pro 5 schůzek se 4 lidmi

- schůzka = úloha, člověk = operátor
- všechny schůzky trvají jednu hodinu

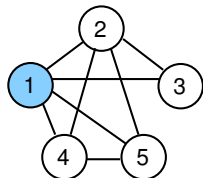
	1	2	3	4	5
Joe	1	1	0	1	1
Lisa	1	1	1	0	0
Jane	1	0	1	0	0
Larry	0	1	0	1	1



Můžeme vybrat buď
úlohu 1 nebo úlohu 2

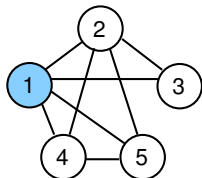
Např. vybereme 1 a obarvíme
barvou 1

Příklad: plánování schůzek (dokončení)

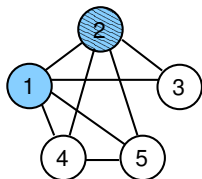


Úroveň saturace = 1 pro všechny úlohy
Vyber 2 vzhledem k nejvyššímu stupni

Příklad: plánování schůzek (dokončení)

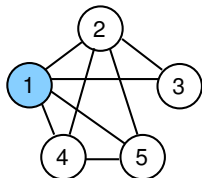


Úroveň saturace = 1 pro všechny úlohy
Vyber 2 vzhledem k nejvyššímu stupni

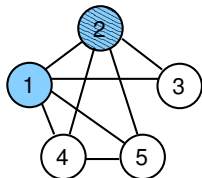


Úroveň saturace = 2 pro všechny uzly
Vyber 4 vzhledem k nejvyššímu stupni

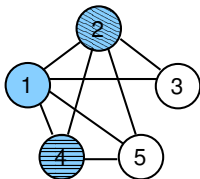
Příklad: plánování schůzek (dokončení)



Úroveň saturace = 1 pro všechny úlohy
Vyber 2 vzhledem k nejvyššímu stupni

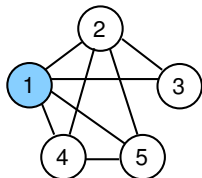


Úroveň saturace = 2 pro všechny uzly
Vyber 4 vzhledem k nejvyššímu stupni

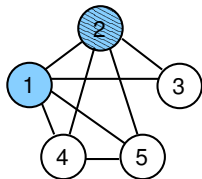


Úroveň saturace = 2 pro uzel 3
Úroveň saturace = 3 pro uzel 5
Vyber 5 na obarvení

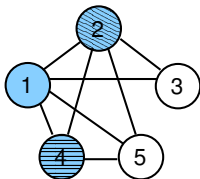
Příklad: plánování schůzek (dokončení)



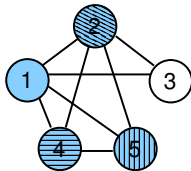
Úroveň saturace = 1 pro všechny úlohy
Vyber 2 vzhledem k nejvyššímu stupni



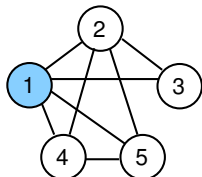
Úroveň saturace = 2 pro všechny uzly
Vyber 4 vzhledem k nejvyššímu stupni



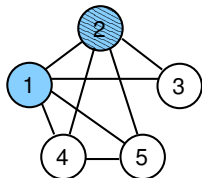
Úroveň saturace = 2 pro uzel 3
Úroveň saturace = 3 pro uzel 5
Vyber 5 na obarvení



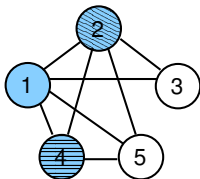
Příklad: plánování schůzek (dokončení)



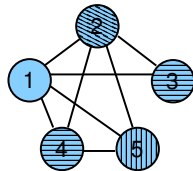
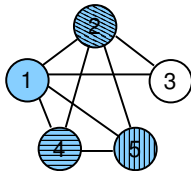
Úroveň saturace = 1 pro všechny úlohy
Vyber 2 vzhledem k nejvyššímu stupni



Úroveň saturace = 2 pro všechny uzly
Vyber 4 vzhledem k nejvyššímu stupni



Úroveň saturace = 2 pro uzel 3
Úroveň saturace = 3 pro uzel 5
Vyber 5 na obarvení



V posledním kroku obarvi 3
stejnou barvou jako 4
⇒ celkem 4 barvy, tj. *makespan*=4

Přiřazení místností

- vrchol: předmět úloha
- hrana: mezi předměty vyžadujícími stejný čas průnik časových bodů
- barva vrcholu: odpovídá vybrané místnosti zdroj
 - sousedící vrcholy/předměty musí mít různé barvy/místnosti, protože vyžadují stejný čas

Rezervační problém

- vrchol: rezervace úloha
- hrana: pokud se dvě rezervace překrývají v čase průnik intervalů
- barva vrcholu: odpovídá vybranému zdroji zdroj
 - sousedící vrcholy/rezervace musí mít různé barvy/zdroje, protože se překrývají v čase

Rozvrhování operátorů

- vrchol: úloha úloha
- hrana: mezi úlohami vyžadujícími stejného operátora průnik zdrojů
- barva vrcholu: čas pro realizaci úlohy časový bod
 - sousedící úlohy/vrcholy musí mít různý čas/barvu, protože vyžadují stejného operátora

Jakou grafovou reprezentaci mají následující problémy? Problémy vyřešte a ukažte postup řešení.

- 1 Určete, ve kterých místnostech se mají konat schůzky tak, aby byla v každé místnosti nejvýše jedna schůzka a přitom byly schůzky organizovány v uvedených termínech.

předmět	A	B	C	D	E
časy	(1,3,5)	(2,4)	(1,2)	(3,4)	(1,5)

Nápověda: problém přiřazení místností

- 2 Stroje v továrně mají být využívány uvedenými operacemi v následujících časových intervalech. Určete, kolik strojů je třeba a které stroje budou využívat jednotlivé operace v případě, že stroj může zpracovávat nejvýše jednu operaci.

operace	A	B	C	D	E	F
interval	1-3	2-4	1-4	4-5	5-8	5-6

Nápověda: rezervační problém

- 3 Určete, kolik času je potřeba pro realizaci operací na uvedených strojích, jestliže může být na každém stroji zpracovávána nejvýše jedna operace.

operace	1	2	3	4	5	6	7
stroje	A,B	C,D	A,C,E	E,F	E,G	D,G	G

Nápověda: rozvrhování operátorů

- 1 Barvení grafu
 - Popis problému a jednoduché řešení
 - Přiřazení místností
 - Rezervační problém
 - Rozvrhování operátorů
- 2 Transport
 - Doba na nastavení
 - Doba na dopravu
- 3 Plánování na počítačové síti
 - Úvod
 - Paralelní úlohy s komunikací

Problém obchodního cestujícího

Problém obchodního cestujícího

- obchodní cestující musí projet všechna města tak, aby
 - celková ujetá vzdálenost (resp. doba cesty) byla minimální a
 - každé město projel právě jednou

Grafová reprezentace

- (orientovaný) hranově ohodnocený graf
- vrchol = město
- (orientovaná) hrana z A do B = přímá cesta z A do B
 - hrany mohou být orientované, pokud chceme uvažovat různou náročnost v opačných směrech cesty
- ohodnocení hrany z A do B = doba nutná na cestu z A do B

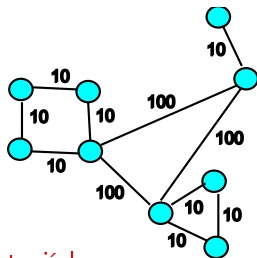
Nastavovací doba a cena (*setup time and cost*)

- Nastavovací doba s_{ijk}, s_{jk} : **závislá na úlohách**
 - udává závislost na posloupnosti provádění úloh
 - s_{ijk} čas nutný pro provádění úlohy k po úloze j na stroji i
 - s_{jk} nastavovací doba nezávislá na stroji
- **Problém obchodního cestujícího** = $1|s_{jk}|C_{\max}$
 - příklad: cesta přes města 12341 odpovídá $s_{12} + s_{23} + s_{34} + s_{41}$
- Klasický příklad: plnění limonád do lahví
 - dána cena za nastavení stroje při změně plnění typu limonády
 - $s_{cola,voda}$ $s_{voda,cola}$ $s_{cola,dzus}$
 - posloupnost plnění: 100 lahví vody, 50 lahví coly, 70 lahví džusu,
- Nastavovací cena c_{ijk}, c_{jk}
 - s přechodem lze spojit i cenu, kterou je nutné zaplatit

Doba na dopravu (*transportation time*)

Multi-operační rozvrhování

- úloha se skládá z několika operací
- může/nemusí být určeno pořadí operací
- operace má zadáno
 - dobu provádění, konkrétní stroj k provádění
- stroj: na každém stroji maximálně jedna operace
- doba na dopravu t_{hl} mezi stroji h a l : **závislá na strojích**
 - kapacita cest mezi stroji neomezená
 - délka cesty mezi stroji = součet odpovídajících dob na dopravu
- cíl: realizovat všechny operace všech úloh
při minimalizaci času dokončení všech úloh



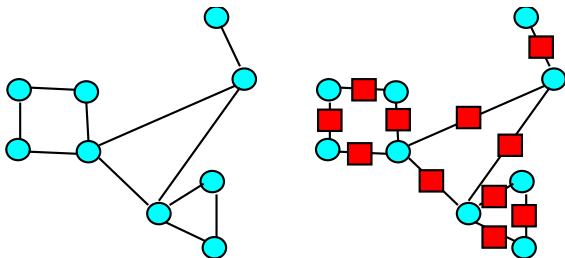
Grafová reprezentace

- **orientovaný hranově ohodnocený graf**
- vrchol: stroj
- hrana: pokud lze přejít přímo z jednoho stroje na druhý
- ohodnocení hrany: doba na dopravu z jednoho stroje na druhý

Plánování na počítačové síti

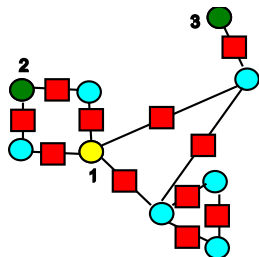
- Stroj: dán počet procesorů
- Úlohy prováděny na jednom uzlu počítačové sítě
 - vyžadují několik procesorů
- Úlohy potřebují k výpočtu **data**
 - data dané velikosti na jednom nebo více uzlech
 - data je nutné přenést na uzel, kde se úloha bude počítat
 - realita: data jsou často zreplikována na několika uzlech
- Linka:
 - **propustnost** = kapacita linky
 - **latence** = doba nutná na přenos dat po lince
- Cíl: **realizovat všechny úlohy tak, jak dynamicky přibývají**
 - úlohy musí mít dostatek procesorů
 - data musí ležet v době výpočtu na uzlu, kde se počítá úloha
 - je nutné plánovat i přenosy dat tak, aby bylo možné data přenést vzhledem k latenci i propustnosti linek na cestě

- Vrcholově ohodnocený neorientovaný graf
- Vrchol: stroj nebo linka
- Ohodnocení vrcholu-stroje: počet procesorů
- Ohodnocení vrcholu-linky: propustnost linky
 - linka je chápána jako zdroj, jehož kapacita odpovídá propustnosti
 - doba trvání úlohy na lince odpovídá latenci
- Hrany: pokud jsou stroje A a B přímo spojeny linkou C, pak existují hrany AC a BC

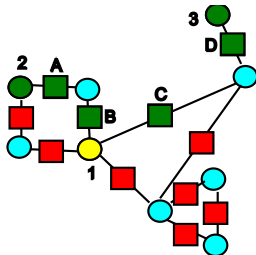


Plánování úlohy na počítačové síti: příklad

- Úloha naplánována k provádění na uzlu 1
- Data na uzlech 2 a 3

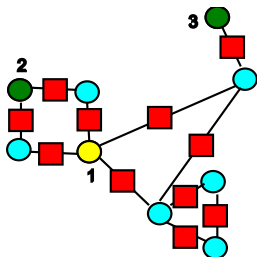


- Data jsou přenesena přes D,C a A,B
- Propustnost/kapacita linky/zdrojů A,B,C,D musí být v daném čase postačující
- Celková doba přenosu do 1:
 $\max(\text{latenceA} + \text{latenceB}, \text{latenceD} + \text{latenceC})$

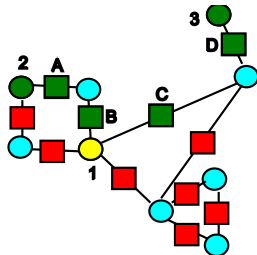


Plánování úlohy na počítačové síti: příklad

- Úloha naplánována k provádění na uzlu 1
- Data na uzlech 2 a 3



- Data jsou přenesena přes D,C a A,B
- Propustnost/kapacita linky/zdrojů A,B,C,D musí být v daném čase postačující
- Celková doba přenosu do 1:
 $\max(\text{latenceA}+\text{latenceB}, \text{latenceD}+\text{latenceC})$



Otázky:

- Je možné takovouto úlohu naplánovat za probíhajícího provozu na síti?
- Je možné ji naplánovat při modifikaci cest pro přenosy?
- Obecně: jak naplánovat úlohu(y) za daného provozu na síť?

směrování

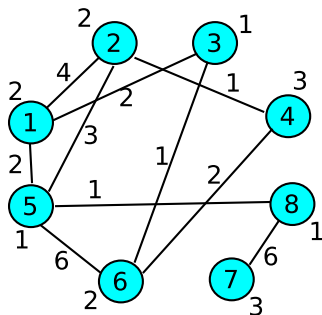
Paralelní komunikující úlohy

Paralelní aplikace

- n komunikujících úloh
- m procesorů
- několik úloh prováděno **zároveň** na každém procesoru

Grafová reprezentace

- **hranově a vrcholově ohodnocený neorientovaný graf**
- ohodnocené vrcholy: úlohy s danou výpočetní náročností
- ohodnocené hrany: **průběžně** komunikující úlohy s komunikační náročností



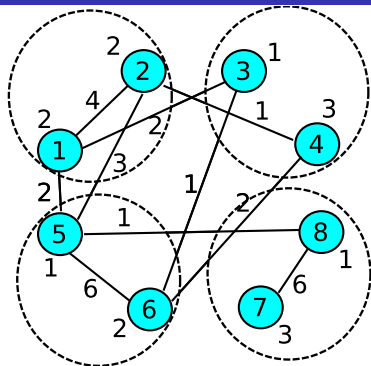
Paralelní komunikující úlohy

Paralelní aplikace

- n komunikujících úloh
- m procesorů
- několik úloh prováděno **zároveň** na každém procesoru

Grafová reprezentace

- **hranově a vrcholově ohodnocený neorientovaný graf**
- ohodnocené vrcholy: úlohy s danou výpočetní náročností
- ohodnocené hrany: **průběžně** komunikující úlohy s komunikační náročností



Vyvažování zátěže (load balancing): přiřazení úloh na procesory tak, aby byla

- vyvážená zátěž jednotlivých procesorů
- minimalizována komunikace úloh na různých procesorech

Formulace problému vyvažování zátěže jako
problému rozdělení grafu (*graph partitioning*)

Rozdělení grafu $G = (V, E)$ na $V = V_1 \cup \dots \cup V_m$ tak, že je

- $V_1 \cap \dots \cap V_m = \emptyset$
- $G_1 = (V_1, E_1), \dots, G_m = (V_m, E_m)$
- E_i tvořeno hranami, jejichž oba vrcholy patří do V_i
- součet ohodnocení vrcholů v jednotlivých V_i „zhruba stejný“
- součet ohodnocení hran $E \setminus \{E_1 \cup \dots \cup E_m\}$ spojujících různé V_j a V_k minimalizován

Speciální případ: $V = V_1 \cup V_2$ bisekce grafu (graph bisection), tj. z grafu $G = (V, E)$ vytvoříme dva podgrafy (V_1, E_1) (V_2, E_2) tak, že

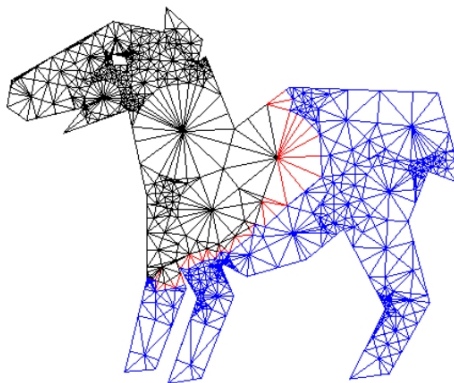
- $V = V_1 \cup V_2, V_1 \cap V_2 = \emptyset$
- E_i tvořeno hranami, jejichž oba vrcholy patří do V_i , tj.
 $E_1, E_2 \subset E, E_1 \cap E_2 = \emptyset, E_i$
- součet ohodnocení vrcholů ve V_1 a V_2 je „zhruba stejný“
- součet ohodnocení hran $E \setminus \{E_1 \cup E_2\}$ spojující vrcholy z V_1 a V_2 je minimalizován

Jak nalézt vhodné rozdělení grafu?

- problém optimálního rozdělení je NP-úplný
 - už pro bisekci: prohledání všech podmnožin množiny vrcholů (podmnožina a její doplněk tvoří V_1 a V_2)
- nutné použít dobré heuristiky

Heuristika: opakovaná bisekce grafu

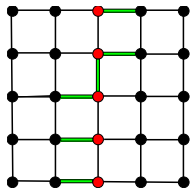
Základní používaný princip při rozdělení grafu:
rozdělení množiny vrcholů V na 2^k částí:
rekurzivní bisekce grafu k -krát



58 dělicích hran

Dělicí hrany (edge separator) vs. dělicí vrcholy (vertex separator)

- **Dělicí hrany:** $E_S \subseteq E$ dělí G po odstranění E_S z E na dvě stejně velké nesouvisející komponenty V : V_1 a V_2
- **Dělicí vrcholy:** $V_S \subseteq V$ dělí G po odstranění V_S a všech jejich hran na dvě stejně velké nesouvisející komponenty V : V_1 a V_2



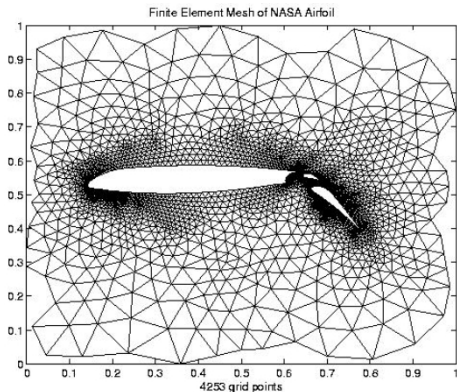
$E_S =$ zelené hrany

$V_S =$ červené vrcholy

Rozdělení se souřadnicemi vrcholů (partitioning with nodal coordinates)

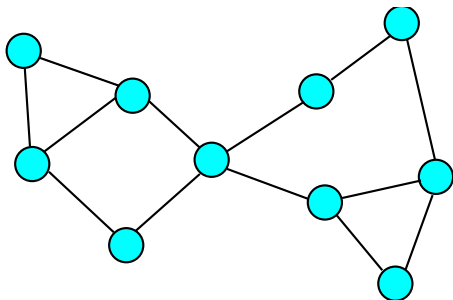
Myšlenka rozdělení pomocí souřadnic vrcholů

- každý vrchol má souřadnice v prostoru → rozdělení prostoru

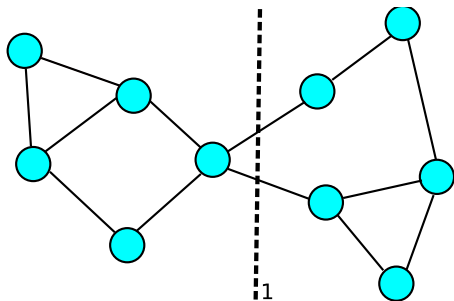


- pomocí dělicí přímky, která dělí vrcholy v prostoru na poloviny

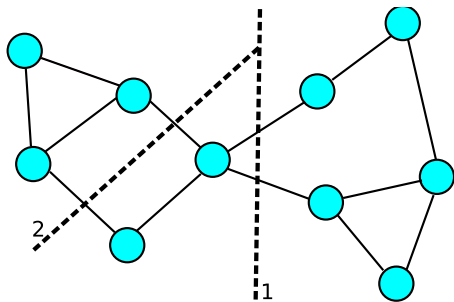
Opakovaná bisekce grafu s dělicí přímkou: příklad



Opakovaná bisekce grafu s dělicí přímkou: příklad



Opakovaná bisekce grafu s dělicí přímkou: příklad



Opakovaná bisekce grafu s dělicí přímkou: příklad

