Sbírka příkladů k předmětu PA167 Rozvrhování

Hana Rudová

Fakulta informatiky, Masarykova univerzita

11. dubna 2022

Obsah

A Úvod do rozvrhování **3**
1 Rozvrh . . . 3
2 Terminologie . . . 3
3 Klasifikace rozvrhovacích problémů . . . 3
4 Složitost . . . 4

B Řídící pravidla **4**
5 Řídící pravidla . . . 4

C Lokální prohledávání **5**
6 Lokální prohledávání obecně . . . 5
7 Tabu prohledávání . . . 6
8 Simulované žíhání . . . 7
9 Genetické algoritmy . . . 7

D Matematické programování **8**
10 Matematické programování . . . 8

E Omezující podmínky **9**
11 Problém splňování podmínek . . . 9
12 Rozvrhování jako problém splňování podmínek . . . 10
13 Podmínky pro zdroje . . . 10
14 Globální omezení . . . 12
15 Rozvrhovací strategie . . . 13

F Plánování projektu **13**
16 Úvod . . . 13
17 Reprezentace projektu . . . 13
18 Neomezené zdroje . . . 13
19 Variabilní doba trvání . . . 14
20 Přidání pracovní síly . . . 14

G Plánování úloh **15**
21 Řídící pravidla . . . 15
22 Metoda větví a mezí . . . 15
23 Paprskové prohledávání . . . 16
24 Matematické programování (jeden stroj) . . . 16
25 Disjunktivní grafová reprezentace . . . 16
26 Matematické programování a job shop . . . 17

27 Metoda větví a mezí pro job shop 17
28 Posunování kritického místa (shifting bottleneck) 17

H Rozvrhování montážních systémů **17**
29 Montážní linka s flexibilním časem 17
30 Montážní linka s fixním časem 18

I Rezervace **19**
31 Úvod 19
32 Intervalové rozvrhování 19
33 Rezervační systém s rezervou 20

J Rozvrhování jako timetabling **20**
34 Úvod 20
35 Rozvrhování s operátory 20
36 Rozvrhování s pracovní silou 21

K Rozvrhování předmětů na univerzitě **21**
37 Popis problému 21
38 Iniciální tvorba rozvrhu 22
39 Interaktivní rozvrhování 22

A Úvod do rozvrhování

1 Rozvrh

1.1. Načrtněte Ganttův diagram pro tabulkou daný rozvrh:

úloha	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
S_j	0	0	1	2	4	4	5	7
p_j	4	2	3	2	4	3	2	1
stroj	M_1	M_2	M_3	M_2	M_2	M_1	M_3	M_1

1.2. Jaké znáte třídy problémů rozvrhování v průmyslu nebo ve službách? Popište stručně charakteristiku tří co nejvíce odlišných problémů.

1.3. Vysvětlete pojmy *rozvrhování* a *rozvrh*.

1.4. Vysvětlete pojmy *částečný rozvrh* a *úplný rozvrh* tak, aby byl zřejmý rozdíl mezi nimi.

1.5. Čím se liší *úplný konzistentní rozvrh* a *optimální rozvrh?*

2 Terminologie

2.1. Jaký je rozdíl mezi pojmy *scheduling (rozvrhování/plánování)* a *timetabling (rozvrhování)*?

2.2. Jaký je rozdíl mezi pojmy *scheduling (rozvrhování/plánování)* a *planning (plánování)*? Co je uvažováno pod pojmem *AI planning (plánování v umělé inteligenci)*?

2.3. Vysvětlete pojmy *sequencing (seřazení)* a *rostering*.

2.4. Vysvětlete termíny *termín dostupnosti (release date)*, *termín dokončení (due date)* a *deadline* tak, aby bylo jasné, čím se liší.

2.5. Čím se liší statické a dynamické parametry úlohy? Jaké znáte základní statické a dynamické parametry úlohy? Popište alespoň tři statické a dva dynamické parametry.

3 Klasifikace rozvrhovacích problémů

3.1. Co je to Grahamova klasifikace? Popište jednotlivé komponenty Grahamovy klasifikace. Uveďte příklad reálného problému s jeho Grahamovou klasifikací a definujte jeho jednotlivé komponenty.

3.2. Grahamova klasifikace umožňuje jako vlastnosti stroje popsat mj. *identické paralelní stroje*, *paralelní stroje s různou rychlostí* a *nezávislé paralelní stroje s různou rychlostí*. Popište tyto vlastnosti strojů a rozdíly mezi nimi.

3.3. Čím se liší *multi-operační (shop)* problémy od problémů na jednodušších paralelních strojích (Pm, Qm, Rm)? Uveďte příklad rozvrhovacího problému znázorněného pomocí Ganttova diagramu, který

- je příkladem Pm a není příkladem multi-operačního problému,
- není příkladem Pm a je příkladem multi-operačního problému.

3.4. Jaké znáte problémy *multi-operačního rozvrhování (shop problems)*? Stručně je popište tak, aby bylo vidět, jaké jsou mezi nimi rozdíly a v čem jsou stejné.

3.5. Popište *flexible flow shop* problém (FFs). Uveďte příklad problému znázorněného Ganttovým diagramem pro 4 úlohy a pro $s = 4$, kde jednotlivé fáze *nejsou* tvořeny jedním strojem. Úlohy mají navíc dobu provádění ve všech fázích nenulovou. Nezapomeňte uvést, které stroje patří jednotlivým fázím.

3.6. Na co se používá *nastavovací (setup) doba*? Jaké znáte typy nastavovací doby podle typu závislosti? Uveďte praktický příklad použití.

3.7. Jaká omezení (komponenta β Grahamovy klasifikace) znáte? Uveďte alespoň tři příklady s jejich stručným popisem.

3.8. Napište dva příklady různých problémů zapsaných pomocí Grahamovy klasifikace tak, aby se jednotlivé komponenty problémů lišily. Uveďte zároveň stručný popis (u kritérii definici) jednotlivých parametrů.

3.9. Formulujte optimalizační kritéria *makespan* (maximální čas konce úloh), *lateness* (zpoždění) a *tardiness* (nezáporné zpoždění). Porovnejte rozdíly mezi nimi.

3.10. Jaká znáte optimalizační kritéria spojená s termínem dokončení? Uveďte dva příklady zahrnující definice kritérií.

3.11. Jakým optimalizačním kritériem lze minimalizovat cenu za skladování při výrobě? Uveďte jeho definici.

3.12. Jaké vlastnosti bude mít rozvrh při volbě robustnosti jako optimalizačního kritéria? Pro jaké problémy může mít taková volba smysl?

4 Složitost

4.1. Uveďte příklady polynomiálního a NP-těžkého rozvrhovacího problému i s jejich Grahamovou klasifikací.

B Řídící pravidla

5 Řídící pravidla

5.1. Co je to řídící pravidlo? Na příkladu s minimálně šesti úlohami demonstrujte užití řídícího pravidla.

5.2. Popište chování čtyř různých řídících pravidel.

5.3. Jaká řídící pravidla byste použili pro minimalizaci *maximálního zpoždění* a *odlišnosti v době čekání na stroj*. Popište je.

5.4. Popište, jakým způsobem se chová řídící pravidlo minimální rezervy a vytvořte rozvrh pro následující úlohy a jeden stroj pomocí tohoto pravidla.

úloha	1	2	3	4	5
p_j	2	3	3	1	3
d_j	7	7	4	11	12

Spočítejte pro Vaše řešení, jaké hodnoty nabývají účelové funkce $L_{\max}$ a $\sum_j T_j$.

5.5. Vytvořte rozvrh pro následující úlohy a jeden stroj pomocí (1) pravidla minimální rezervy (*Minimal Slack – MS*) a (2) pomocí pravidla pravidla nejdřívější termín dokončení (*Earliest Due Date – EDD*). Pozor, pokud pravidlo nevybere jednoznačného kandidáta, tak mezi možnými úlohami vyberte takovou, která má menší identifikátor.

úloha	1	2	3	4	5
r_j	2	3	2	0	3
p_j	5	3	1	1	3
d_j	7	11	4	12	8
w_j	2	1	1	1	1

Při výpočtu dle pravidla EDD vždy uveďte, které úlohy je možné v daném čase rozvrhovat. U výpočtu s MS navíc u těchto úloh uveďte jejich rezervu. Spočítejte pro obě Vaše řešení, jaké hodnoty nabývají účelové funkce $L_{\max}$ a $\sum_j w_j T_j$.

5.6. Jaký je rozdíl mezi statickým a dynamickým řídícím pravidlem? Uveďte definici jednoho statického a jednoho dynamického řídícího pravidla.

5.7. Která řídící pravidla jsou vhodná pro minimalizaci *vážených součtů časů konců úloh* a *makespan*? Popište je.

5.8. Vytvořte rozvrh pro problém $1|r_j|\sum_j C_j$ pomocí pravidla Nejkratší doba provádění (*Shortest Processing Time – SPT*).

úloha	1	2	3	4	5
r_j	15	4	1	0	2
p_j	2	5	1	3	4

Spočítejte pro Vaše řešení hodnotu účelové funkce.

Jak by se výsledný rozvrh lišil, pokud bychom neuvažovali termín dostupnosti r_j a řešili bychom problém $1||\sum_j C_j$ (uveďte výsledný rozvrh a spočítejte pro něj hodnotu účelové funkce)?

5.9. Popište, jakým způsobem se chová řídící pravidlo Vážená nejkratší doba provádění (*Weighted Shortest Processing Time – WSPT*) a vytvořte rozvrh pro problém $1|r_j|\sum_j w_j C_j$

úloha	1	2	3	4	5
r_j	15	4	1	0	2
p_j	2	5	1	3	4
w_j	3	3	2	5	1

Spočítejte pro Vaše řešení hodnotu účelové funkce.

Jak by se výsledný rozvrh lišil, pokud bychom neuvažovali termín dostupnosti r_j a řešili bychom problém $1||\sum w_j C_j$ (uveďte výsledný rozvrh a spočítejte pro něj hodnotu účelové funkce)?

5.10. Popište řídící pravidlo vhodné pro problémy s precedenčními omezeními.

5.11. Popište chování řídícího pravidla *Least Flexible Job First* (LFJ).

5.12. Zhodnoťte výhody a nevýhody řídících pravidel. Kde zejména nacházejí své uplatnění?

C Lokální prohledávání

6 Lokální prohledávání obecně

6.1. Co rozumíte pod pojmem lokální změna? Uveďte příklad pro problém $1|r_j|C_{max}$.

6.2. Popište rozdíl mezi konstruktivními metodami a lokálním prohledáváním.

6.3. Napište základní algoritmus lokálního prohledávání pro problém rozvrhování.

6.4. Jakým způsobem je reprezentován rozvrh pro jeden stroj při použití algoritmů lokálního prohledávání? Uveďte dva příklady, jakým způsobem může být definováno *okolí*. V příkladech uveďte velikost okolí.

6.5. Jaký je rozdíl mezi deterministickými a pravděpodobnostními metodami výběru rozvrhu v algoritmech lokálního prohledávání? Uveďte algoritmus s deterministickou a pravděpodobnostní metodou výběru.

6.6. Uveďte příklady alespoň 2 typů podmínek ukončení používaných v kontextu algoritmů lokálního prohledávání.

6.7. Uvažujte algoritmus lokálního prohledávání pro následující problém minimalizace *makespan* na 1 stroji bez preempce.

úlohy	1	2	3	4	5
r_j	0	4	7	2	7
p_j	3	1	3	1	2

Do okolí zahrňte všechny rozvrhy, které mohou vzniknout párovou výměnou sousedních úloh. Z těch vyberte do další iterace vždy nejlepší rozvrh, a to pouze pokud zlepšuje hodnotu optimalizačního kritéria. Jako iniciální rozvrh zvolte:

(a) 2, 3, 4, 5, 1

(b) 2, 4, 5, 1, 3

Pro obě varianty realizujte tři kompletní iterace algoritmu, pokud výpočet neskončí dříve.

7 Tabu prohledávání

7.1. Co je to tabu seznam v kontextu tabu prohledávání? Jaká délka tabu seznamu se obvykle používá? K čemu může dojít, je-li příliš krátký či dlouhý?

7.2. Jak se používá aspirační kritérium v algoritmech tabu prohledávání?

7.3. Napište algoritmus tabu prohledávání pro řešení obecného rozvrhovacího problému.

7.4. Uvažujte rozvrhovací problém s $1||\sum w_j T_j$

úlohy	1	2	3	4
p_j	1	2	3	1
d_j	2	1	2	10
w_j	1	2	5	3

a tabu prohledávání s následujícími parametry:

- okolí tvoří všechny rozvrhy, které získáme párovou výměnou sousedních úloh;
- z okolí je vybrán nejlepší rozvrh;
- iniciální rozvrh je $(1, 2, 3, 4)$;
- tabu seznam tvoří pár úloh, který byl přehozen při jedné poslední změně.

Ukažte, jakým způsobem probíhá výpočet v prvních dvou iteracích.

7.5. Uvažujte rozvrhovací problém s $1|r_j|C_{\max}$ a tabu prohledávání s uvedenými parametry:

úlohy	1	2	3	4
r_j	2	0	8	4
p_j	3	2	2	2

- okolí tvoří všechny rozvrhy, které získáme párovou výměnou sousedních úloh;

- z okolí je vybrán nejlepší rozvrh;
- iniciální rozvrh je $(1, 2, 3, 4)$;
- tabu seznam tvoří pár úloh, který byl přehozen při jedné poslední změně.

Ukažte, jakým způsobem probíhá výpočet v prvních dvou iteracích.

8 Simulované žíhání

8.1. Používá algoritmus simulovaného žíhání deterministickou nebo pravděpodobní metodu výběru řešení? Jakou konkrétní roli ve výběru hraje výše parametru chlazení?

8.2. Jakým způsobem se v algoritmu simulovaného žíhání mění parametr chlazení? Popište hlavní praktický důsledek těchto změn. Uveďte příklad funkce, která se používá pro změnu parametru chlazení.

8.3. Uveďte a vysvětlete *Metropolisovo kritérium* používané u algoritmu simulovaného žíhání.

8.4. Napište algoritmus simulovaného žíhání.

8.5. Proveďte tři kompletní iterace algoritmu simulovaného žíhání pro problém $1||\sum w_j T_j$:

úlohy	1	2	3	4
p_j	9	9	12	3
d_j	10	8	5	28
w_j	14	12	1	12

Do okolí zahrňte všechny rozvrhy vzniklé párovou výměnou sousedních úloh. Z okolí vyberte vždy úlohu s nejnižší hodnotu účelové funkce. Vyjděte z iniciálního rozvrhu (3,1,4,2). Zvolte $\alpha(t) = 0.9 \cdot t, t_0 = 0.99$. Jako náhodná čísla použijte v řadě 0.02, 0.74, 0.16. Můžete využít znalost $e^{-3/0.891} \doteq 0.034$.

9 Genetické algoritmy

9.1. Porovnejte v čem spočívají rozdíly a společné rysy mezi tabu prohledáváním a genetickými algoritmy.

9.2. Popište význam a princip křížení v kontextu genetických algoritmů. Jaké nevýhody má operátor jednoduchého křížení a jak se jim můžeme vyhnout?

9.3. Uveďte příklad operátoru křížení pro genetické algoritmy a problém rozvrhování na jednom stroji. Na příkladu ukažte, jak Vámi definovaný operátor pracuje.

9.4. Popište algoritmus pro *křížení dané pořádím (order-crossover)*.

9.5. Uvažujte genetický algoritmus řešící problém, jehož výstupem je pořadí zpracování úloh na jednom stroji. Pomocí *křížení daného pořadím (OX)* najděte potomky jedinců 812649735 (rodič 1) a 567912348 (rodič 2). V úvodním kroku do potomka zkopírujte pás alel z pozic 3–7 z 1. rodiče, pro druhého potomka zkopírujte pás stejných alel z 2.rodiče. Uveďte postup řešení.

9.6. Co to je mutace a jak se používá? V čem může být mutace užitečná pro genetický algoritmus? Uveďte tři příklady mutace pro rozvrhovací problém s jedním strojem.

9.7. Popište princip výběru jedince pomocí *ruletového kola* v kontextu genetických algoritmů a na příkladu ukažte, jak funguje.

9.8. V jedné iteraci genetického algoritmu mají jedinci v populaci vhodnost po řadě 5, 1, 6, 8. Spočtěte pravděpodobnosti výběru každého z nich v jednom kroku výběru metodou ruletového kola.

9.9. V kontextu genetických algoritmů popište princip turnajového výběru.

9.10. Jak mohou genetické algoritmy zaručit přežití nejlepšího jedince? Může to mít i nějaké nevýhody?

9.11. Napište genetický algoritmus pro konstrukci rozvrhu.

9.12. Uvažujte rozvrhovací problém s $1||\sum T_j$ a pomocí genetického algoritmu nalezněte řešení po provedení tří iterací algoritmu s uvedenými parametry:

úlohy	1	2	3	4
p_j	3	6	5	2
d_j	4	8	7	15

- velikost populace je 3;
- iniciální populace je následující 3421, 2314, 4231;
- pro reprodukci je použita párová výměna sousedů;
- ze všech možných potomků (pozor, každý jedinec se může reprodukovat) je vybrán nejvhodnější a nahradí nejméně vhodného jedince z předchozí generace;

Jako součást řešení uveďte v každé iteraci všechny možné potomky a jejich vhodnost.

9.13. Uvažujte montážní linku produkující 5 různých výrobků. Pomocí genetického algoritmu nalezněte optimální pořadí produkce těchto výrobků minimalizující celkovou nastavovací dobu pro jeden výrobní cyklus (účelová funkce zahrnuje i cenu za nastavení z posledního na první výrobek). Nastavovací doby jsou následující (př. cena za nastavení z úlohy 2 na úlohu 1 je 30):

úlohy	1	2	3	4	5
1	–	2	8	4	6
2	30	–	6	40	20
3	4	30	–	50	30
4	20	18	12	–	24
5	40	20	30	6	—

Rozvrh reprezentujte jako posloupnost čísel výrobků. Použijte populaci o 3 jednotlivcích, začněte s iniciální populací 32415, 42351, 25134. V každé populaci se reprodukuje jedinec vybraný turnajovým výběrem pomocí párové výměny sousedních prvků (poslední prvek s prvním se nemění). Mezi možnými párovými výměnami vyberte jako potomka nejlepšího jedince a nahraďte jím nejhoršího jedince populace. Proveďte kompletní výpočet dvou iterací algoritmu (turnajový výběr vybere v první iteraci první dva jedince, ve druhé iteraci poslední dva jedince).

D Matematické programování

10 Matematické programování

10.1. Popište lineární program, včetně oboru proměnných a koeficientů. Čím se od něj liší celočíselný program či mixed-integer program? Které z těchto druhů matematických programů jsou užitečnější pro rozvrhování a proč?

10.2. Jakými metodami se řeší lineární programy (stačí název)?

10.3. Stručně popište metodu řezné roviny a metodu větví a mezí pro řešení celočíselných programů. Co předchází aplikaci těchto metod?

10.4. Popište problém rozvrhování směn a formulujte ho jako problém celočíselného programování.

10.5. Pracovní doba v továrně je od 6:00 do 22:00. Zaměstnanci mohou pracovat v následujících směnách a za každou z nich jsou placeni uvedeným platem

směna	doba	plat
1	6-14	750
2	14-22	850
3	6-18	1200
4	8-14	550
5	6-10	370

V době od 6:00 do 8:00 jsou potřeba 4 zaměstnanci, mezi 8:00 a 14:00 je vyžadováno 8 zaměstnanců, od 14:00 do 18:00 celkem 5 zaměstanců a večer od 18:00 do 22:00 jsou nutni pouze 3 zaměstnanci. Naformulujte jako *problém celočíselného programování*, jakým způsobem je možné nalézt takový počet zaměstnanců na každou směnu, aby byla minimalizována celková výše peněz nutná na platy.

10.6. Naformulujte následující problém jako *problém celočíselného programování.*
Na lince pracují zaměstnanci od 6:00 do 20:00. Každý ze zaměstnanců může pracovat na ranní směně (6:00-14:00), na krátké polední směně (10:00-14:00), na dlouhé polední směně (10:00-16:00) a nebo na večerní směně (14:00-20:00). Zaměstnavatel jim platí za ranní směnu 800 Kč, za krátkou polední 450 Kč, za dlouhou polední 650 Kč a za večerní směnu 700 Kč. V době od 6:00 do 10:00 je potřeba minimálně 20 zaměstnanců, od 10:00 do 14:00 30 zaměstnanců, od 14:00 do 16:00 25 zaměstnanců a od 16:00 do 20:00 je třeba minimálně 22 zaměstnanců. Jaký počet zaměstnanců je potřeba obsadit na každé směně, aby byla minimalizována celková cena za zaměstance za den?

E Omezující podmínky

11 Problém splňování podmínek

11.1. V kontextu omezujících podmínek definujte *množinu doménových proměnných* a *doménu*.

11.2. Definujte pojem *omezení* v kontextu programování s omezujícími podmínkami. Co musí platit, aby se omezení nazývalo *splněné*? Uveďte příklad omezení a ukažte, kdy je splněno a kdy není splněno.

11.3. Popište *problém splňování podmínek*. Jak je definováno jeho *řešení*? Ukažte příklad problému splňování podmínek, který má právě dvě řešení.

11.4. Uveďte konkrétní příklad *problému splňování podmínek* a na něm ukažte pojmy *množina doménových proměnných*, *doména*, *omezení* a *řešení problému splňování podmínek*. Příklad problému splňování podmínek navrhněte tak, aby měl alespoň pět proměnných a pět omezení.

11.5. Co je to filtrace domén? K čemu slouží při řešení problému splňování podmínek? Uveďte příklad filtrace domény.

11.6. Co je to hranová konzistence (AC)? Kdy nazýváme problém splňování podmínek (CSP) hranově konzistentní? Ukažte příklad problému, který je hranově konzistentní a příklad problému, který není hranově konzistentní.

11.7. Jak se v problému splňování podmínek zajišťuje hranová konzistence? Napište algoritmus AC-8 pro zajištění hranové konzistence. Čím se liší od algoritmu AC-3?

11.8. Použijte algoritmus AC-8 pro zajištění hranové konzistence na problém splňování podmínek:

- $A, B, C \in \{1, 2, 3, 4\}$
- $c1 : A \neq B, \quad c2 : B \neq C, \quad c3 : A + B = 4, \quad c4 : B + C = 3$

V algoritmu použijte uspořádání proměnných A, B, C při zařazování proměnných do fronty a uspořádání omezení $c1, c2, c3, c4$ při výběru k revizi. Ve svém řešení uveďte vždy stav fronty, všechny provedené revize omezení (i když nedojde ke změně domén) a domény proměnných, jejichž doména se změnila.

11.9. Uveďte kostru algoritmus přiřazování (labeling) používaný při řešení problému splňování podmínek. Jaký typ prohledávání se zde používá?

11.10. Proč není pro řešení problému splňování podmínek dostačující algoritmus pro hranovou konzistenci a musí být doplněn prohledávacím algoritmem?

11.11. Popište techniku pohledu dopředu (MAC) pro prohledávání při řešení problému splňování podmínek.

11.12. Navrhněte a napište modifikaci techniky pohledu dopředu (MAC) tak, aby používala větvení stavového prostoru typu (Promenna=Hodnota $\vee$ Promenna$\neq$Hodnota).

11.13. Použijte algoritmus MAC (technika pohledu dopředu) na problém splňování podmínek:

- $A \in \{0, 1\}, B \in \{0, 1, 2, 3\}, C \in \{0, 1, 2\}$
- $A \neq B, A \neq C, A < B, A + B + C = 4$
- při výběru proměnné preferujte tu, která má nejmenší doménu
- preferujte hodnoty, které jsou více podporovány v součtu přes všechny podmínky, při stejném součtu preferujte menší hodnoty v doméně

12 Rozvrhování jako problém splňování podmínek

12.1. Popište, jaké doménové proměnné se používají k popisu rozvrhování jako problému splňování podmínek (včetně uvedení domén jednotlivých typů proměnných).

12.2. Jak se liší omezení pro začátek, dobu trvání a konec úlohy, je-li preempce (přerušení aktivity) *zakázána* nebo *povolena*?

12.3. Pomocí omezujících podmínek vyjádřete

(a) precedenční omezení mezi úlohami A, B,

(b) disjunktivní omezení (tj. nepřekrývání) těchto úloh.

V obou případech předpokládejte, že úlohy jsou nepřerušitelné.

13 Podmínky pro zdroje

13.1. Vysvětlete pojmy *unární zdroj*, *kumulativní zdroj* a *produkovatelný/spotřebovatelný* zdroj. Ke každému typu zdroje uveďte příklad reálného zdroje, který je vhodné tímto typem modelovat.

13.2. Popište některé z odvozovacích pravidel pro zdroje používané v kontextu programování s omezujícími podmínkami a uveďte jakým způsobem pro něj probíhá filtrace domén.

13.3. Ukažte na příkladu, kdy a jak je možné aplikovat algoritmus *hledání hran (edge finding)*.

13.4. Jaká jsou odvozovací pravidla pro algoritmus *hledání hran (edge finding)*? Demonstrujte na příkladu.

13.5. Ukažte na příkladu, kdy a jak je možné aplikovat algoritmus *Ne-první/ne-poslední* (*not-first/not-last*).

13.6. Jaká jsou odvozovací pravidla pro algoritmus *Ne-první/ne-poslední (not-first/not-last)*? Demonstrujte na příkladu.

13.7. Popište principy hledání hran a metody *Ne-první/ne-poslední* tak, aby bylo jasné v čem se tyto metody liší a co mají společného.

13.8. Napište odvozovací pravidla metod *hledání hran* a *Ne-první/ne-poslední* pro úlohy sdílející jeden zdroj o kapacitě 1. Lze pomocí nich vytvořit úplný rozvrh pro uvedený problém v případě, že r_j je nulový pro všechny úlohy? Uveďte postup výpočtu pro každou metodu zvlášť. Pokud existuje po aplikaci jednotlivých metod více možných řešení, tak uveďte, která to jsou.

úloha	1	2	3	4
p_j	6	3	2	4
d_j	8	13	16	10

13.9. Máme dánu množinu úloh $\Omega = \{X, Y, Z, V\}$.

úloha	X	Y	Z	V
p_j	3	3	4	2
r_j	0	2	3	1
d_j	10	10	12	9

Uvažujme úlohy $A \in \Omega$. Uveďte všechny dvojice typu $(A, \Omega - \{A\})$, pro které metoda *hledání hran* umožní redukci domény proměnných. Pro případy, kdy dojde k redukci domén, zároveň formulujte odvozovací pravidla pro příslušnou dvojici (úloha,množina) a dosaďte do nich následně odpovídající numerické hodnoty, aby bylo zřejmé, jakým způsobem výpočet pomocí pravidel funguje.

Totéž realizujte pro metodu *Ne-první/ne-poslední*.

13.10. Vysvětlete pojem *optimistický zdrojový profil* v kontextu programování s omezujícími podmínkami a uveďte vztah pro jeho výpočet.

13.11. Vysvětlete pojem *pesimistický zdrojový profil* v kontextu programování s omezujícími podmínkami a uveďte vztah pro jeho výpočet.

13.12. Srovnejte pojmy *optimistický zdrojový profil* a *pesimistický zdrojový profil* v kontextu programování s omezujícími podmínkami.

13.13. Vypočtěte hodnotu *optimistického zdrojového profilu* pro úlohu C a je-li to možné, rozhodněte, může-li být naplánována po A. Počáteční (maximální) kapacita zdroje je 4 a minimální 0. Je dána precedence $C << D$.

úloha	A	B	C	D	E
cap_j	-3	1	-4	3	1

13.14. Vypočtěte hodnotu *pesimistického zdrojového profilu* pro úlohu C a je-li to možné, rozhodněte, může-li být naplánována po A. Počáteční (minimální) kapacita zdroje je 0 a maximální 5. Je dána precedence $C << D$.

úloha	A	B	C	D	E
cap_j	4	1	4	-5	1

13.15. Jaká technika se používá pro modelování alternativních zdrojů pro aktivitu? Popište související odvozovací pravidla.

13.16. Jak je možné modelovat zdroj s kapacitou proměnnou v čase s pomocí zdroje konstantní kapacity?

13.17. Definujte tabulku (*timetable*) pro aktivitu A a podmínku tabulky (*timetable constraint*). K čemu tato podmínka slouží?

13.18. Popište odvozovací pravidla pro podmínku tabulky (*timetable constraint*).

13.19. Odvoďte rozvrh pomocí odvozovacích pravidel *podmínky tabulky* (*timetable constraint*) pro následující úlohy

úloha	1	2	3	4
r_j	2	0	2	0
$deadline_j$	10	5	6	9
p_j	4	3	3	8

sdílející zdroj o kapacitě 2, kde každá úloha vyžaduje kapacitu zdroje 1. Popište jednotlivé kroky odvození včetně inicializace a změn hodnot proměnných $X(j,t)$ pro úlohy j v čase t. Pokud existuje více možných řešení, tak uveďte, která to jsou.

14 Globální omezení

14.1. Uveďte tři různé typy globálních podmínek používaných pro rozvrhování. Pro každou z těchto podmínek ukažte na příkladu způsob jejich použití.

14.2. Popište syntaxi a sémantiku globální omezující podmínky `allDifferent`. Ukažte dále příklad použití této podmínky.

14.3. K čemu se používají *intervalové a sekvenční proměnné*? Jaký je mezi nimi rozdíl? Ukažte jejich použití na příkladu.

14.4. Popište syntaxi a sémantiku globální omezující podmínky `noOverlap` využívanou pro reprezentaci *unárního zdroje*. Ukažte dále příklad použití této podmínky.

14.5. Popište syntaxi a sémantiku kumultivní fukce `cumulFunction`. Ukažte dále příklad použití této podmínky.

14.6. Vyjádřete globální omezující podmínkou, že státní zkouška každého studenta musí začínat v jiný čas (reprezentovaný hodinou). Doba trvání každé státní zkoušky je jedna hodina. Intervaly možných začátků jsou:

student	1	2	3	4	5
od	8	8	9	10	9
do	10	12	11	12	12

14.7. Zapište pomocí globálních omezujících podmínek problém rozvržení následujících úloh sdílejících zdroj o kapacitě 3.

úloha	1	2	3	4	5
p_j	2	3	1	2	5
cap_j	2	3	1	1	2

14.8. Vyjádřete globální omezující podmínkou problém rozvrhování výpočetních úloh na 4 stroje, kdy každá úloha běží bez přerušení na vybraných strojích.

úloha	1	2	3	4	5
p_j	5	3	1	3	2
cap_j	1	2	4	3	2

15 Rozvrhovací strategie

15.1. Popište způsoby větvení prohledávacího algoritmu pro řešení problému splňování podmínek. Co jsou to *princip prvotního neúspěchu (first-fail)* a *princip prvotního úspěchu (succeed-first)*? Který z nich se obvykle používá pro výběr proměnné k přiřazení a který pro výběr hodnoty?

15.2. Popište mechanismus větvení v technikách prohledávání a přiřazování v kontextu programování s omezujícími podmínkami. Jaký mají při větvení význam rozvrhovací strategie?

15.3. K čemu se používá *rezerva* při rozvrhování s omezujícími podmínkami? Definujte pojem rezervy dvou aktivit v kontextu programování s omezujícími podmínkami.

15.4. Popište využití rezervy dvou aktivit v rozvrhovacích strategiích. Jaký pár aktivit je uspořádán první, jaké pořadí bývá zvoleno, a proč?

15.5. Popište strategii větvení *první/poslední* v kontextu programování s omezujícícmi podmínkami.

15.6. Definujte pojem *zdrojová rezerva* v kontextu programování s omezujícími podmínkami (včetně uvedení matematického vztahu, jak se tato rezerva spočítá pro danou množinu aktivit na zdroji). Jak jí lze využít v rozvrhovacích strategiích?

F Plánování projektu

16 Úvod

16.1. Popište základní problém plánování projektu. Jak se rozšiřuje o *variabilní dobu trvání* a *pracovní sílu*? Uveďte konkrétní příklady takových problémů.

17 Reprezentace projektu

17.1. Různé způsoby reprezentace projektu znázorňují úlohu jako *uzel* nebo *obdélník*. Popište tyto způsoby reprezentace a ukažte je na příkladě jednoduchého projektu.

18 Neomezené zdroje

18.1. Popište problém plánování projektu s neomezenými zdroji jako problém plánování úloh. Uveďte Grahamovu klasifikaci problému.

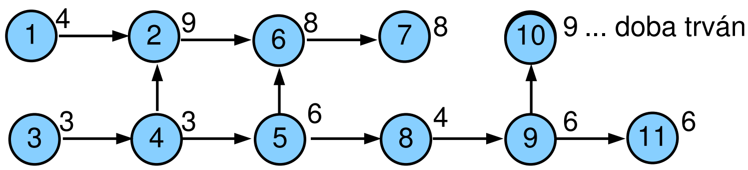
18.2. Co jsou to v kontextu metody kritické cesty *úloha s rezervou* a *kritická úloha*?

18.3. Napište algoritmus pro nalezení kritické cesty. Jako jeho součást uveďte vztahy pro výpočet nejdřívějšího startovního a koncového času a vztahy pro výpočet nejpozdějšího koncového a startovního času.

18.4. Pro následující problém plánování projektu uveďte

- graf kritických cest,

- úlohy s rezervou (slack jobs) a jejich rezervu,
- nejpozdější startovní čas všech úloh.



19 Variabilní doba trvání

19.1. Definujte pojem *marginální cena* v kontextu problému plánování projektu.

19.2. Jakým způsobem je definovaná *cena za provádění projektu* používaná při řešení problému plánování projektu s variabilní dobou trvání (je nutné nejen slovně popsat jednotlivé parametry, ale zároveň je i definovat)?

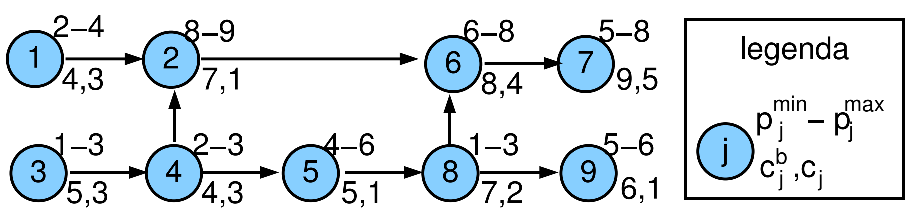
19.3. Co je to (minimální) vrcholový řez grafu?

19.4. Napište algoritmus *kompromisní heuristiky mezi časem a cenou* pro problém plánování projektu s variabilní dobou trvání.

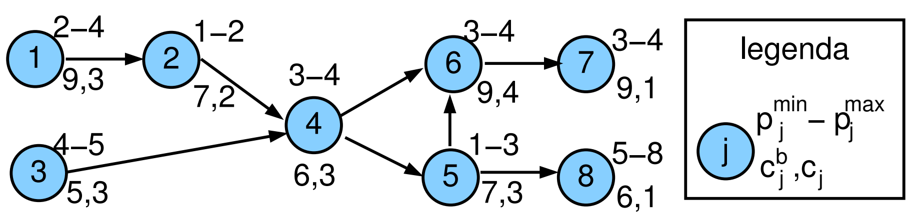
19.5. Pro následující vstup realizujte jeden kompletní krok algorimu kompromisní heuristiky mezi časem a cenou pro problém plánování projektu. V řešení nezapomeňte uvést iniciální kritické cesty, všechny minimální řezy a nové kritické cesty po proběhnutí kroku algoritmu.

Jaká bude iniciální cena a cena po realizaci jednoho kroku algoritmu za předpokladu, že fixní režijní náklady jsou i) $c_0 = 4$, ii) $c_0 = 5$

i)



ii)



Víte, za jakých podmínek algoritmus kompromisní heuristiky skončí?

19.6. Napište formulaci lineárního programu pro řešení problému plánování projektu s variabilní dobou trvání. Uveďte popis proměnných, optimalizační funkci a omezení.

20 Přidání pracovní síly

20.1. Formulujte problém plánování projektu s omezenými zdroji *(resource-constrained project scheduling problem, RCPSP)*.

20.2. Napište formulaci celočíselného programování pro problém plánování projektu s přidáním pracovní síly. Uveďte popis proměnných, optimalizační funkci a omezení.

G Plánování úloh

21 Řídící pravidla

21.1. Která řídící pravidla se používají pro řešení problémů plánování úloh? Dokážete uvést tři reprezentanty těchto problémů (zapište je v Grahamově klasifikaci) spolu s doporučeným řídícím pravidlem?

21.2. Jaký je vztah mezi řídícími pravidly *nejdřívější termín dokončení* (EDD) a *minimální rezerva* (MS)?

21.3. Jaká je složitost problému $Pm||C_{\max}$? Lze jej efektivně řešit použitím řídícího pravidla?

21.4. Jakým řídícím pravidlem je možné řešit problém $Pm||\sum C_j$ pro $r_j = 0$? Liší se nějak řešení pro preemptivní variantu problému? Je řešení problému $Pm||\sum C_j$ stejně obtížné jako řešení problému $Pm||\sum w_j C_j$ (uveďte složitost řešení obou problémů)? Jaké řídící pravidlo byste použili pro problém $Pm||\sum w_j C_j$?

21.5. Dokažte, že řídící pravidlo *vážená nejkratší doba trvání* (weighted shortest processing time, WSPT) je optimální pro problém $1||\sum w_j C_j$ (tj. $r_j = 0, d_j = \infty$).

21.6. Porovnejte složitost řešení problémů $1|r_j|L_{\max}$ a $1||L_{\max}$ (uveďte zároveň, zda a v čem je mezi problémy rozdíl v obtížnosti řešení). Jaké metody řešení se pro tyto problémy používají? Dávají některé metody optimální řešení?

21.7. Porovnejte problémy $1|r_j|L_{\max}$ a $1|r_j, prmp|L_{\max}$. Který z těchto problému je složitější a proč? Lze některý z nich optimálně řešit řídícími pravidly?

21.8. Vytvořte rozvrh podle pravidla EDD (Earliest Due Date, nejdřívější termín dokončení) pro problém $1|r_j|L_{\max}$ a podle pravidla preemptivní EDD pro problém $1|r_j, prmp|L_{\max}$, kde

úloha	1	2	3	4	5
p_j	2	2	3	1	3
r_j	5	4	0	4	8
d_j	7	7	4	11	12

Jaká je hodnota účelové funkce pro jednotlivé problémy?

22 Metoda větví a mezí

22.1. Popište princip metody větví a mezí. Víte, jak lze v této metodě využít hranice ceny řešení relaxovaného/zjednodušeného problému? Uveďte příklad problému a odpovídajícího relaxovaného problému.

22.2. Který problém lze použít jako relaxaci problému $1|r_j|L_{\max}$ při jeho řešení metodou větví a mezí? Proč je v tomto konkrétním příkladě řešení relaxovaného problému jednodušší než řešení problému původního?

22.3. Popište postup větvení při aplikaci metody větví a mezí na řešení problému $1|r_j|L_{\max}$. V rámci popisu uveďte, která větvení musíme na dané úrovni uvažovat.

22.4. Popište princip práce s mezemi při aplikaci metody větví a mezí na řešení problému $1|r_j|L_{\max}$ (jak se meze počítají a jak se využívají).

22.5. Pomocí metody větví a mezí (branch & bound) nalezněte optimální řešení zadaného problému typu $1|r_j|L_{\max}$. V řešení uveďte, jakým způsobem vypadá prozkoumávaný graf prohledávacího prostoru.

úloha	1	2	3
p_j	4	2	6
r_j	0	1	3
d_j	8	11	10

23 Paprskové prohledávání

23.1. Popište princip paprskového prohledávání (*beam search*). Na příkladu ukažte, jak tento algoritmus funguje. Od kterého algoritmu je paprskové prohledávání odvozeno a v čem se od něj liší?

23.2. Popište a srovnejte metodu větví a mezí (*branch&bound*) s paprskovým prohledáváním (*beam search*). Uveďte příklady problémů, které lze pomocí těchto metod řešit.

23.3. K čemu v metodě paprskového prohledávání slouží parametr *šířka paprsku* (beam width)? Jak ovlivňuje přesnost a rychlost výpočtu?

23.4. Popište princip *dvoufázové evaluační procedury* pro metodu paprskového prohledávání. Čím je tato procedura výhodná?

24 Matematické programování (jeden stroj)

24.1. Formulujte problém $1|r_j, p_j = 1|\sum w_j C_j$ pomocí celočíselného programování (tj. uveďte popis proměnných, optimalizační funkci a předpoklady).

24.2. Formulujte problém $1|deadline_j, prec|\sum w_j C_j$ pomocí celočíselného programování (tj. uveďte popis proměnných, optimalizační funkci a předpoklady).

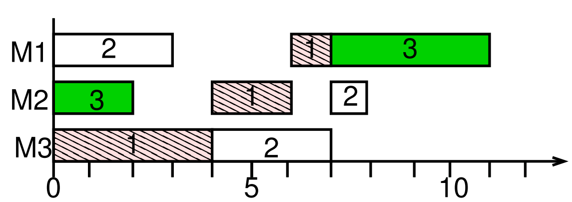
25 Disjunktivní grafová reprezentace

25.1. Co to je disjunktivní grafová reprezentace (uveďte definici)? Ukažte na netriviálním problému (alespoň 3 stroje, 3 úlohy a 8 operací) jeho disjunktivní grafovou reprezentaci.

25.2. Co to je disjunktivní grafová reprezentace a splnitelný výběr (pro získání plného počtu bodů uveďte definici)? Zkonstruujte disjunktivní grafovou reprezentaci, splnitelný výběr a rozvrh pro následující *job shop* problém tak, aby platilo $S_{31} = 6, S_{14} = 1, S_{23} = 3$ a startovní časy ostatních operací byly co nejmenší.

- $J1: (2,1) \to (1,1) \to (3,1)$ $p_{21} = 2, p_{11} = 3, p_{31} = 1,$
- $J2: (4,2) \to (1,2)$ $p_{42} = 2, p_{12} = 1,$
- $J3: (4,3) \to (2,3)$ $p_{43} = 3, p_{23} = 2,$
- $J4: (2,4) \to (1,4) \to (3,4)$ $p_{24} = 1, p_{14} = 1, p_{34} = 4$

25.3. Uveďte disjunktivní grafovou reprezentaci a splnitelný výběr pro následující problém. Vysvětlete čím se disjunktivní grafová reprezentace a splnitelný výběr liší. Napište také úplné zadání tohoto problému včetně všech jeho parametrů a jejich klasického značení.



25.4. Popište algoritmus výpočtu rozvrhu pro job shop problém ze splnitelného výběru jeho disjunktivní grafové reprezentace.

25.5. Napište postup nalezení splnitelného výběru disjunktivní grafové reprezentace z rozvrhu pro job shop problém.

26 Matematické programování a job shop

26.1. Popište, jak jsou formulovány problémy, které řešíme pomocí disjunktivního programování.

26.2. Popište *job shop* problém typu $J_m||C_{\max}$ a uveďte jeho formulaci pomocí disjunktivního programování (tj. uveďte formulaci matematického programování určením proměnných, optimalizačního kritéria a předpokladů). Co reprezentuje disjunkce v disjunktivním programování?

27 Metoda větví a mezí pro job shop

27.1. Jaké vlastnosti má rozvrh, pokud je

   (a) bez zdržení,
   (b) aktivní?

   Uveďte příklad neaktivního rozvrhu a upravte ho tak, aby se z něj stal aktivní rozvrh. Podobně uveďte příklad rozvrhu se zdržením a upravte ho tak, aby se z něj stal rozvrh bez zdržení.

27.2. Napište algoritmus pro *generování všech aktivních rozvrhů* pro problémy typu job-shop.

27.3. Pro zadaný job-shop problém ukažte jak bude inicializován algoritmus *generování všech aktivních rozvrhů* a proveďte dva kroky tohoto algoritmu.

   - úlohy:
     - J_1 $(2,1) \rightarrow (3,1)$
     - J_2 $(1,2) \rightarrow (2,2) \rightarrow (3,2)$
     - J_3 $(3,3) \rightarrow (1,3) \rightarrow (2,3)$
   - doby provádění:
     - $p_{21} = 4, p_{31} = 3, p_{12} = 2, p_{22} = 1, p_{32} = 3, p_{33} = 3, p_{13} = 2, p_{23} = 2$

27.4. Jakým způsobem vypočítáme dolní hranici při řešení job shop problému metodou větví a mezí? V rámci odpovědi diskutujte, které podproblémy používáme pro stanovení dolní meze, a jak je jejich řešení (hodnota účelové funkce) využito při definici dolní hranice.

28 Posunování kritického místa (shifting bottleneck)

28.1. Stručně popište základní myšlenky heuristiky posunování kritického místa.

H Rozvrhování montážních systémů

29 Montážní linka s flexibilním časem

29.1. Popište rozdíly mezi problémy typu job shop a montážní linka (*flexible assembly system*). Jak se liší problémy montážní linky s flexibilním časem, fixním časem a s paralelními pracovními stanicemi?

29.2. Popište problém montážní linky s flexibilním časem a uveďte reálný příklad.

29.3. Vysvětlete pojem blokování v kontextu montážní linky s flexibilním časem. Jak se v tomto problému pracuje s frontami mezi stroji?

29.4. Co je to cyklický rozvrh pro problém montážní linky? Zhodnoťte vliv nastavovacích dob na cyklické rozvrhy.

29.5. Co to je minimální podílová množina (*minimum part set*)? Uveďte netriviální příklad minimální podílové množiny. Jaké znáte různé typy problémů, při jejichž řešení se minimální podílová množina používá?

29.6. Co to je profil úlohy v rámci heuristiky padnoucího profilu pro problém montážní linky s flexibilním časem? Ukažte příklad profilu úlohy pro první dvě úlohy na třech strojích.

29.7. Popište heuristiku padnoucího profilu pro problém montážní linky s flexibilním časem.

29.8. Pomocí *heuristiky padnoucího profilu* vypočtěte cyklický rozvrh pro montážní linku s flexibilním časem. Vypočítaný rozvrh zopakujte ve dvou cyklech, na tomto rozvrhu ukažte, kdy začíná a končí jeden cyklus, a uveďte jeho dobu.

j	N_j	p_{1j}	p_{2j}	p_{3j}
1	1	1	1	1
2	1	4	0	3
3	1	0	0	4
4	1	1	2	0

Jako první rozvrhujte úlohu číslo 1. Pokud by byla neproduktivní doba úloh stejná, vyberte mezi nimi tu, která má nižší index.

Ve svém řešení vždy uveďte pro každého kandidáta na danou pozici odpovídající rozvrh, neproduktivní dobu na jednotlivých strojích a celkovou neproduktivní dobu.

30 Montážní linka s fixním časem

30.1. Popište problém montážní linky s fixním časem a uveďte příklad takového problému. Čím se liší od problému montážní linky s flexibilním časem?

30.2. Co jsou to *skupiny výrobků* v kontextu montážní linky s fixním časem?

30.3. Jak jsou v problému montážní linky s fixním časem definovány operace omezující kapacitu (kritické operace)?

30.4. Napište heuristický algoritmus seskupování a distribuce (grouping and spacing) pro řešení problému montážní linky s fixním časem.

30.5. Je zadán následující problém montážní linky s fixním časem:

(a) doba trvání všech úloh je jednotková

(b) úlohy mají zadán atribut a_j, termín dokončení d_j a váhu w_j

(c) pokud je úloha dokončena po termínu dokončení, pak je penalizována w_jT_j

(d) pokud mají dvě po sobě prováděné úlohy k a l odlišný atribut, pak je nutná cena za nastavení $|a_k - a_l|$.

Úlohy	1	2	3	4	5	6
a_j	2	2	5	5	9	9
d_j	∞	4	∞	∞	3	∞
w_j	0	3	0	0	3	0

Nalezněte posloupnost úloh s minimální cenou pomocí heuristiky seskupování a distribuce (grouping and spacing). Jaká je cena řešení?

30.6. Je zadán následující problém montážní linky s fixním časem:

(a) doba trvání všech úloh je jednotková
(b) úlohy mají zadán atributy a_j a b_j, termín dokončení d_j a váhu w_j
(c) pokud je úloha dokončena po termínu dokončení, pak je penalizována $w_j T_j$
(d) pokud mají dvě po sobě prováděné úlohy k a l odlišný atribut, pak je nutná cena za nastavení $|a_k - a_l|$.
(e) pokud $b_j = b_k = 1$ a mezi j a k je $l - 1$ úloh, pak je nutná penalizace $\psi(l) = \max(3 - l, 0)$

Úlohy	1	2	3	4	5	6	7	8
a_j	2	2	2	5	5	5	8	8
b_j	0	1	0	1	0	1	0	1
d_j	∞	∞	4	∞	∞	∞	2	∞
w_j	0	0	5	0	0	0	5	0

Nalezněte posloupnost úloh s minimální cenou pomocí heuristiky seskupování a distribuce (grouping and spacing). Uveďte popis funkce pro výpočet ceny řešení a vypočítejte, jaká je cena řešení pro uvedený příklad.

I Rezervace

31 Úvod

31.1. Popište problém rezervačního systému, cíl jeho řešení a obvykle používaná optimalizační kritéria. Uveďte příklad aplikace rezervačních systémů.

31.2. Čím se liší rezervační systémy *bez rezervy* a *s rezervou*? Které účelové funkce jsou pro tyto problémy charakteristické (v čem je hlavní rozdíl od účelových funkcí uvažovaných pro plánování strojů)?

32 Intervalové rozvrhování

32.1. Popište problém intervalového rozvrhování a napište optimální algoritmus pro řešení instance tohoto problému, který nalezne minimální počet identických strojů potřebných pro naplánování všech úloh s jednotkovou váhou.

32.2. Napište formulaci celočíselného programování pro problém intervalového rozvrhování.

32.3. Jak se změní problém intervalového rozvrhování, jeho složitost a formulace celočíselného programování, mají-li všechny úlohy jednotkovou dobu trvání?

32.4. Napište algoritmus pro řešení problému intervalového rozvrhování s jednotkovou vahou úloh a identickými stroji (máme konečný počet strojů). Jaká se v tomto případě používá účelová funkce?

32.5. Řešte následující problém intervalového rozvrhování pro úlohy jednotkové váhy a dva identické stroje tak, aby byl maximalizován počet realizovaných úloh. Ukažte, jak vypadá rozvrh po *každé* iteraci algoritmu.

úloha	1	2	3	4	5	6	7
r_j	1	2	3	6	6	8	9
d_j	4	5	5	11	10	9	11

32.6. Napište algoritmus pro řešení problému intervalového rozvrhování s jednotkovou vahou úloh a nelimitovaným počtem identických strojů. Jaký je v tomto případě optimalizační cíl?

32.7. Pro problém intervalového rozvrhování s jednotkovou váhou a identickými stroji najděte minimální počet potřebných strojů tak, aby byly všechny úlohy realizovány. Rozepište všechny iterace výpočtu.

j	1	2	3	4	5	6	7
r_j	3	0	7	6	1	4	2
d_j	6	2	9	10	3	7	5

32.8. Problém intervalového rozvrhování s jednotkovou vahou úloh a nelimitovaným počtem identických strojů lze přeformulovat na speciální případ barvení grafu. Popište tuto korespondenci a diskutujte, v čem je tento speciální případ jednodušší než obecný problém barvení grafu.

33 Rezervační systém s rezervou

33.1. Napište algoritmus maximalizace váženého počtu aktivit pro rezervační systém s rezervou. Popište princip výpočtu prioritních indexů pro úlohy i stroje.

33.2. Specifikujte příklad funkce, která se používá pro výpočet prioritního indexu pro úlohu při maximalizaci váženého počtu aktivit pro rezervační systém s rezervou. Podobně napište příklad funkce pro výpočet prioritního indexu pro stroj.

33.3. Pomocí algoritmu *maximalizace váženého počtu aktivit* vytvořte rozvrh pro problém rezervačního systému s rezervou. Rozepište jednotlivé iterace algoritmu.

j	1	2	3	4	5	6
p_j	3	3	2	3	4	2
r_j	0	2	0	1	0	3
d_j	6	10	4	5	7	6
w_j	2	1	4	6	2	3
M_j	$\{1,2\}$	$\{1,2\}$	$\{1\}$	$\{2\}$	$\{1\}$	$\{2\}$

Pro uspořádání úloh použijte funkci $I_j = \frac{|M_j|}{w_j/p_j}$ a pro uspořádání strojů pak využijte funkci $g(\psi_{i,t+1},\ldots,\psi_{i,t+p_j}) = \left(\sum_{l=1}^{p_j} \psi_{i,t+l}\right)/p_j$.

J Rozvrhování jako timetabling

34 Úvod

34.1. Popište problém plánování projektu s omezenými zdroji *(resource-constrained project scheduling problem, RCPSP)*. Jak jsou od něj odvozeny problémy *rozvrhování s operátory* a *rozvrhování s pracovní sílou*? Uveďte aplikace obou problémů.

35 Rozvrhování s operátory

35.1. Popište problém rozvrhování s operátory jako problém barvení grafu. Rozlište varianty problému existence řešení a optimalizačního problému.

35.2. Napište algoritmus heuristiky, která umožní řešit problém barvení grafu. Které rozvrhovací problémy je možné pomocí této heuristiky řešit?

35.3. Pomocí heuristiky pro barvení grafu najděte rozvrh s minimálním makespan pro problém rozvrhování s operátory. Firma zaměstnává 4 operátory různých odborností.

úloha	1	2	3	4	5
A	1	1	0	0	0
B	1	0	1	0	1
C	0	0	1	1	1
D	0	1	1	1	0

35.4. Popište problém intervalového rozvrhování a problém rozvrhování s operátory. Jaká se zde používají optimalizační kritéria?

Reformulujte následující problém intervalového rozvrhování jako problém rozvrhování s operátory.

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

36 Rozvrhování s pracovní silou

36.1. Znáte nějaký problém z oblasti rozvrhování, který lze namapovat na problém plnění košů (*bin-packing*)? Popište tento problém a uveďte mapování. Uveďte příklad a popis alespoň jedné heuristiky, která se využívá při řešení problému plnění košů.

36.2. Řešte následující problém rozvrhování s pracovní silou pomocí *heuristiky prvního padnoucího koše (first fit FF)*. Firma zaměstnává 50 zaměstnanců, kteří mají realizovat 10 zakázek. Každá zakázka je zpracovávána jeden den a pracuje na ní zadaný počet zaměstnanců. Cílem řešení je narozvrhování všech zakázek v minimálním počtu dnů.

Zakázky	1	2	3	4	5	6	7	8	9	10
Počet zaměstnanců	10	10	10	10	20	20	30	30	40	40

Znáte nějakou efektivnější heuristiku pro řešení tohoto problému? Jaké bude řešení v tomto případě?

36.3. Popište heuristiku prvního padnoucího koše se zmenšováním úloh. Jaký rozvrhovací problém je možné řešit pomocí této heuristiky?

K Rozvrhování předmětů na univerzitě

37 Popis problému

37.1. Jaký je rozdíl mezi rozvrhováním s curriculy (*curriculum-based timetabling*) a rozvrhováním se zápisy studentů (*enrollment-based timetabling*)? Co je cílem řešení jednotlivých problémů?

37.2. Co to je *vážený problém splňování podmínek*? Co to jsou *měkké* a *pevné* podmíny a jaký je mezi nimi rozdíl?

37.3. U *rozvrhování předmětů na univerzitě* lze měkké podmínky lze rozdělit do čtyř základních kategorií, uveďte jejich název a příklad ke každé z nich. Uveďte dále jeden příklad pevné podmínky, která se používá v tomto problému, a přitom se nepoužívá jako měkká podmínka.

38 Iniciální tvorba rozvrhu

38.1. Napište algoritmus *iterativního dopředného prohledávání*, který se používá například při rozvrhování předmětů na univerzitě.

38.2. Jakým způsobem se udržují hodnoty pro *konfliktní statistiku* používanou u algoritmu *iterativního dopředného prohledávání*?

38.3. Předpokládejte, že *konfliktní statistika* u algoritmu *iterativního dopředného prohledávání* má uloženy následující hodnoty:

- $X = a \quad \Rightarrow \quad 1 \times \neg\, Y = b, \quad 5 \times \neg\, Y = c, \quad 50 \times \neg\, Z = a, \quad 100 \times \neg\, W = a$
- $X = b \quad \Rightarrow \quad 1 \times \neg\, Y = a, \quad 10 \times \neg\, Y = b, \quad 20 \times \neg\, Z = a$

Jaká bude evaluace hodnot a a b pomocí konflitní statistiky při výběru hodnoty proměnné X, pokud

- je současné přiřazení $Y = c, Z = a, W = a$,
- X/a vede ke konfliktu s Y/c a W/a,
- X/b vede ke konfliktu s Z/a a W/a.

Které čítače konfliktů budou změněny při výběru X/a a které při výběru X/b a jak?

39 Interaktivní rozvrhování

39.1. Jaký je smysl *interaktivního rozvrhování*? Čím se interaktivní rozvrhování liší od *iniciální tvorby rozvrhu*?

39.2. Jaké typy *mezí* využívá algoritmus *opravné verze metody větví a mezí* používaný například při rozvrhování předmětů na univerzitě? Proč jsou využívány právě tyto meze?

39.3. Popište principy algoritmu *opravné verze metody větví a mezí* používaného při interaktivním rozvrhování.