PŘIŘAZOVACÍ PROBLÉMY

LINEÁRNÍ, OBECNÝ, ÚZKOPROFILOVÝ A KVADRATICKÝ PŘIŘAZOVACÍ PROBLÉM

LINEÁRNÍ PŘIŘAZOVACÍ PROBLÉM

Přiřazovací problém spočívá v hledání optimálního vzájemně jednoznačného přiřazení prvků **dvou množin**. Obě množiny jsou konečné a obsahují stejný počet prvků. Například mějme tři firmy: F_1, F_2, F_3 a tři projekty P_1, P_2 a P_3. Každá firma může realizovat pouze jeden z projektů. Cílem je přidělit projekty jednotlivým firmám tak, aby **celkové náklady na realizaci všech projektů byly minimální**.

Označme:
c_{ij} - náklady firmy F_i na realizaci projektu P_j
x_{ij} – binární proměnná, nabývající hodnoty 1, jestliže i-té firmě přiřadíme j-tý projekt, jinak 0

Matematický model má tvar:

$\sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}\, x_{ij} \rightarrow min$	*Minimalizujeme celkové náklady na realizaci všech projektů.*
$\sum_{j=1}^{n} x_{ij} = 1, i = 1, 2, \dots, n,$	*Každé z i firem musí být přiřazen právě jeden projekt, proto součet proměnných x_{ij} přes všechny j musí být roven jedné.*
$\sum_{i=1}^{n} x_{ij} = 1, j = 1, 2, \dots, n,$	*Každý z j projektů musí být přiřazen právě jedné firmě, proto součet proměnných x_{ij} přes všechny i musí být roven jedné.*
$x_{ij} \in \{0, 1\},\ \ i, j = 1, 2, \dots, n$	$x_{ij} = 1, jestliže\ firmě\ i\ přiřadíme\ projekt\ j, jinak\ 0$

V účelové funkci nemusí být nutně jen náklady, ale třeba i čas, zkrátka cokoli, u čeho se snažíme minimalizovat součet.

```
!Bob, Tom a Jana si chtějí vypracovat státnicové otázky. Každý státnicový okruh se
skládá ze tří otázek a) (nelineární, diskrétní a simulační modely),b)(teorie her a
rozhodování),c)(ekonometrie), rozhodli se tedy, že každý vypracuje otázky k jednomu
z těchto písmen a pak si je vzájemně půjčí. Každému jde něco jiného, proto i počet
hodin, který potřebují na vypracování dané skupiny otázek, je odlišný, jak
zachycuje následující tabulka. Jejich cílem je, aby všichni dohromady strávili nad
vypracováním otázek co nejméně.
```

čas (hod)	skupina otázek		
student	a)	b)	c)
Bob	230	150	220
Tom	140	135	180
Jana	120	150	190

```
model:
sets:
student/S1,S2,S3/;
otazky/O1,O2,O3/;
prirazeni(student, otazky): x,cas;
endsets

data:
cas =
230 150 220
140 135 280
120 150 190;
enddata

min = @sum(prirazeni: cas*x);
@for(otazky(j): @sum(student(i): x(i,j)) = 1); !Každá se tří skupin otázek musí být
vypracována jedním studentem.;
@for(student(i): @sum(otazky(j): x(i,j)) = 1); !Každý student musí vypracovat jednu
ze tří skupin otázek.;
@for(prirazeni: @bin(x));
end
!Výsledek: dohromady jim to zabere 475 hodin. Nejvíce času (220 hodin) stráví Bob
na otázkami c) z ekonometrie.
```

	a)	*b)*	*c)*
Bob			x
Tom		x	
Jana	x		

ÚZKOPROFILOVÝ (BOTTLENECK) PŘIŘAZOVACÍ PROBLÉM

Stejně jako v předchozím případě hledáme optimální vzájemně jednoznačné přiřazení prvků **dvou množin**. Tentokrát se ale nesnažíme minimalizovat součet, nýbrž nejvyšší hodnotu.

Mějme opět tři firmy: F_1, F_2, F_3 a tři projekty P_1, P_2 a P_3. Každá firma může realizovat pouze jeden z projektů. Cílem je však tentokrát přidělit projekty jednotlivým firmám tak, aby celková doba realizace všech projektů byla minimální (všechny firmy začnou realizovat svůj projekt současně). Celková doba realizace bude rovna největšímu číslu c_{ij} ze všech přiřazení, protože ostatní firmy dokončí svůj projekt dřív či nejpozději zároveň. Označíme-li celkovou dobu realizace T, musí platit $c_{ij}x_{ij} \leq T$.

c_{ij} - doba realizace projektu P_j firmou F_i

x_{ij} – binární proměnná, nabývající hodnoty 1, jestliže i-té firmě přiřadíme j-tý projekt, jinak 0

$T \to min$	*Celková doba realizace musí být co nejmenší.*
$c_{ij}x_{ij} \leq T, \quad i,j = 1,2,\dots,n$	*Všechny firmy musí realizovat svůj projekt v kratší době než tato nejvyšší hodnota.*
$\sum_{j=1}^{n} x_{ij} = 1 \quad i = 1,2,\dots,n$	*Každé z i firem musí být přiřazen právě jeden projekt, proto součet proměnných x_{ij} přes všechny j musí být roven jedné.*
$\sum_{i=1}^{n} x_{ij} = 1 \quad j = 1,2,\dots,n$	*Každý z j projektů musí být přiřazen právě jedné firmě, proto součet proměnných x_{ij} přes všechny i musí být roven jedné.*
$x_{ij} \in \{0,1\}, \quad i,j = 1,2,\dots,n$	$x_{ij} = 1, jestliže\ firmě\ i\ přiřadíme\ projekt\ j, jinak\ 0$

Ne vždy je naším cílem minimalizace. Pro opačnou situaci, maximalizaci minima (např. kdyby nešlo o časy, ale o výnos jednotlivých projekt při realizaci danou firmou a chtěli bychom, aby i ten nejnižší výnos byl co možná nejvyšší), je možné od největšího prvku matice odečíst ostatní, a tak získat jakousi ztrátu proti nejlepší variantě. Tu můžeme následně minimalizovat.

```
!Bob, Tom a Jana si pomocí přiřazovacího problému rozvrhli vypracovávání otázek, viz
výše. Bob ale protestuje, protože by nad tím strávil o mnoho více času než zbylí
dva studenti. Proto se rozhodnou, že si otázky raději rozvrhnou tak, aby maximální
doba, kterou někdo z nich nad vypracováváním stráví, byla co nejmenší. Tabulka časů
se nemění:
```

čas (hod)	skupina otázek		
student	a)	b)	c)
Bob	230	150	220
Tom	140	135	180
Jana	120	150	190

	a)	*b)*	*c)*
Bob		x	
Tom	x		
Jana			x

```
model:
sets:
student/S1,S2,S3/;
otazky/O1,O2,O3/;
prirazeni(student, otazky): x,cas;
endsets

data:
cas =
230 150 220
140 135 280
120 150 190;
enddata

min = T;
@for(prirazeni(i,j): cas*x <= T);
@for(otazky(j): @sum(student(i): x(i,j)) = 1); !Každá se tří skupin otázek musí být
vypracována jedním studentem.;
@for(student(i): @sum(otazky(j): x(i,j)) = 1); !Každý student musí vypracovat jednu
ze tří skupin otázek.;
@for(prirazeni: @bin(x));
end
!Výsledek: dohromady jim to zabere 480 hodin. Nejvíce času (190 hodin) stráví Jana
nad otázkami c) z ekonometrie. Bob je spokojený, protože na něj oproti předchozímu
řešení vyšla teorie rozhodování a her, nad kterou stráví jen 150 hodin.
```

OBECNÝ PŘIŘAZOVACÍ PROBLÉM

Cílem je optimálně spárovat prvky v rámci **jedné množiny**. Může jít například o přiřazení pracovníků, tedy vytvoření dvojic pracovníků podle jejich vzdálenosti na pracovišti. Přiřazení pracovníků je ohodnoceno číslem $c_{ij} \geq 0$. Binární proměnné optimálního párování x_{ij} tvoří horní trojúhelníkovou matici X (dolní trojúhelník a diagonála nejsou v modelu užity).

Model lze formulovat takto:

$$z = \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} c_{ij}\, x_{ij} \rightarrow min$$

$$x_{ij} \in \{0, 1\}, \qquad i = 1, 2, \dots, m, \quad j = i+1, i+2, \dots, m.$$

$$\sum_{i=1}^{j-1} x_{ij} + \sum_{k=j+1}^{m} x_{jk} = 1, \qquad j = 1, 2, \dots, m,$$

Každý pracovník je přiřazen právě jednomu jinému pracovníkovi.

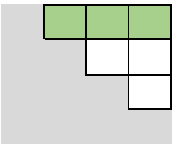
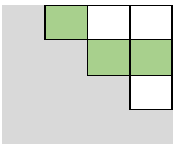
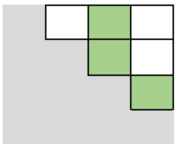
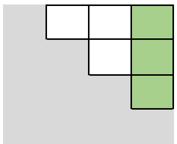
Například pro množinu o velikosti 4 bychom sčítali takto:

j = 1
$\cancel{\sum_{i=1}^{0} x_{i1}} + \sum_{2}^{4} x_{1k} = 1$
$x_{12} + x_{13} + x_{14}$

j = 2
$\sum_{i=1}^{1} x_{i2} + \sum_{3}^{4} x_{2k} = 1$
$x_{12} + x_{23} + x_{24}$

j = 3
$\sum_{i=1}^{2} x_{i3} + \sum_{4}^{4} x_{3k} = 1$
$x_{13} + x_{23} + x_{34}$

j = 4
$\sum_{i=1}^{3} x_{i4} + \cancel{\sum_{5}^{4} x_{4k}} = 1$
$x_{14} + x_{24} + x_{34}$

```
!Bob, Tom a Jana a jejich tři kamarádi, kteří se také chystají na státnice, bydlí
na jedné koleji. Týden před státnicemi nebozí studenti s hrůzou zjistili, že ve
školní knihovně jsou k dispozici pouze tři výtisky Ekonometrické analýzy, z níž se
chtějí učit. Protože by nikoho z nich ani nenapadlo knihu šířit nelegálně nebo ji
stahovat z internetu, ale kupovat si ji také nechtějí, vzali si z knihovny tyto tři
výtisky s tím, že vytvoří dvojice, v nichž si budou vždy jeden výtisk půjčovat.
Chtějí vytvořit takové dvojice, aby vzdálenost, kterou musí v cestě za knihou
překonávat, byla minimální. Vzdálenosti, které musí překonávat, zachycuje
následující matice.
1 5 6 3 2
  8 4 2 1
    2 1 7
      4 6
        5;
model:
sets:
student/1..6/;
prirazeni(student,student)|&1#LT#&2: x,c; !LT znamená Less Than, pracujeme jen s
horní trojúhelníkovou maticí;
endsets

data:
c =
1 5 6 3 2
   8 4 2 1
      2 1 7
         4 6
            5;
enddata

min = @sum(prirazeni(i,j): c*x); !nemusíme vypisovat o které indexy se jedná,
protože v množinách je máme definované;
@for(student(j): @sum(student(i)|i#LT#j: x(i,j)) + @sum(student(k)|k#GT#j: x(j,k))
= 1); !kazdy student bude spárovaný právě s jedním dalším studentem;
@for(prirazeni: @bin(x));
end
```

KVADRATICKÝ PŘIŘAZOVACÍ PROBLÉM

Uvažujme *n* strojů $S_1, S_2, \ldots, S_n$ a *n* míst $M_1, M_2, \ldots, M_n$. Pro každou dvojici míst známe jejich vzdálenost d_{kl}. Stroje vyrábí součástky, které se mezi těmito místy přepravují na vozících, přičemž množství součástek přemístěných mezi dvěma stroji označíme c_{ij}. Cílem je rozmístit stroje na místa tak, aby celková vzdálenost, kterou vozík urazí při rozvozu výrobků mezi stroji, byla minimální.

Kvadratický přiřazovací problém lze formulovat takto:

$$\sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij} \sum_{k=1}^{n}\sum_{l=1}^{n} d_{kl}x_{ik}x_{jl} \rightarrow min$$

Pokud umístíme stroj i na místo k a zároveň stroj j na místo l, pak mezi nimi c_{ij} součástek urazí vzdálenost d_{kl}.

$$\sum_{j=1}^{n} x_{ij} = 1, i = 1, 2, \ldots, n,$$

Každý stroj musí být umístěn právě na jedno místo.

$$\sum_{i=1}^{n} x_{ij} = 1, j = 1, 2, \ldots, n,$$

Na každém místě musí být právě jeden stroj.

$$x_{ij} \in \{0, 1\},\ \ i = 1, 2, \ldots, n,\ \ j = 1, 2, \ldots, n.$$

x_{ij} je binární proměnná, která je rovna 1 tehdy, pokud umístíme stroj i na místo j, jinak je rovna 0.

Model linearizujeme následovně: součin $x_{ik}x_{jl}$ nahradíme binární proměnnou y_{ijkl}, která se bude rovnat 1 v případě, že jak x_{ik}, tak x_{jl} jsou rovny 1, tedy jen tehdy, když bude stroj *i* na místě *k* a stroj *j* na místě *l*. Přidáme omezující podmínku:

$$y_{ijkl} \geq x_{ik} + x_{jl} - 1 \quad i, j, k, l = 1 \ldots n$$

```
!Vedení podniku řeší problém, jak umístit v nové výrobní hale 4 stroje na 4
pracovní plochy. Vzdálenosti mezi pracovními plochami jsou dány maticí D. Při
výrobě se na strojích zpracovávají polotovary, které se přepravují v bednách na
vozíku. Počty beden, které se za hodinu přepraví mezi stroji, jsou dány hodnotami v
matici C. Cílem je minimalizovat celkovou vzdálenost, kterou vozík přejede při
přepravě polotovarů mezi stroji. Stroje mohou být umístěny na libovolné místo.
```

C =	0	3	4	5		D =	0	10	15	12
	2	0	5	7			10	0	11	13
	4	5	0	6			15	11	0	14
	3	4	6	0			12	13	14	0

```
; model:
sets:
stroj/1..4/;
misto/1..4/;
umisteni(stroj,misto):x;
SS(stroj,stroj): c;
MM(misto,misto): d;
tabulka(stroj,stroj,misto,misto):y;
endsets

data:
d=     0 10 15 12
       10 0 11 13
       15 11 0 14
       12 13 14 0;
c=     0 3 4 5
       2 0 5 7
       4 5 0 6
       3 4 6 0;
enddata

@for(stroj(i): @sum(misto(j): x(i,j)) = 1);
@for(misto(j): @sum(stroj(i):x(i,j)) = 1);
@for(umisteni: @bin(x));
@for(tabulka: @bin(y));
@for(tabulka(i,j,k,l): y(i,j,k,l)>=x(i,k)+x(j,l)-1);
min=@sum(umisteni(i,j): c(i,j)*@sum(umisteni(k,l): d(k,l)*y(i,j,k,l)));
end
Zdroj: Ing. Jan Fábry, Ph.D.: 4EK314 Diskrétní modely
```

ZDROJE:

Ing. J. Fábry, Ph.D.: přednášky 4EK314 Diskrétní modely, 2011.