
Optimalizácia rezania hutného materiálu

Ivan Brezina¹, Jaroslav Kultán², Juraj Pekár³

Abstrakt

Pri rezaní hutného materiálu, hlavne kovových tyčí, je potrebné vytvoriť plán optimálneho využitia dĺžok daného materiálu s cieľom minimalizácie strát materiálu. Materiál je dodávaný vo forme tyčí určitej dĺžky a objednávky sú rozložené do súpisky materiálu, ktorý je potrebné narezať. V prípade zostatku tyče, táto je uložená naspäť do skladu. Cieľom príspevku je vytvorenie metodiky optimalizácie rezania kovových tyčí s cieľom minimalizácie odpadu. Bolo tiež vytvorené základné programové vybavenie v tabuľkovom editore MS Excel a v jazyku Python. V závere príspevku je uvedený príklad riešenia optimálneho rozdelenia tyčového materiálu pre firmu. Na základe požiadaviek odberateľa a zoznamu disponibilného materiálu bol vytvorený rezný plán a vypočítaný celkový odpad. Príspevok môže slúžiť ako podklad na vytvorenie dátového skladu, ktorý uchováva údaje nielen o nakúpených materiáloch, realizovanej produkcii, ale aj o zostatkoch materiálu, ktorý môže byť využitý pri ďalšom rezaní. Dátový sklad pritom umožňuje multidimenzionálny pohľad na riadenie procesu rezania.

Kľúčové slová

optimálne rozdelenie materiálu, Python, minimalizácia strát, rezný plán, matematické programovanie

Abstract

When cutting of metallurgical materials, especially metal bars, it is necessary to create a plan for optimal use of the material lengths to minimize losses of material. The material is supplied in the form of bars and some of orders are distributed checks to the material that needs to be cut. In the case of the balance of the rod, it is stored back in the warehouse. The aim of the paper is to create a methodology for optimizing the cutting of metal bars in order to minimize waste. The basic software in the spreadsheet editor MS Excel and in the Python language was also created. At the end of the article, an example of the solution of the optimal distribution of bar material for the company is given. Based on the customer's requirements and the list of available material, a cutting plan was created and the total waste was calculated. The article can serve as a basis for the creation of a data warehouse, which stores data not only on purchased materials, realized production, but also on the balances of material that can be used in further cutting. Data warehouse thereby allows a multidimensional view at cutting process management..

Keywords

optimal material distribution, Python, loss minimization, cutting plan, mathematical programming

JEL classification

C61, C63

¹ Ekonomická univerzita v Bratislave, Fakulta hospodárskej informatiky, Katedra operačného výskumu a ekonometrie, Dolnozemska cesta 1/b, 852 35 Bratislava, ivan.brezina@euba.sk.

² Ekonomická univerzita v Bratislave, Fakulta hospodárskej informatiky, Katedra aplikovanej informatiky, Dolnozemska cesta 1/b, 852 35 Bratislava, jaroslav.kultan@euba.sk.

³ Ekonomická univerzita v Bratislave, Fakulta hospodárskej informatiky, Katedra operačného výskumu a ekonometrie, Dolnozemska cesta 1/b, 852 35 Bratislava, juraj.pekar@euba.sk.

1 Úvod

Pri rezaní hutného materiálu (napr. kovových tyčí) často vznikajú veľké straty v dôsledku nesprávne zvoleného rezného plánu. V prípade, ak ide o veľké množstvo kovových tyčí rôzneho priemeru a dĺžky, z ktorých treba narezat' určitý počet kratších kusov pre ďalšie spracovanie, akákoľvek nepresnosť môže spôsobiť veľké materiálne a teda aj ekonomické straty. Práve preto je otázke optimálneho rozdelenia rezov jednotlivých disponibilných kovových tyčí na požadované množstvo zadanej dĺžky venovaná veľká pozornosť, pretože pri každej dodávke môže dochádzať k značným zbytočným finančným nákladom.

Mnohé slovenské firmy tento problém riešia skúsenými pracovníkmi, ktorí na základe empirických skúseností odhadom tvoria rezný plán. Efektívnosť takejto činnosti je veľmi sporná a často sa stáva, že firmy straty materiálu započítajú do ceny svojich výrobkov. Tým sa stávajú menej konkurenčne schopnými.

V príspevku je prezentované možné riešenie tohto problému pre firmu, ktorá má veľké objednávky na prípravu železných armatúrových konštrukcií na veľké stavby. Optimálne rozdelenie rezov môže priniesť veľké finančné úspory pre danú firmu.

Uvedený problém možno formulovať ako matematický model, ktorý predstavuje zjednodušený opis reálneho systému rezania hutného materiálu (kovových tyčí). Ako efektívne sa pritom javí použitie najjednoduchších matematického programovania, ktoré umožňujú pomerne jednoducho transformovať analyzované procesy do matematického modelu v tvare funkcií, nerovnic, resp. rovníc. Formulovaný model možno riešiť nástrojmi matematického programovania, podporenými zodpovedajúcimi softvérovými nástrojmi, konkrétne programom Python. Takýmto postupom možno relatívne presne kvantitatívne realizovať výber určitého riešenia z veľkého množstva možných riešení, ktoré je z hľadiska matematicky formulovaného cieľa najlepšie.

Príspevok vznikol ako výsledok riešenia konkrétneho problému firmy, ktorá realizuje rezanie železných armatúrových tyčí pre potreby veľkých stavieb, konkrétne pre diaľničnú výstavbu. Autori v ňom zovšeobecnilí skúsenosti nadobudnuté počas riešenia tejto úlohy s využitím nástrojov matematického programovania a programu Python.

Najzaujímavejším problémom pritom nebolo riešenie skonštruovanej úlohy matematického programovania z literatúry známej ako rezný problém, ale konštrukcia vlastných rezných plánov, ktoré sú podkladom pre zostavenie zodpovedajúcej úlohy matematického programovania. Zostavenie rezných plánov pri rozsiahlych zadaniach v sebe zahŕňa veľké množstvo variantov, ktoré možno eliminovať pomocou vhodne formulovaných pravidiel a využiť pritom tiež program Python.

Na základe riešenia príslušnej úlohy matematického programovania možno vytvoriť dátovú štruktúru – dátový sklad (DW), ktorá umožní uchovávať nielen údaje o materiáloch na sklade, výsledných produktoch, ale aj jednotlivé vypočítané riešenia. DW taktiež uchováva údaje o zostatkoch materiálu. DW môže poskytnúť mnohorozmerný pohľad na získané údaje, čo môže byť predpokladom pre efektívnejšie využívanie pracovného času a materiálu.

2 Popis problému

Firma RB, Veľké Kapušany s.r.o. sa zaoberá rozdelením dlhých kovových tyčí rôznych rozmerov na požadované dĺžky v súlade s požiadavkami odberateľa. Na základe objednávok a výrobných výkresov sú vytvorené štítky určené pre ohýbacie stroje.

Na Obrázku 1 sú uvedené jednotlivé objednávky, ktoré musí firma v danom čase realizovať. Na základe jednotlivých štítkov treba vytvoriť zoznam požadovaných priemerov, tvarov a dĺžok kovových tyčí.

Obr. 1: Zadania na jednotlivé objednávky



Zdroj: Firma RB, Veľké Kapušany s.r.o.

Obrázky 2 a 3 predstavujú príklady podrobného zadania počtu a typu tyčí treba narezať. V ľavej časti Obrázku 2 je uvedené číslo nakładky (103), číslo výrobného výkresu (10547), číslo výkresu, na základe ktorého sa jednotlivé kusy vyrábajú (Hlava P10), číslo položky (13.S). Hlavné údaje o príslušnom materiáli sú priemer 14 mm, dĺžka 2,29 m, počet kusov 4. Okrem uvedených údajov sú tam údaje o odberateľovi (Váhostav a.s.) a dodávateľovi (RB, Veľké Kapušany s.r.o.) a mieste dodania resp. názov stavby (Most na R4 v km 3,10 nad). Obrázok 3 obsahuje zodpovedajúce údaje pre iného odberateľa, VHS – PS s.r.o pre stavbu EUROVEA 2.

Obr. 2: Základný lístok o výrobku pre Most R4


Stavba : Most na R4 v km 3,10 nad	Váha: 11,01	pol : 13.S
č. výkresu : Hlava P10	č. výkresu : Hlava P10	
Ø : 14	č. nákladky : 103	
dĺžka : 2,29	č. výrobn. výkresu : 10547	
ks : 4	ks : 4	Ø 14 Dĺžka: 2.28
č. výr. výkresu : 10547		
Odberateľ: Vahostav- SK a.s		Twinmaster 16 50
Dodávateľ: RB, Veľké Kapušany s.r.o.		

Diagram: 75° , $M=205$, $R=3$, $M=13$, 10 , 199 , 5 , 10 , 10

pol:13.S
103

Zdroj: Firma RB, Veľké Kapušany s.r.o.

Obr. 3: Základný lístok o výrobku pre EUROVEA 2

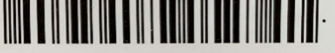
Stavba : EUROVEA 2	Váha: 373,69	pol : 2.1
č. výkresu : v1120	č. výkresu : v1120	
Ø : 12	č. nákladky : 110	
dĺžka : 4,07	č. výrobn. výkresu : 10562	
ks : 106	ks : 106	Ø 12 Dĺžka: 3.97
č. výr. výkresu : 10562		
Odberateľ: VHS - PS s.r.o.		Twinmaster 16 48
Dodávateľ: RB, Veľké Kapušany s.r.o.		

Diagram: 20 , 20 , 36 , 36 , 295

pol:2.1
110

Zdroj: Firma RB, Veľké Kapušany s.r.o.

Z obrázku 3 je pri porovnávaní dĺžky výrobku uvedenej na pravej strane 4,07 a na ľavej strane 3,97 rozdiel, ktorý je daný tým, že na ľavej strane je schéma výrobku s pravými uhlami, ale pri reálnom ohybe vzniká nie pravý uhol ale oblúk, ktorého celková dĺžka je menšia. Uvedené rozmery je vidieť aj na **Chyba! Nenašiel sa žiaden zdroj odkazov.**, na ktorom sú uvedené položky zaokrúhlenia. Napr. pre položky 1, 6, 11, 12 sú uvedené teoretické rozmery a praktické oblúky.

Z uvedeného výkresu na Obrázku 4 sú zrejme aj ďalšie základné informácie o rezaných prvkoch pre Most na R4 v km 3,10 nad riekou Torysou (napr. pre položku 1) reprezentované Tabuľkou 1.

Tab. 1: Základné informácie o rezaných prvkoch pre Most na R4 v km 3,10 nad riekou Torysou pre položku 1

Výrobný výkres	10505
Položka	1
Množstvo	758
Priemer	20
Dĺžka	2,94
/doplňujúci údaj z obr. 2: číslo nakládky	Napr. 103

Zdroj: Firma RB, Veľké Kapušany s.r.o.

Na základe objednávky, resp. výrobného výkresu, možno vidieť, že jeden a ten istý priemer sa používa na rezanie viacerých položiek a tiež aj rôznych tvarov.

Obr. 4: Nákres jednotlivých prvkov v objednávke

Výrobný výkres číslo :		10505		Vypracoval: ATTILA TITKA				
Objednávateľ :	Vahostav- SK a.s., 821 09 Bratislava			Objednávka dosla:	10.07.20			
Stavba :	Most na R4 v km 3,10 nad riekou Torysou			Spracovan:	0,00			
Stavbyvedúci :				Dátum dodania:	03.08.20			
Preberajúci :				Váha spolu :	30734kg			
Názov projektu :	Most na R4 v km 3,10 nad riekou Torysou							
Adresa dodania :				Dátum :	13.07.2020			
Most na R4 v km 3,10 nad riekou Torysou				Strana :	1			
				Vystavil :	ATTILA			
Rozvrh c : VYSTUŽ NOSNEJ KONSTRUKCIE								
Popis rozvrhu : 0.3.7								
Polozka	Mnozstvo	Kvalita	Priemer (mm)	Dlžka (m)	Celková dlžka (m)	Váha (kg)	Kód tvaru	Poznámka
1	758	B500B	20	2,94	2 224,73	5 495,08	18ki	
6	758	B500B	25	3,74	2 834,92	10 914,44	18la	
11	758	B500B	16	6,50	4 923,21	7 778,67	18lk	
12	120	B500B	22	12,00	1 440,00	4 291,20	1	

Zdroj: Firma RB, Veľké Kapušany s.r.o.

Pre účely firmy je pritom kľúčovým slovom číslo nakládky a celkový kľúč je číslo nakládky, číslo výkresu (103 z Obrázku 2 a 10505 z Obrázku 4)

Optimalizácia rezania zodpovedajúcich tyčí vychádza z predpokladu, že na rezanie jednotlivých tyčí treba zostrojiť rezný plán, v rámci ktorého budú uvedené základné údaje jednotlivých rezov. V súlade s reznými plánmi pre každý priemer treba realizovať viacero objednávok, v rámci ktorých je uvedených pre daný priemer viacero tyčí, smerovaných na rez.

Postup a previazanosť jednotlivých činností (algoritmizácia postupu):

1. Vytvorenie tabuľky, v ktorej budú údaje o objednávkach, priemeroch, množstvách
 - 1.1. Vypísanie požiadaviek z každej objednávky do tabuľky
 - 1.2. Uloženie jednotlivých položiek podľa priemeru železných tyčí
2. Výber tabuľky podľa priemerov a dĺžok disponibilných tyčí na sklade
3. Vytvorenie rezného plánu pre prvý uvažovaný priemer
 - 3.1. Zostavenie tabuľky rezných plánov
 - 3.2. Vytvorenie zodpovedajúcej úlohy matematického programovania (rezný problém)
 - 3.3. Vyriešenie rezného problému pre prvý priemer
 - 3.4. Zapísanie údajov do tabuľky rezov, ktorá bude obsahovať nasledujúce údaje:
 - a. Číslo objednávky / číslo nakládky
 - b. Číslo položky
 - c. Priemer
 - d. Počet tyčí
 - e. Počet rezov
 - f. Odpad z tyče
 - g. Celkový odpad
 - 3.5. Ukončenie prípravy rezu pre daný priemer
4. Výber ďalšieho priemeru a opakovanie bodu 3
5. Vytvorenie súboru obsahujúceho jednotlivé priemery zoradené podľa objednávky, položiek, počtu použitých tyčí a počtov rezov jednotlivých dĺžok

Uvedenú tabuľku možno v rámci Excelu ukladať tak podľa priemerov, ako aj podľa objednávok alebo iných parametrov.

3 Rezný problém ako úloha matematického programovania

Rezný problém (cutting-stock problem – CSP, napr. Kellerer, 2004, Jahromi, 2012) patrí do triedy úloh matematického programovania, ktoré sú konštruované s cieľom hľadania najlepšieho riešenia pri zadaných ohraničujúcich predpokladoch.

Model celočíselného lineárneho programovania možno zapísať vo všeobecnom tvare:

$$\begin{aligned} \max (\min) f(\mathbf{x}) &= \sum_{j=1}^n c_j x_j \\ \sum_{j=1}^n a_{ij} x_j &\begin{cases} \leq \\ = \\ \geq \end{cases} b_i, \quad i = 1, 2, \dots, m \\ x_j &\geq 0, \quad j = 1, 2, \dots, n \\ x_j &\in D_j, \quad j = 1, 2, \dots, n \\ x_j &\in Z, \quad j = 1, 2, \dots, n \end{aligned} \tag{1}$$

kde n – počet rozhodovacích premenných,
 m – počet ohraničujúcich podmienok (štruktúrnych ohraničení),
 x_j – rozhodovacie premenné, $j = 1, 2, \dots, n$,

- c_j – koeficienty účelovej funkcie (ÚF), $j = 1, 2, \dots, n$,
 a_{ij} – technologické koeficienty sústavy ohraňení, $i = 1, 2, \dots, m, j = 1, 2, \dots, n$,
 b_i – koeficienty pravej strany (PS), $i = 1, 2, \dots, m$,
 D_j – definičné obory čísiel, ktoré môžu nadobúdať x_j , $j = 1, 2, \dots, n$,
 Z – množina celých čísiel.

Pri riešení zodpovedajúcich úloh celočíselného lineárneho programovania sa hľadá taký vektor premenných $\mathbf{x}^T = (x_1, x_2, \dots, x_n)$, pre ktorý je hodnota ÚF maximálna (alebo minimálna) pri splnení ohraňujúcich predpokladov.

Rezný problém (úloha o delení materiálu, optimálny rezný plán) predstavuje model celočíselného lineárneho programovania, v rámci ktorého sa rieši rozdelenie (rezanie) určitých väčších celkov na menšie tak, aby bol minimalizovaný odpad po ich rozdelení. Pri reznom probléme treba pritom rešpektovať požiadavky na počet vzniknutých menších celkov po rezaní, teda koľko ich má minimálne resp. maximálne rezaním vzniknúť a takisto požiadavky na maximálny prípustný odpad pri ich rezaní, prípadne ďalšie obmedzenia.

Úloha môže byť jednorozmerná, v ktorej je delenie charakterizované iba jedným rozmerom – napr. tyče, latky, káble, pásy rovnakej šírky a pod. (one-dimensional cutting stock problem, napr. Tanir et al., 2016, Tanir et al., 2019, Ágoston, 2019) alebo dvojrozmerná, v ktorej sa z plochy sa vyrezávajú menšie celky (two-dimensional cutting stock problem, napr. Malaguti et al., 2014, Furini et al., 2016). Vo všeobecnosti možno charakterizovať rezné problémy podľa ich dimenzie alebo podľa ich tvaru. Dimenzia v Euklidovskom priestore môže byť jednorozmerná 1D, dvojrozmerná 2D, resp. trojrozmerná 3D. Dimenzia mimo Euklidovského priestoru môže byť závislá napríklad na čase, hmotnosti atď., pričom táto dimenzia sa zvyčajne aplikuje ako doplnujúce požiadavky pre priestorové objekty. Podľa tvaru objektov možno charakterizovať rezné problémy ako pravouhlé objekty (2D – obdĺžniky, 3D – kvádre, atď.), na základe geometrických tvarov na obdĺžniky, kruhy elipsy atď.

Jednorozmerný rezný problém možno formulovať ako klasickú úlohu celočíselného lineárneho programovania, podstata dvojrozmerného rezného problému (ako aj ďalších variantov rezného problému) je podstatne zložitejšia.

Formulovaný problém rezania kovových tyčí predstavuje vo svojej podstate jednorozmerný rezný problém. Problémom tohto typu úlohy lineárneho programovania je určiť, čo predstavuje v úlohe rozhodovacie premenné. Každý uvažovaný väčší celok možno rozdeliť (rozrezať) na menšie celky veľkým množstvom spôsobov a preto sa pod procesom rozumie aplikácia jedného z potenciálnych spôsobov delenia. Potom budú procesom (spôsobom delenia) odpovedať jednotlivé rozhodovacie premenné a hodnoty rozhodovacích premenných budú udávať, koľkokrát bol príslušný spôsob delenia použitý. Ohraňenia úlohy celočíselného lineárneho programovania pritom musia zahrnúť všetky obmedzujúce podmienky vyplývajúce zo zadania (minimálny, resp. presný počet menších celkov, obmedzenie počtu disponibilných štandardných väčších celkov (tyčí), požiadavky na maximálny prípustný odpad a pod.).

Praktické nájdenie kvalitného rezného plánu (všetkých možných spôsobov rezania) predstavuje matematicky NP – úplný problém, ktorého riešenie je aj v súčasnosti časovo a pamäťovo náročné.

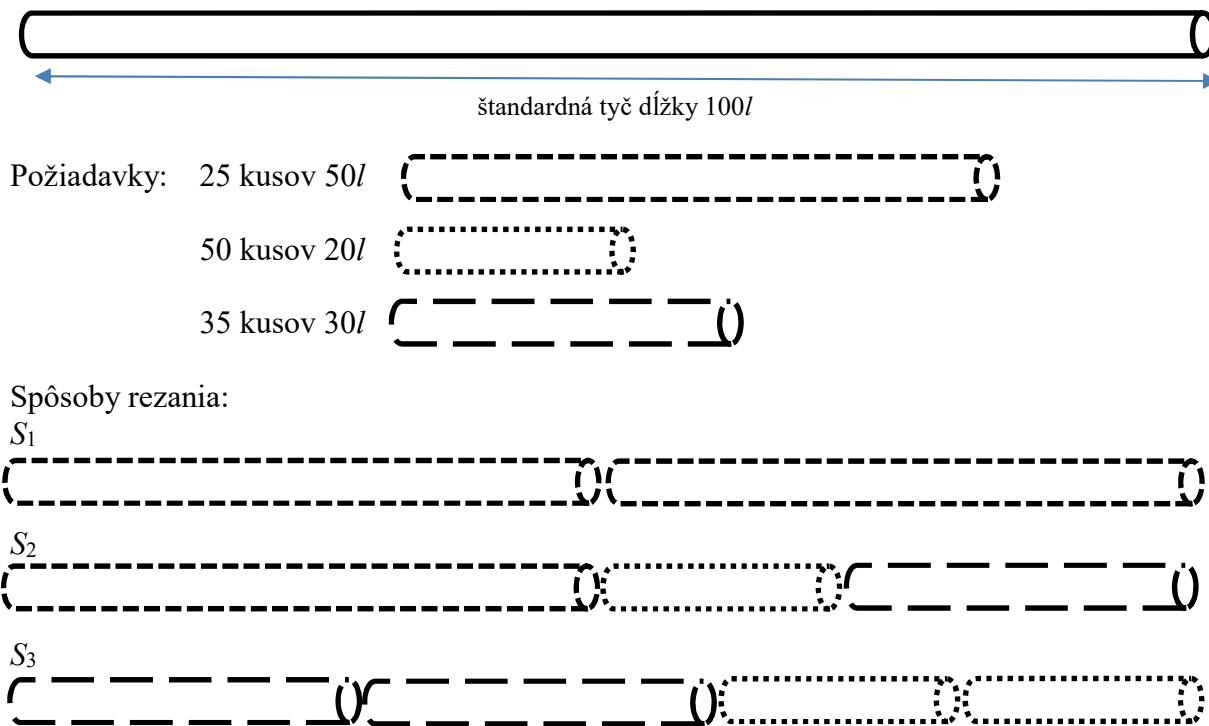
Pri formulovaní rezného problému ako úlohy celočíselného lineárneho programovania treba najskôr vytvoriť rezné plány (možné spôsoby delenia väčších celkov), treba teda zistiť koľko spôsobov delenia je pri delení väčších celkov možných. Počet spôsobov rezania potom reprezentuje počet rozhodovacích premenných (n), ktoré sú do úlohy celočíselného lineárneho programovania zakomponované. Štruktúrne ohraňenia (m) potom reprezentujú všetky ohraňenia, ktoré treba do úlohy zahrnúť, pričom základné ohraňenia pre počty narezaných menších celkov sú zvyčajne v tvare „=“, ak treba narezať presný počet menších celkov, resp. v tvare „ \geq “, a sa má dosiahnuť aspoň ich určitý počet. Úlohu nájdenia optimálneho rezného plánu možno potom modifikovať z úlohy (1) na tvar:

$$\begin{aligned} \min f(\mathbf{x}) &= \sum_{j=1}^n c_j x_j \\ \sum_{j=1}^n a_{ij} x_j &\begin{cases} = \\ \geq \end{cases} b_i, \quad i = 1, 2, \dots, m \\ x_j &\geq 0, \quad j = 1, 2, \dots, n \\ x_j &\in D_j, \quad j = 1, 2, \dots, n \\ x_j &\in Z, \quad j = 1, 2, \dots, n \end{aligned} \quad (2)$$

kde podmienky $x_j \in D_j, j = 1, 2, \dots, n$ reprezentujú, ako už bolo uvedené, dodatočné obmedzenia (napr. počtu disponibilných štandardných väčších celkov (tyčí), požiadavky na maximálny prípustný odpad a pod.).

Rezný plán potom môže reprezentovať Obrázok 5 (predpokladáme celkovú dĺžku štandardnej tyče 100, požadovaných 25 tyčí dĺžky 50, 50 tyčí dĺžky 20 a 35 tyčí dĺžky 30, odpad nie je prípustný), resp. všeobecná Tabuľka 2 (hodnota m_1 predstavuje celkový počet rôznych druhov požadovaných menších celkov).

Obr. 5: Možnosti rezania menších celkov



Zdroj: Vlastné spracovanie

Na základe vytvoreného rezného plánu možno sformulovať úlohu celočíselného lineárneho programovania a riešiť ju zodpovedajúcimi metódami, ktoré boli vypracované v rámci teórie celočíselného programovania (metóda vetiev a hraníc, Gomoryho metóda rezných nadrovín...).

Tab. 2: Vytvorenie rezného plánu

Dĺžka celku	Spôsoby rezania						Požadovaný počet kusov b_i
	S_1	S_2	S_3	S_n	
d_1 cm	a_{11}	a_{12}	a_{13}			a_{1n}	b_1
d_2 cm	a_{21}	a_{22}	a_{23}			a_{2n}	b_2
...							...
d_{m1} cm	$a_{m1,1}$	$a_{m1,2}$	$a_{m1,3}$			$a_{m1,n}$	b_{m1}
Odpad	o_1	o_2	o_3			o_n	
Rozhodovacia premenná	x_1	x_2	x_3	x_n	

Zdroj: Vlastné spracovanie

Na reálne riešenia úloh matematického programovania, a teda aj úloh celočíselného lineárneho programovania, boli vyvinuté profesionálne softvérové produkty ako sú GAMS, LINDO, LINGO, Python, Gurobi, atď. Ale v praxi, aj na akademickej pôde sa používa mnoho ďalších softvérov a softvérových jazykov na riešenie matematického programovania (napr. MATLAB, jazyk R, Python a pod.). Ďalej prezentujeme spôsob riešenia úlohy rezného problému ako úlohy celočíselného lineárneho programovania v jazyku Python.

4 Riešenie rezného problému ako úlohy lineárneho programovania v jazyku Python

Pri riešení úloh v jazyku Pythone treba v prvom rade určiť prostredie programovania (táto otázka nie je predmetom príspevku). Okrem základného prostredia IDLE (Integrated Development and Learning Environment) možno použiť nástroje ako Jupyter, Spyder, ktoré možno nájsť pod integrovaným systémom Anaconda. Ďalej bude predstavené riešenie úlohy celočíselného lineárneho programovania (ako podmnožina úloh matematického programovania) v prostredí IDLE.

Ďalším krokom je výber vhodného modulu určeného na riešenie úloh matematického programovania všeobecne. Pre praktické riešenie sú k dispozícii dva základné moduly SCIPY a MIP. Prvý uvedený modul je určený na riešenie úloh matematického programovania (lineárne a nelineárne úlohy). Modul MIP je na rozdiel od modulu SCIPY zameraný na riešenie úloh lineárneho, celočíselného a tiež aj bivalentného programovania. Variabilitu modulu MIP možno charakterizovať možnosťou výberu optimalizačného riešiteľa Coin-or branch and cut (CBC), ktorý je automaticky inštalovaný s modulom MIP, Gurobi (GRB), SCIP, GLPK, ktoré možno použiť v prípade ich nainštalovania z iných zdrojov ako Python.

Ďalej bude prezentovaný príklad riešenia rezného problému ako úlohy celočíselného lineárneho programovania, ktorý bol ilustratívne znázornený na Obrázku 5. Uvažujme teda úlohu, keď sú k dispozícii štandardné tyče s dĺžkou 100 (predpokladáme dostatočný počet týchto tyčí), je potrebné narezať 25 tyčí dĺžky 50, 50 tyčí dĺžky 20 a 35 tyčí dĺžky 30, pričom odpad nie je prípustný. Rezný plán je uvedený v Tabuľke 3.

Tab. 3: Vytvorenie rezného plánu s odpadom 0

Dĺžka tyče	Spôsoby rezania				Požadovaný počet kusov b_i
	S_1	S_2	S_3	S_4	
50 cm	2	1	-	-	25
30 cm	-	1	2	-	35
20 cm	-	1	2	5	50
Odpad	0	0	0	0	
Rozhodovacia premenná	x_1	x_2	x_3	x_4	

Zdroj: Vlastné spracovanie

Potom možno formulovať nasledujúcu úlohu lineárneho programovania:

$$\begin{aligned} \min f(\mathbf{x}) &= 0x_1 + 0x_2 + 0x_3 + 0x_4 \\ 2x_1 + 1x_2 + 0x_3 + 0x_4 &= 25 \\ 0x_1 + 1x_2 + 2x_3 + 0x_4 &= 35 \\ 0x_1 + 1x_2 + 2x_3 + 5x_4 &= 50 \\ x_1, x_2, x_3, x_4 &\geq 0 \\ x_1, x_2, x_3, x_4 &\in Z \end{aligned} \tag{3}$$

Ďalej je opísaný kód (# - reprezentuje komentár v jazyku Python) na riešenie úlohy (3) v Pythone pomocou modulu MIP:

```
# importovanie prvkov z modulu MIP
from mip import Model, xsum, minimize
# vstupne udaje
c = [0, 0, 0, 0]
A = [[2,1,0,0],[0,1,2,0],[0,1,2,5]]
b=[25,35,50]
n, V = len(c), set(range(len(c)))
m, U = len(b), set(range(len(b)))
md = Model('ULP')
# definovanie premenných
x = [md.add_var(var_type="I") for i in V]
# ucelova funkcia
md.objective = minimize(xsum(c[i] * x[i] for i in V))
# ohranicenia ulohy
for i in U:
    md += xsum(A[i][j]* x[j] for j in V) == b[i]
# spustenie riesenia
vysledok=md.optimize()
# vypis vysledkov)
for i in md.vars:
    print(i.name,i.x)
print(vysledok)
print("Ucelova funkcia: ",md.objective_value)
```

Výpis výsledkov po spustení v prostredí IDLE:

```
var(0) 0.0
var(1) 25.0
var(2) 5.0
var(3) 3.0
OptimizationStatus.OPTIMAL
Ucelova funkcia: 0.0
```

5 Implementácia problému rezania železných armatúrnych tyčí v jazyku Python

Pri riešení uvedeného problému bol použitý programovací jazyk Python, ktorý bol prezentovaný ako nástroj na riešenie rezného problému v časti 3. Prezentovaný kód obsahuje dva základné bloky, blok vytvorenia prípustných rezných plánov a blok vlastnej optimalizácie úlohy lineárneho programovania.

Prvý blok možno rozdeliť na dve podčasti. V prvej časti sa vytvorí množina všetkých potenciálne použiteľných kombinácií spôsobov rezania – rezný plán (Obrázok 6). Prvá časť

kódu slúži k zisteniu maximálneho možného počtu kusov získaných z najdlhšej tyče, t. j. stanovenie hodnoty premennej mm . V ďalšej časti je zostrojená množina všetkých m prvkových kombinácií, pričom hodnoty nadobúdajú hodnoty od 0 po $mm+1$, aby bolo zabezpečené, že nebude vynechaná žiadna prípustná kombinácia rezného plánu.

Obr. 6: Vytvorenie rezného plánu pri stanovení maximálneho počtu kusov výrobkov rovnakej dĺžky z jednej tyče

```
#stanovenie maximálneho počtu kusov výrobkov rovnakej dĺžky z jednej tyče
mm=0
for k in T:
  for i in V:
    if mm<=tyce[k][0]//vstup[i][0]:
      mm=tyce[k][0]//vstup[i][0]
#vytvorenie všetkých potenciálne prípustných kombinácií
res = [ele for ele in product(range(0, mm + 1), repeat = m)]
```

Zdroj: Vlastné spracovanie

V druhej časti je potrebné vybrať z množiny všetkých spôsobov rezania iba tie, ktoré spĺňajú podmienky maximálnej novej hodnoty odpadu pri rezaní (Obrázok 7). V tomto bloku sa realizuje kontrola, či je pre jednotlivé typy vstupných tyčí rezný plán prípustný. Kontrola prípustnosti spočíva v zistení, či rezný plán je možné realizovať, t. j. súčet narezaných tyčí nie je väčší ako jej dĺžka a odpad je maximálne na stanovenej úrovni. Po výbere prípustných spôsobov rezania možno získať požadované parametre matematického modelu, kde v naprogramovanom kóde premenná rp reprezentuje množstvá získaných požadovaných tyčí z jedného rezania, tyc dĺžku rezanej tyče a o odpad po použití príslušného rezného plánu.

V druhom bloku sa rieši optimalizačná úloha, ktorá je všeobecne formulovaná ako úloha lineárneho programovania (2), pričom na obrázku (7) je úloha zapísaná v programovacom jazyku Python.

Obr. 7: Výber prípustných rezných plánov pri stanovenom maximálnom odpade

```
#výber prípustných rezných plánov pri stanovenom maximálnom odpade 200 mm
modpad=200
for k in T:
  for i in list(res):
    if sum(np.multiply(dlzkky,np.array(i)))<=tyce[k][0]:
      if tyce[k][0]-sum(np.multiply(dlzkky,np.array(i)))<=modpad:
        j+=1
        rp[j-1]=np.array(i)
        tyc[j-1]=tyce[k][0]
        o[j-1]=tyce[k][0]-sum(np.multiply(dlzkky,np.array(i)))
```

Zdroj: Vlastné spracovanie

Obr. 8: Deklarácia premenných matematického modelu

```
#deklarácia premenných matematického modelu
x = [m.add_var(name='x', var_type = "I", lb=0) for i in PR]
#definovanie účelovej funkcie
m.objective = minimize(xsum(o[i]*x[i] for i in PR)
#definovanie štruktúrnych ohraničení
for j in V:
    m += PS[j]-xsum(rp[i][j]*x[i] for i in PR) == 0
for j in T:
    m += tyce[j][1]-xsum(x[i] for i in PR if tyce[j][0]==tyc[i]) >= 0
m.optimize()
```

Zdroj: Vlastné spracovanie

6 Riešenie problému rezania železných armatúrnych tyčí ako úlohy lineárneho programovania v jazyku Python

Pri riešení uvedeného problému sú využité ako vstupné údaje dva excelovské súbory. Zadané dát v prvom súbore reprezentuje tabuľka 4, kde užívateľ zadáva dĺžky a počty tyčí, ktoré budú použité pri výrobe požadovaných výsledných tyčí, pričom v tabuľke 5 je zobrazenie požiadaviek reprezentované požadovanou dĺžkou a ich počtom.

Ako je uvedené v časti 4, po spustení programu v jazyku Python budú načítané údaje z MS Excelu, pričom následne stanoví všetky prípustné rezné plány a vyrieši formulovanú optimalizačnú úlohu.

Výsledky uvedenej úlohy sú exportované do súboru MS Excel, ktorý formát výstupu je uvedený v tabuľke 6.

Tab. 4: Počet disponibilných tyčí

Tyče	Počet
200	500
230	200
250	100

Zdroj: Vlastné spracovanie

Tab. 5: Požiadavky na rezanie

Požiadavky	Počet
120	100
90	130
30	320
50	200
45	120
100	200
150	140

Zdroj: Vlastné spracovanie

V našom prípade, ako výsledok optimalizácie vzniká rezný plán, ktorého celkový odpad je 230. V Tabuľke 6 sú uvedené možné rezné plány.

Tab. 6: Rezné plány

	Dĺžka tyče	Počet ks	Odpad na 1 ks	120 cm-ks	90 cm-ks	30 cm-ks	50 cm-ks	45 cm-ks	100 cm-ks	150 cm-ks
1. spôsob	200	28	5	0	1	2	0	1	0	0
2. spôsob	200	77	0	1	0	1	1	0	0	0
3. spôsob	230	50	0	0	0	1	0	0	2	0
4. spôsob	230	62	0	0	0	1	1	0	0	1
5. spôsob	230	37	0	0	1	0	1	2	0	0
6. spôsob	230	24	0	0	2	0	1	0	0	0
7. spôsob	230	18	5	1	0	2	0	1	0	0
8. spôsob	250	78	0	0	0	0	0	0	1	1
9. spôsob	250	17	0	0	1	2	0	0	1	0
10. spôsob	250	5	0	1	0	1	0	0	1	0
OptimizationStatus,OPTIMAL										
Celkový odpad 230,0										

Zdroj: Vlastné spracovanie

Celkovo na získanie požadovaného počtu tyčí je potrebné realizovať 10 spôsobov rezania tyčí jednotlivých dĺžok, pričom sa využijú všetky druhy tyčí (dĺžky 200, 230, 250). Celkový odpad je 230, pričom najväčší odpad z jednej tyče je 5 a bude realizovaný pri 1. a 7. reznom pláne.

7 Tvorba informačného systému s využitím dátového skladu

Prezentovaný spôsob tvorby rezného plánu môže byť základom pre vytvorenie automatizovaného informačného systému firmy RB, Veľké Kapušany s.r.o., ktorý bude na základe požiadaviek z jednotlivých objednávok zbierať údaje o počte tyčí v každej disponibilnej dĺžke a bude pre každú objednávku poskytovať hotový rezný plán pre všetky požadované priemery. Výhodou komplexného informačného systému je spojenie viacerých objednávok v obmedzenom časovom intervale a tým aj celkové zníženie strát z nakupovaných tyčí.

Základom nového informačného systému (IS), ktorý môže byť integrovaný do informačného systému využívaného danou firmou, je databázový systém (DBS) rozvinutý do dátového skladu (DW). Okrem údajov o jednotlivých materiáloch a ich výskyte na sklade sa budú uchovávať aj časové a priestorové informácie o možnostiach nákupu nových materiálov. Taktiež budú v systéme automaticky uchovávané údaje o odrezkoch jednotlivých tyčí. Tiež bude možné zoskupiť dané odrezky do skupín podľa dĺžky a priemeru. Takto získané údaje o disponibilných materiáloch bude možné znovu zaradiť do výpočtu optimálneho rezného plánu. Okrem toho v dátovom sklade možno uchovávať už získané rezné plány a na ich základe dekomponovať riešenie úlohy na skupinu menších riešení. Taktiež bude možné uchovávať aj údaje o ekonomických stratách pri využívaní jedného alebo iného rezného plánu.

Výhodou daného riešenia je využitie multidimenzionálnej dátovej kocky, ktorá umožní mnohostranný pohľad na získané výsledky rezných plánov, ekonomické veličiny realizácie jednotlivých rezných plánov, časové údaje o naplňaní skladu a postupu realizácie rezov. Využívaním jednotlivých operácií s kockou Slice, Dice, RollUp, DrillDown umožní podrobnú analýzu získaných výsledkov.

8 Záver

Príspevok vznikol ako reakcia na konkrétnu požiadavku firmy RB, Veľké Kapušany s.r.o., ktorá sa zaoberá rozdelením dlhých kovových tyčí rôznych rozmerov na požadované dĺžky v súlade s požiadavkami odberateľa (rezanie železných armatúrových tyčí pre potreby veľkých stavieb, konkrétne pre diaľničnú výstavbu.). Na základe objednávok a výrobných výkresov sú vytvorené štítky určené pre ohýbacie stroje. Autori v ňom zovšeobecniť skúsenosti nadobudnuté počas riešenia tejto úlohy s využitím nástrojov matematického programovania a programu Python.

Ako efektívny nástroj na riešenie uvedeného problému bol formulovaná matematický model (rezný problém), ktorý predstavuje zjednodušený opis reálneho systém rezania kovových tyčí. Formulovaný model možno pritom riešiť nástrojmi matematického programovania, podporenými zodpovedajúcimi softvérovými nástrojmi, konkrétne programom Python. Takýmto postupom možno relatívne presne kvantitatívne realizovať výber určitého riešenia z veľkého množstva možných riešení, ktoré je z hľadiska matematicky formulovaného cieľa najlepšie.

Najzaujímavejším problémom pritom nebolo riešenie skonštruovanej úlohy matematického programovania, ale konštrukcia vlastných rezných plánov, ktoré sú podkladom pre zostavenie zodpovedajúcej úlohy matematického programovania. Zostavenie rezných plánov pri rozsiahlych zadaniach v sebe zahŕňa veľké množstvo variantov, ktoré možno eliminovať pomocou vhodne formulovaných pravidiel a využiť pritom tiež program Python.

Na základe riešenia príslušnej úlohy matematického programovania možno vytvoriť dátovú štruktúru – dátový sklad (DW), ktorá umožní uchovávať nielen údaje o materiáloch na sklade, výsledných produktov, ale aj jednotlivé vypočítané riešenia. DW taktiež uchováva údaje o zostatkoch materiálu. DW môže poskytnúť mnohorozmerný pohľad na získané údaje, čo môže byť predpokladom pre efektívnejšie využívanie pracovného času a materiálu.

Príspevok bol spracovaný v rámci riešenia grantovej úlohy VEGA 1/0339/20 „Hidden Markov Model Utilization in Financial Modeling“.

Literatúra

- [1] Ágoston, K. C. (2019). The effect of welding on the one-dimensional cutting-stock problem: The case of fixed firefighting systems in the construction industry. *Advances in Operations Research*, 2019, 1-12.
- [2] Anderson, D. R., Sweeney, D. J., Williams, T. A., Camm, J. D. & Cochran, J. J. (2018). *An introduction to management science: quantitative approach*. Cengage learning.
- [3] Brezina, I. & Pekár, J. (2019). *Úvod do operačného výskumu I*. Bratislava: LetraEdu.
- [4] Furini, F., Malaguti, E. & Thomopulos, D. (2016). Modeling Two-Dimensional Guillotine Cutting Problems via Integer Programming. *INFORMS Journal on Computing*, 28(4), 736-751.
- [5] Jahromi, M. H., Tavakkoli-Moghaddam, R., Makui, A. & Shamsi, A. (2012). Solving an one dimensional cutting stock problem by simulated annealing and tabu search. *Journal of Industrial Engineering International*, 8.1, 1-8.
- [6] Kellerer, H., Pferschy, U. & Pisinger, D. (2004). *Knapsack Problems*. Berlin Heidelberg: Springer-Verlag.
- [7] Malaguti, E., Durán, R. M. & Toth, P. (2014). *Approaches to real world two-dimensional cutting problems*. *Omega*, 47, 99-115.
- [8] Tanir, D., Ugurlu, O., Guler, A., & Nuriyev, U. (2016). One-dimensional Cutting Stock Problem with Divisible Items. ArXiv, abs/1606.01419.

- [9] Tanir, D., Ugurlu, O., Guler, A., & Nuriyev, U. (2019). One-dimensional cutting stock problem with divisible items: a case study in steel industry. *TWMS Journal of Applied and Engineering Mathematics*, 9(3), 473-484.