

Metody lineárního programování

Do Dang An

Bakalářská práce
2025



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně

Fakulta aplikované informatiky

Ústav informatiky a umělé inteligence

Akademický rok: 2024/2025

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: Dang An Do
Osobní číslo: A22659
Studijní program: B0613A140020 Softwarové inženýrství
Forma studia: Prezenční
Téma práce: Metody lineárního programování
Téma práce anglicky: Linear Programming Methods

Zásady pro vypracování

- Seznamte se s historií, základními principy a metodami řešení úloh lineárního programování. Zaměřte se na různé typy úloh, jejich strukturu a klasifikaci, například úlohy s omezením, bez omezení, maximalizační a minimalizační.
- Řešte úlohy lineárního programování pomocí simplexové metody, zejména s jednofázovým a dvoufázovou verzí. Zaměřte se na klíčové kroky, jako je výběr pivotu, aktualizace tabulky a kritéria optimality.
- Znázorněte výsledky simplexových metod pomocí grafů, které zachycují průběh řešení.
- Implementujte metody lineárního programování v softwaru, které zvládnou řešit úlohy. Program by měl umožňovat uživatelsky přívětivé zadávání vstupních dat a poskytovat detailní výstupy včetně vizualizace postupu řešení.
- Napište studijní oporu a vypracujte ilustrativní příklady.

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

1. JABLONSKÝ, Josef, 2002. Operační výzkum kvantitativní modely pro ekonomické rozhodování. Praha: Professional publishing. ISBN 80-86419-23-1.
2. LINDA, Bohdan, VOLEK, Josef, 2009. Lineární programování. Pardubice: Univerzita Pardubice. ISBN 978-80-7395-207-5
3. LUENBERGER, David G., YE, Yinyu, 2008. Linear and nonlinear programming. New York: Springer. ISBN 978-0387-74502-2
4. PLESNÍK, Ján, DUPAČOVÁ, Jitka, VLACH, Milan 1990. Lineárne programovanie. Bratislava: Alfa. ISBN 80-05-00679-9
5. PEKAŘ, Libor, 2013. Optimalizace, studijní materiály, přednášky. Zlín: FAI Univerzita Tomáše Bati.
6. PROKOP, Roman. Lineární programování. Zlín: FAI Univerzita Tomáše Bati.

Vedoucí bakalářské práce: **prof. Ing. Roman Prokop, CSc.**
Ústav matematiky

Datum zadání bakalářské práce: **27. října 2024**
Termín odevzdání bakalářské práce: **2. června 2025**

doc. Ing. Jiří Vojtěšek, Ph.D. v.r.
děkan



prof. Mgr. Roman Jašek, Ph.D., DBA v.r.
ředitel ústavu

Ve Zlíně dne 17. listopadu 2024

PROHLÁŠENÍ AUTORA BAKALÁŘSKÉ PRÁCE

Beru na vědomí, že

- odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb., v platném znění bez ohledu na výsledek obhajoby;
- bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému a bude dostupná k nahlédnutí;
- jedno vyhotovení bakalářské práce v listinné podobě bude ponecháno Univerzitě Tomáše Bati ve Zlíně k uložení;
- na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- podle § 60 odst. 1 autorského zákona má Univerzita Tomáše Bati ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- podle § 60 odst. 2 a 3 mohu užít své dílo – bakalářskou práci – nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tj. k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá; neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji, že

- jsem na bakalářské práci pracoval(a) samostatně a použitou literaturu jsem řádně citoval(a); v případě publikace výsledků budu uveden(a) jako spoluautor;
- odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou obsahově totožné.
- Prohlašuji, že při tvorbě této práce jsem použil/a nástroj generativního modelu AI Chat GPT 4o; <https://chatgpt.com> za účelem jazykové úpravy textu, návrhu vhodnějších formulací a pomoci s programováním. Po použití tohoto nástroje jsem provedl/a kontrolu obsahu a přebírám za něj plnou zodpovědnost.

Ve Zlíně, dne

.....
podpis autora

ABSTRAKT

Bakalářská práce se zabývá řešením úloh lineárního programování pomocí simplexové metody. V první části jsou popsány teoretické základy potřebné pro porozumění dané metodě a jejím variantám. Následující část práce je zaměřena na návrh a implementaci softwarového nástroje v jazyce Python, který umožňuje řešit úlohy lineární optimalizace a vizuálně sledovat postup výpočtu. Aplikace je doplněna o sadu praktických příkladů, které ověřují její funkčnost a ukazují možnosti využití v různých typech úloh. Výsledkem je nástroj, který může sloužit jako pomůcka pro výuku i samostatné studium základních principů lineárního programování.

Klíčová slova: lineární programování, klasifikace úlohy, simplexová metoda, dualita, redukované a stínové ceny, aplikace metod, Python.

ABSTRACT

The bachelor thesis deals with solving linear programming problems using the simplex method. The first part describes the theoretical foundations necessary for understand the method and its variants. The following part of the thesis focuses on the design and implementation of a software tool in Python that allows to solve linear optimization problems and visually monitor the calculation process. The application is complemented by a set of practical examples that verify its functionality and show the possibilities of its use in different types of tasks. The result is a tool that can serve as a learning aid for education and self-study of the basic principles of linear programming

Keywords: linear programming, problem classification, simplex method, duality, reduced and shadow prices, method applications, Python.

Chtěl bych poděkovat vedoucímu své bakalářské práce, prof. Ing. Romanu Prokopovi, CSc., za jeho čas, ochotu a trpělivé vedení. Velmi si vážím jeho podpory a cenných rad, které mi pomohly práci dotáhnout do konce. Děkuji také své rodině za trpělivost a neustálou podporu v průběhu celého studia. Zvláštní poděkování patří mé sestře za její pomoc a rady při přípravě práce. V neposlední řadě děkuji svým blízkým přátelům za to, že při mně stojí.

Prohlašuji, že odevzdaná verze bakalářské/diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

OBSAH

ÚVOD.....	11
I TEORETICKÁ ČÁST	12
1 TEORETICKÝ ZÁKLAD LINEÁRNÍHO PROGRAMOVÁNÍ.....	13
1.1 HISTORIE A VÝVOJ LINEÁRNÍHO PROGRAMOVÁNÍ.....	13
1.2 STRUKTURA ÚLOHY A ZÁKLADNÍ POJMY	14
1.2.1 Přehled základních pojmů	15
1.3 KLASIFIKACE ÚLOH LP	16
1.3.1 Typy optimalizace	16
1.3.2 Typy nerovnosti	16
1.3.3 Podmínky nezápornosti	17
1.3.4 Geometrická interpretace	17
1.3.5 Základní a přípustná řešení	21
2 METODY ŘEŠENÍ ÚLOH LINEÁRNÍHO PROGRAMOVÁNÍ.....	23
2.1 SIMPLEXOVÁ METODA.....	23
2.1.1 Princip simplexové metody	23
2.1.2 Simplexová tabulka	24
2.2 JEDNOFÁZOVÁ SIMPLEXOVÁ METODA	25
2.2.1 Výchozí stav jednofázové simplexové tabulky	25
2.2.2 Test optimality	25
2.2.3 Výběr klíčového (pivovního) prvku:.....	26
2.2.4 Přepočítání simplexové tabulky	27
2.2.5 Řešení příkladu.....	27
2.3 DVOUFÁZOVÁ SIMPLEXOVÁ METODA.....	31
2.3.1 Výchozí stav dvoufázové simplexové tabulky.....	31
2.3.2 Ověření přípustnosti řešení (I. fáze).....	32
2.3.3 Optimalizace původní účelové funkce (II. fáze)	32
2.3.4 Řešení příkladu.....	32
2.4 DUALITA V LP	39
2.4.1 Symetrická duální úloha.....	39
2.4.2 Nesymetrická duální úloha.....	41
2.5 ODEČTENÍ SIMPLEXOVÉ TABULKY	43
2.5.1 Stínové ceny	44
2.5.2 Redukované ceny	45
2.6 MOŽNOSTI ZAKONČENÍ ÚLOH LP	47
2.6.1 Úloha s jediným optimálním řešením	47
2.6.2 Úloha má alternativní optimální řešení	47
2.6.3 Úloha nemá omezenou hodnotu účelové funkce	50
2.6.4 Úloha nemá přípustné řešení	50
II PRAKTICKÁ ČÁST	51

3	STUDIJNÍ OPORA	52
3.1	ILUSTRATIVNÍ ŘEŠENÉ PŘÍKLADY	52
3.1.1	Maximalizační úloha s omezeními ve tvaru $\leq$	53
3.1.2	Minimalizační úloha s omezeními ve tvaru $\geq$	56
3.1.3	Maximalizační úloha s omezeními ve tvaru $=$	60
3.1.4	Maximalizační úloha s kombinovanými omezeními.....	65
3.1.5	Minimalizační úloha s kombinovanými omezeními	67
4	APLIKACE METOD V PRAXI	69
4.1	PROBLÉM ALOKACE ZDROJŮ	69
4.2	NUTRIČNÍ PROBLÉM.....	72
4.3	OPTIMALIZACE PORTFOLIA	76
4.4	DĚLENÍ MATERIÁLŮ.....	78
4.5	DOPRAVNÍ PROBLÉM	82
5	IMPLEMENTACE METODY V SOFTWARE	86
5.1	PŘEHLED EXISTUJÍCÍCH SOFTWAREVÝCH NÁSTROJŮ PRO LP	86
5.1.1	GAMS	86
5.1.2	MS Excel Řešitel	88
5.2	IMPLEMENTACE VLASTNÍHO SOFTWAREVÉHO ŘEŠENÍ	91
5.2.1	Programovací jazyk Python.....	91
5.2.2	Výchozí tvar simplexové tabulky	91
5.2.3	Nalezení klíčového prvku.....	93
5.2.4	Nalezení klíčového prvku.....	94
5.2.5	Přepočítání simplexové tabulky.....	94
5.2.6	Jednofázová a dvoufázová metoda.....	95
5.2.7	Pomocné funkce	96
5.2.8	Grafické rozhraní.....	96
5.2.9	Grafické řešení v GUI	98
5.2.10	Zadávatel a výstup v GUI.....	99
5.2.11	Dualita v GUI	100
5.2.12	Chybové hlášení	101
	ZÁVĚR.....	102
	SEZNAM POUŽITÉ LITERATURY	103
	SEZNAM OBRÁZKŮ	106
	SEZNAM TABULEK.....	107
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	109
	SEZNAM PŘÍLOH.....	110

ÚVOD

Lineární programování představuje důležitou součást operačního výzkumu a matematického modelování. Umožňuje nalézt optimální řešení úloh, které jsou omezené určitými podmínkami. Díky své jednoduchosti se lineární programování uplatňuje v celé řadě oblastí, jako je průmysl, doprava, ekonomie, logistika nebo řízení zdrojů.

Cílem této práce je seznámit se s historickým vývojem, základními principy a metodami řešení úloh lineárního programování. Hlavní důraz je kladen na simplexovou metodu, a to jak ve její jednofázové, tak i dvoufázové variantě. Součástí práce je podrobný rozbor klíčových kroků metody, ale také návrh a realizace vlastního softwarového nástroje, který slouží k vizualizaci a řešení těchto úloh. Ten je doplněn o konkrétní příklady z různých oblastí, na nichž je demonstrována funkčnost metody i samotného nástroje.

Práce je navržena tak, aby byla přístupná i čtenářům bez zkušeností s touto problematikou a usnadnila pochopení principů lineárního programování.

I TEORETICKÁ ČÁST

1 TEORETICKÝ ZÁKLAD LINEÁRNÍHO PROGRAMOVÁNÍ

Lineární programování (LP) je metoda operačního výzkumu, která se zaměřuje na rozhodování v situacích, kde je cílem určit optimální způsob realizace procesů za daných podmínek. Typicky jde o situace, kdy je třeba alokovat omezené zdroje tak, aby bylo dosaženo co nejlepšího výsledku vzhledem k definovanému cíli. Termín „programování“ zde neoznačuje programování v informatickém smyslu, ale plánování nebo tvorbu scénářů pro budoucí činnosti a „lineární“ pak odkazuje na lineární vztahy mezi proměnnými v matematickém modelu. Lineární programování tak slouží jako prostředek pro efektivní plánování a řízení procesů s důrazem na optimalizaci. [1]

1.1 Historie a vývoj lineárního programování

První základy LP vznikly už na konci 30. let 20. století díky sovětskému matematikovi Leonidu Vitalijeviči Kantorovičovi. Ten se zabýval využitím matematických metod při řešení ekonomických problémů. Formuloval první teoretické základy LP v ekonomické analýze a plánování, za což byl později oceněn Nobelovou cenou za ekonomii. Jeho práce, přestože byla mimo Sovětský svaz dlouho přehlížena, znamenala důležitý krok pro další rozvoj tohoto oboru. K většímu rozšíření LP přispěla druhá světová válka, kdy se tyto metody začaly využívat například v dopravě, při plánování výroby a rozdělování omezených zdrojů. [2][3]

K zásadnímu posunu došlo až v roce 1947, kdy americký matematik George Bernard Dantzig představil simplexovou metodu. Díky ní bylo možné efektivně řešit velké množství úloh LP. Tato metoda se brzy stala hlavním nástrojem pro optimalizaci v oblastech, jako je plánování výroby, rozvrhování pracovníků, sestavování investičních portfolií nebo tvorba marketingových a vojenských strategií. Kromě samotného algoritmu Dantzig také popsal obecnou matematickou strukturu LP, čímž položil pevné základy celého oboru. LP se tak stalo nepostradatelným nástrojem v teoretické i aplikované matematice a v řadě praktických oblastí. [2][4]

V České republice a na Slovensku se problematika lineárního programování vyučuje na mnoha vysokých školách. Mezi naše významné představitele v oblasti LP patří Ján Plesník, Bohdan Linda, Josef Volek a zejména Josef Jablonský, z jehož odborné literatury jsem při zpracování této práce čerpal nejvíce.

1.2 Struktura úlohy a základní pojmy

V následující části jsou definovány základní termíny související s LP.

Základní formulace účelové funkce i omezující vazební podmínky jsou vyjádřeny lineárními vztahy s konstantními koeficienty u jednotlivých proměnných i s konstantními pravými stranami soustavy omezení. Tyto vztahy lze přehledně vyjádřit také v maticovém zápisu, který se v praxi běžně používá. [5]

Základní formulace:

maximalizovat (minimalizovat)

$$f = \sum_{j=1}^n c_j x_j = c_1 x_1 + c_2 x_2 + \dots + c_n x_n \Leftrightarrow c^T x, \quad x \in \mathbb{R}^n \quad (1.1)$$

Vazební podmínky¹:

$$\sum_{j=1}^n a_{ij} x_j \leq b_i = \begin{matrix} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \leq b_1 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n \leq b_m \end{matrix} \Leftrightarrow Ax \leq b \quad (1.2)$$

Podmínky nezápornosti:

$$x_j \geq 0 \Leftrightarrow x \geq 0, \quad j = 1, 2, \dots, n \quad (1.3)$$

kde:

n – počet strukturních proměnných modelu,

m – počet vlastních omezení,

$c_j, j = 1, 2, \dots, n$ – cenový koeficient příslušející j -té proměnné,

$b_i, i = 1, 2, \dots, m$ – hodnotu pravé strany příslušející i -tému vlastnímu omezení,

$a_{ij}, i = 1, 2, \dots, m, j = 1, 2, \dots, n$ – strukturní koeficient,

$c^T = (c_1, c_2, \dots, c_n)$ – n -složkový řádkový vektor koeficientů,

$x = (x_1, x_2, \dots, x_n)$ – n -složkový sloupcový vektor strukturních proměnných modelů,

A – matice strukturních koeficientů o rozměru $m \times n$,

$b = (b_1, b_2, \dots, b_m)$ – m -složkový sloupcový vektor hodnot pravé strany.

¹ Omezení mohou být i jiného typu, tedy kombinace $\leq, =, \geq$.

Na základě shodného obsahu uvedeného ve zdrojích [1], [6], [7].

1.2.1 Přehled základních pojmů

Účelová funkce (1.1) – funkce, kterou chceme optimalizovat;

Strukturní (rozhodovací) proměnné – proměnné, o jejichž hodnotách rozhodujeme s cílem optimalizovat účelovou funkci;

Vlastní omezení (1.2) – požadavky na řešení, určují tvar přípustné množiny;

Podmínky nezápornosti (1.3) – požadavky, aby proměnné byly nezáporné;

Omezující podmínky – soustava vlastních omezení a podmínky nezápornosti;

Volná proměnná – proměnná, která není omezena na kladné nebo záporné hodnoty;

Přídavné proměnné² – slouží k transformaci vlastních omezení na soustavu rovnic, kde u nerovností typu $\leq$ se přičítá k levé straně, u nerovností typu $\geq$ se od levé strany odečítá;

Pomocné (umělé) proměnné – slouží k transformaci vlastních omezení typu $\geq a$ na soustavu rovnic přičtením k levé straně;

Kanonický tvar – formulace, kde jsou vlastní omezení zapsána ve formě rovnic a jsou doplněné o přídavné nebo pomocné proměnné;

Pomocná účelová funkce – účelová funkce sloužící k vyloučení pomocných proměnných;

Přípustné řešení – vektor $(x_1, x_2, \dots, x_n)$, který vyhovuje všem omezujícími podmínkám;

Přípustná množina – množina přípustných řešení všech bodů splňujících podmínky;

Konvexní polyedr (polygon)³ – konvexní útvar omezen ohraničenou množinou;

Simplexová metoda – iterační postup s cílem nalézt optimální řešení úloh LP;

Bazické (základní) řešení – hodnoty určené výpočtem ze soustavy.

Optimální řešení – přípustné řešení, které dosahuje nejlepší hodnoty účelové funkce;

Redukované ceny – koeficienty strukturních proměnných, vyjadřují míru výhodnosti zařazení jednotlivých procesů do optimálního řešení;

Stínové ceny (duální hodnoty) – koeficienty omezujících podmínek, vyjadřují změnu hodnoty účelové funkce při změně pravé strany o jednotku;

² Vyjadřují nevyužitou kapacitu ($\leq$) nebo překročení požadavku ($\geq$).

³ Polygon je rovinný útvar, polyedr je jeho prostorová obdoba.

Na základě shodného obsahu uvedeného ve zdrojích [1], [7], [8], [9].

1.3 Klasifikace úloh LP

Úlohy LP lze rozdělit podle různých kritérií, každé z nich určuje specifické vlastnosti zadání i způsob řešení.

1.3.1 Typy optimalizace

Podle typu optimalizace se úlohy LP dělí na **maximalizační** a **minimalizační**. Tento rozlišující typ ovlivňuje jak samotnou formulaci účelové funkce, tak i způsob řešení.

U maximalizačních úloh je cílem najít takové přípustné řešení, které přináší nejvyšší možnou hodnotu účelové funkce. Typickým příkladem je situace, kdy podnik usiluje o maximalizaci zisku, produkce nebo jiné veličiny. Účelová funkce v tomto případě zachycuje hodnotu, kterou se snažíme zvýšit na maximum.

Naopak u minimalizačních úloh je cílem hledání přípustného řešení, které vede k nejmenší možné hodnotě účelové funkce. Tento typ zadání se v praxi vyskytuje například při optimalizaci nákladů, spotřeby materiálu nebo minimalizaci rizika.

Maximalizační i minimalizační úlohu lze vzájemně převést jednoduchou úpravou účelové funkce, kdy se změní znaménka všech jejích koeficientů. Maximalizace funkce $f(x)$ je ekvivalentní minimalizaci funkce $-f(x)$ a naopak, minimalizace funkce $f(x)$ je ekvivalentní maximalizaci funkce $-f(x)$.⁴ [10][11]

$$\max f(x) = \min(-f(x)) \Leftrightarrow \min f(x) = \max(-f(x)) \quad (1.4)$$

1.3.2 Typy nerovností

V úlohách LP se setkáváme se třemi základními typy nerovností, mezi které patří nerovnost typu „menší nebo rovno“ ($\leq$), „větší nebo rovno“ ($\geq$) a rovnost ($=$).

Každý typ nerovnosti vyžaduje úpravu pomocí přídatných nebo pomocných proměnných, aby bylo možné úlohu převést do **kanonického tvaru**. Přídavné proměnné se přičítají se u nerovností typu $\leq$ (tzv. slack), nebo odečítají u typu $\geq$ (tzv. surplus). V případech, kdy rovnice neumožňuje přímo stanovit základní řešení (zejména u $\geq$ a $=$), se zavádějí pomocné proměnné, které dočasně vstupují do výpočtu. [1][10]

⁴ Tento převod je obecný, nejen v rámci LP.

Shrnutí způsobu doplňování proměnných pro všechny typy nerovnosti uvádí Tab. 1.

Tab. 1: Doplňování přídatných a pomocných proměnných

Typ omezení	Přídatná proměnná	Pomocná proměnná
$\leq$	$+x$	
$\geq$	$-x$	$+y$
$=$		$+y$

(zdroj:[1])

1.3.3 Podmínky nezápornosti

V lineárním programování se většina proměnných omezuje podmínkou nezápornosti, tedy $x_j \geq 0$. Tento požadavek vychází z praktických situací, kdy proměnné reprezentují například množství produkce, dobu trvání či jiné ekonomické veličiny, které nemohou nabývat záporných hodnot. Podmínky nezápornosti jsou nedílnou součástí kanonického tvaru úlohy a jsou nezbytné pro aplikaci simplexové metody, která pracuje výhradně s nezápornými hodnotami ve vektoru základního řešení.

Ne všechny proměnné v lineárním modelu musí být nutně nezáporné. Pokud není proměnná omezena na nezáporné hodnoty, označuje se jako **volná proměnná** (*Free variable* nebo *unrestricted in sign*, zkráceně URS). Takovou proměnnou lze matematicky převést na rozdíl dvou nezáporných proměnných:

$$x_j = x_j^+ - x_j^-, \quad x_j^+, x_j^- \geq 0 \quad (1.5)$$

Tento převod umožňuje zahrnout volnou proměnnou do standardních výpočetních postupů bez porušení podmínek nezápornosti. [9]

1.3.4 Geometrická interpretace

Optimální řešení úlohy LP, pokud existuje, se vždy nachází v množině přípustných bodů (*feasible region*), která je definována soustavou omezujících podmínek. V rámci analýzy řešitelnosti lze rozlišit čtyři základní možnosti zakončení úlohy, které charakterizují zakončení optimalizačního procesu. [1][6]

Pro názornou a teoretickou ukázkou bude v dalších případech uvažováno pouze s dvěma rozhodovacími proměnnými, což umožňuje přehledné geometrické znázornění řešení v rovině.

Ukázkový příklad byl převzat ze zdroje [1].

$$0,5x_1 + 0,25x_2 \leq 40 \quad (1.6)$$

$$0,5x_1 + 0,5x_2 \leq 60$$

$$0,25x_2 \leq 25$$

$$x_j \geq 0$$

Průsečíky přímek s osami souřadnic lze zjistit tak, že jednu z proměnných položíme rovnu nule a druhou dopočítáme jako podíl pravé strany a koeficientu této proměnné, tedy tak, aby odpovídající výraz měl hodnotu 1. Výsledné hodnoty představují průsečíky dané přímky s osami soustavy souřadnic. Takto definovaná přímka rozděluje rovinu na dvě poloroviny⁵.

Shrnutí vypočtených průsečíků uvádí Tab. 2.

Tab. 2: Průsečíky jednotlivých omezení s osami soustavy souřadnic

	$x_2 = 0$	$x_1 = 0$
1. omezení	$x_1 = 80$	$x_2 = 160$
2. omezení	$x_1 = 120$	$x_2 = 120$
3. omezení	$x_1 = 0$	$x_2 = 100$

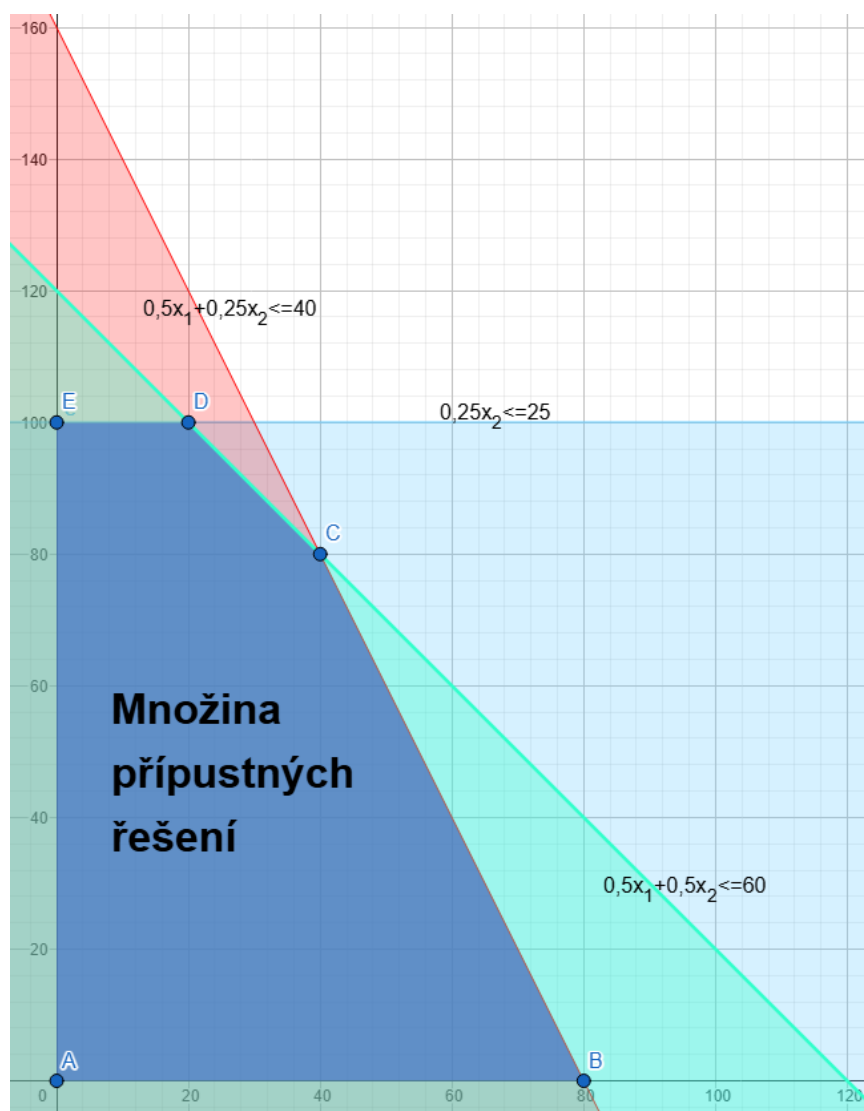
(zdroj: vlastní)

V našem příkladu je množina přípustných řešení určena třemi vlastními omezeními a podmínkami nezápornosti, kde podmínkám nezápornosti vyhovuje v rovině první kvadrant. Množina přípustných řešení je tedy průnikem prvního kvadrantu a tří polorovin odpovídajících třem vlastním omezením.

Tento průnik je znázorněn na Obrázek 1, množina přípustných řešení tvoří konvexní polygon (nepravidelný pětiúhelník) označený vrcholy *ABCDE*.

⁵

U nerovnic typu $\leq$ polorovina obsahuje počátek $[0, 0]$.
U nerovnic typu $\geq$ polorovina neobsahuje počátek.

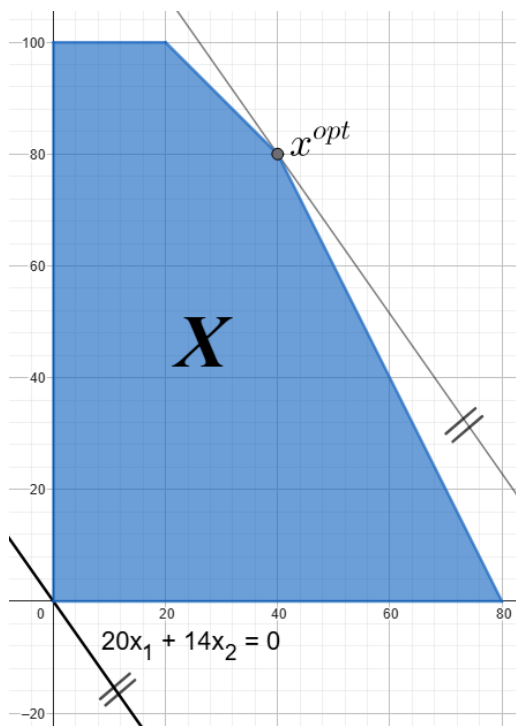


Obrázek 1: Množina přípustných řešení příkladu (1.6)

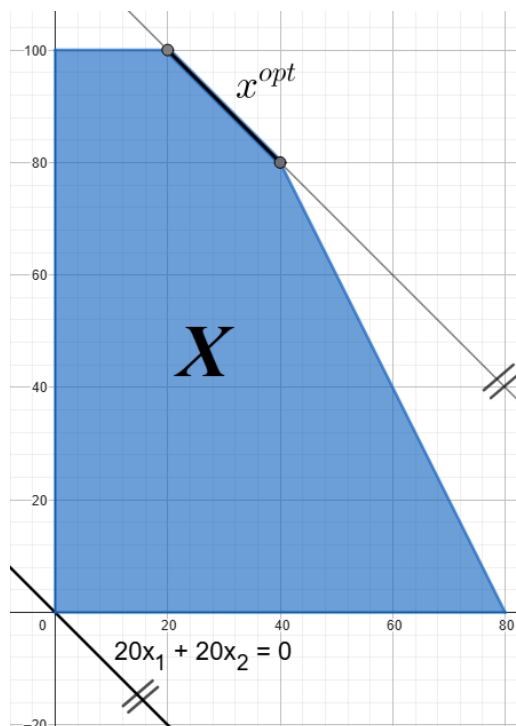
(zdroj: vlastní)

Jak již bylo uvedeno, analýza řešitelnosti úlohy LP rozlišuje čtyři základní způsoby, jimiž může být optimalizační proces ukončen. Tyto případy budou podrobněji rozebrány v následující kapitole věnované simplexové metodě. Pro určení typu zakončení se v každém případě zakresluje přímka účelové funkce ve tvaru $f = 0$, která znázorňuje průběh změn účelové funkce a umožňuje graficky určit místo, kde nabývá svého maxima (nebo minima) v rámci přípustné oblasti.

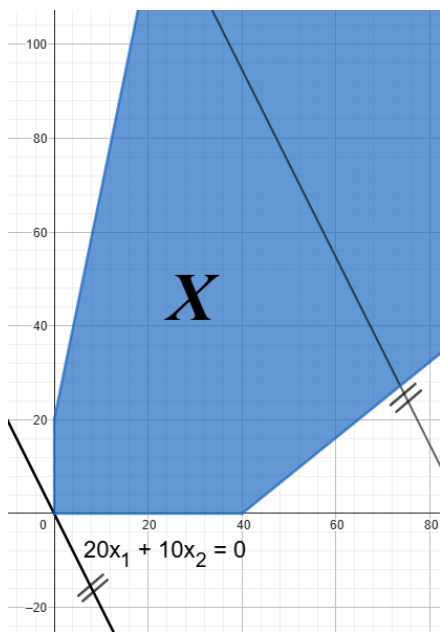
Přehledné grafické znázornění jednotlivých možností zakončení je uvedeno na Obrázek 2, kde je zvýrazněna množina přípustných řešení X a optimální řešení x^{opt} , pokud existuje, je vyznačeno tučně.



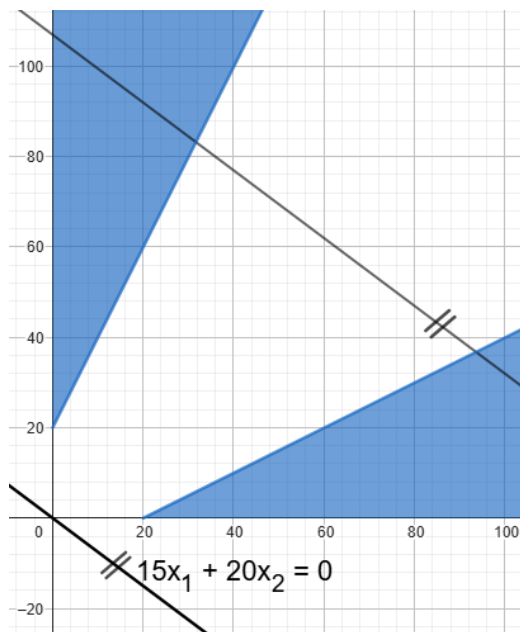
1. jediné optimální řešení



2. alternativní optimální řešení



3. neomezená hodnota účelová funkce



4. neexistuje přípustné řešení

Obrázek 2: Možnosti zakončení výpočtu LP

(zdroj:[1])

1.3.5 Základní a přípustná řešení

Na základě předchozího rozboru typů zakončení optimalizačního procesu lze konstatovat, že má-li úloha LP optimální řešení, má nutně základní optimální řešení. V případě, že úloha připouští více základních optimálních řešení, pak každá jejich konvexní kombinace představuje optimální řešení, což znamená, že úloha má nekonečně mnoho rovnocenných optimálních řešení. Z hlediska definice se **optimální řešení** chápe jako takové přípustné řešení, které maximalizuje nebo minimalizuje hodnotu účelové funkce. [5]

Aby bylo možné analyzovat vlastnosti základních řešení, je nezbytné, aby úloha byla převedena do **kanonického tvaru** $\tilde{A}\tilde{x} = b$, tedy do formy ekvivalentní soustavy rovnic doplněné o přídatné či pomocné proměnné.

Doplnění nerovnic na rovnice: $\tilde{A}\tilde{x} = b$ podle schématu⁶: [5]

$$\underbrace{\begin{pmatrix} a_{11} & \dots & a_{1n} & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ a_{m1} & \dots & a_{mn} & 0 & 0 & \dots & 1 \end{pmatrix}}_{\tilde{A}} \underbrace{\begin{pmatrix} x_1 \\ \dots \\ x_n \\ x_{n+1} \\ \dots \\ x_{n+m} \end{pmatrix}}_{\tilde{x}} = \underbrace{\begin{pmatrix} b_1 \\ \dots \\ b_m \end{pmatrix}}_b \quad (1.7)$$

kde:

$\tilde{A}$ – matice původních koeficientů rozšířená o jednotkovou podmatici, přidáním přídatných nebo pomocných proměnných,

$\tilde{x}$ – vektor rozhodovacích proměnných včetně nových přidanych,

$x_1, \dots, x_n$ – základní proměnné,

$x_{n+1}, \dots, x_{n+m}$ – nezákladní (přídatné, pomocné) proměnné.

Soustava rovnic $\tilde{A}\tilde{x} = b$ reprezentuje m rovnic s $(n + m)$ neznámými, má řešení právě tehdy, když je splněna podmínka rovnosti hodnoty matice koeficientů a hodnoty matice rozšířené, tj. platí:

$$\text{rank}(\tilde{A}) = \text{rank}(\tilde{A}, b)^7 \quad (1.8)$$

⁶ Schéma uvedené v rovnici (1.7) platí výhradně pro nerovnosti typu $\leq$. U jiných typů nerovností je nutné úpravy znamének nebo zavedení pomocných proměnných.

⁷ Hodnota matice udává maximální počet lineárně nezávislých řádků

Základní řešení vzniká volbou m bazických proměnných, zatímco zbývající n proměnných je nulových (nebazických)⁸. Je **přípustné**, pokud všechny hodnoty ve vektoru řešení jsou nezáporné. Jejich celkový počet lze vyjádřit kombinatorickým vztahem: [1][5]

$$\binom{m+n}{m} \quad (1.9)$$

Příklad (1.6) převedeme do kanonického tvaru a určíme všechna základní řešení jako kombinace dvou nebazických proměnných z celkového počtu proměnných.

Kanonický tvar příkladu (1.6):

$$0,5x_1 + 0,25x_2 + x_3 = 40 \quad (1.10)$$

$$0,5x_1 + 0,5x_2 + x_4 = 60$$

$$0,25x_2 + x_5 = 25$$

Tato soustava obsahuje 5 proměnných a 3 rovnice, soustava (1.10) má tedy maximálně

$$\binom{5}{3} = \frac{5!}{3!(5-3)!} = \frac{5!}{3!2!} = \frac{5 * 4}{2 * 1} = 10 \quad (1.11)$$

základní řešení, tj. počet způsobů, jak lze vybrat dvě nezákladní proměnné z celkového počtu pěti proměnných. Výsledky jsou uvedeny v Tab. 3.

Tab. 3: Základní řešení ekvivalentní soustavy rovnic

Řešení	x_1	x_2	x_3	x_4	x_5		Vrchol z Obrázek 1
x^1	0	0	40	60	25	přípustné	A
x^2	80	0	0	20	25	přípustné	B
x^3	40	80	0	0	5	přípustné	C
x^4	20	100	5	0	0	přípustné	D
x^5	0	100	15	10	0	přípustné	E
x^6	0	120	10	0	-5	nepřípustné	-
x^7	0	160	0	-20	-15	nepřípustné	-
x^8	30	100	0	-5	0	nepřípustné	-
x^9	120	0	-20	0	25	nepřípustné	-
x^{10}	-	0	-	-	0	Nemá řešení ⁹	-

(zdroj: [1])

⁸ Pokud je alespoň jedna základní proměnná rovna nule, označuje se jako **degenerované základní řešení**.

⁹ $0,25 * 0 + 0 = 25 \rightarrow 0 \neq 25$

2 METODY ŘEŠENÍ ÚLOH LINEÁRNÍHO PROGRAMOVÁNÍ

2.1 Simplexová metoda

Simplexová metoda představuje jeden z algoritmů pro nalezení optimálního řešení úlohy LP. Je založena na myšlence, že optimální hodnota účelové funkce je dosažena v některém z vrcholů (extrémních bodů) množiny přípustných řešení. Metoda postupně prochází mezi těmito body (bazickými) přípustnými řešeními a vybírá takové, které přinášejí zlepšení hodnoty účelové funkce, dokud není nalezeno optimum. [12]

Postup výpočtu pomocí simplexové metody lze rozdělit na dvě základní fáze:

- **I. fáze** – nalezení výchozího bazického řešení
- **II. fáze** – iterační postup vedoucí k optimalizaci účelové funkce

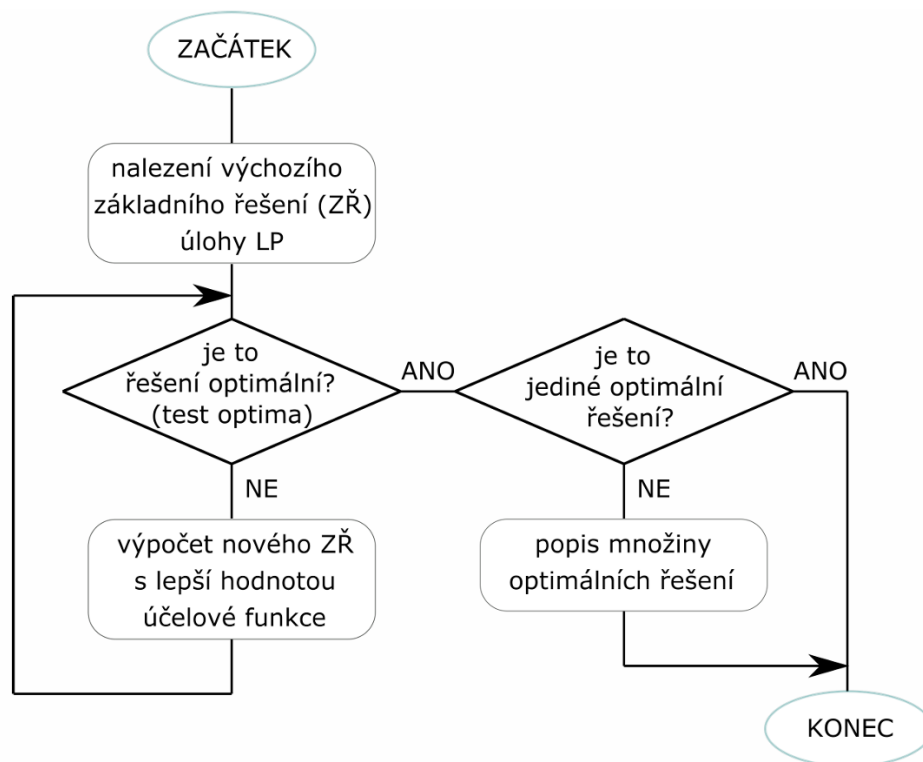
Pokud je nalezení výchozího základního řešení snadné, není třeba provádět první fázi výpočtu a používá se **jednofázová simplexová metoda**. V opačném případě, kdy výchozí řešení neexistuje nebo není zřejmé, se používá **dvoufázová simplexová metoda**. [1]

2.1.1 Princip simplexové metody

Pro aplikaci simplexové metody je nezbytné převést úlohu do **kanonického tvaru**. Tento tvar umožňuje snadnější provádění ekvivalentní řádkové úpravy, aniž by to mělo vliv na množinu přípustných řešení. Při zohlednění podmínek nezápornosti musí být zároveň splněno, že všechny prvky pravé strany rovnic (tj. složky vektoru b) jsou nezáporné.¹⁰ [5][13]

Metoda začíná určením výchozího základního řešení úlohy LP. Pokud takové řešení existuje, simplexová metoda v každém kroku za použití **simplexové tabulky** spočítá nové základní řešení, které v každém kroku posouvá řešení směrem k hodnotě účelové funkce, která odpovídá stanovenému cíli optimalizace. Jedná se o iterativní algoritmus, ve kterém každá další tabulka vzniká úpravou předchozí. Tento proces sleduje posun mezi jednotlivými vrcholy konvexního polyedru tvořeného množinou přípustných řešení. Po konečném počtu kroků výpočet buď nalezne základní řešení s nejlepší možnou hodnotou účelové funkce, nebo prokáže, že takové řešení neexistuje. [1]

¹⁰ Lze celou nerovnici vynásobit hodnotou -1 , čímž zároveň dojde ke změně směru nerovnosti.



Obrázek 3: Hrubé schéma simplexové metody

(zdroj:[7])

2.1.2 Simplexová tabulka

Simplexová tabulka v sobě spojuje nejen přehlednou formu výpočtů, ale zároveň umožňuje efektivní vyhledávání optimálního řešení úlohy LP. Matice omezení pak obsahuje jednotkovou submatici, jejíž sloupce určují bazické proměnné – tedy ty, které mají v počátečním řešení nenulové hodnoty. Dále je do posledního řádku tabulky přidán i řádek účelové funkce ve tvaru $f - c^T x$, jehož hodnoty slouží k rozhodování o pokračování výpočtu v dalších krocích algoritmu.

Tento poslední řádek představuje účelovou funkci ve své **anulované podobě**, kdy jsou všechny proměnné převedeny na levou stranu a na pravé straně zůstává pouze absolutní člen. Tento člen udává aktuální hodnotu účelové funkce (zpravidla na počátku rovnou nule). Obecný anulovaný tvar účelové funkce lze vyjádřit následovně: [1]

$$f - c_1 x_1 - c_2 x_2 - \dots - c_n x_n = 0 \quad (2.1)$$

Výchozí tvar simplexové tabulky je znázorněn v Tab. 4.

Tab. 4 Výchozí tvar simplexové tabulky

x_1	x_2	...	x_n	x_{n+1}	...	x_{n+m}	omezení
a_{11}	a_{12}	...	a_{1n}	1	0 ...	0	b_1
a_{21}	a_{22}	...	a_{2n}	0	1 ...	0	b_2
$\vdots$				$\vdots$			$\vdots$
a_{m1}	a_{m2}	...	a_{mn}	0	0	1	b_m
$-c_1$	$-c_2$	...	$-c_n$	0	0 ...	0	f

(zdroj: [2])

2.2 Jednofázová simplexová metoda

Jednofázová simplexová metoda se používá v případech, kdy jsou všechna omezení úlohy vyjádřena nerovnostmi typu $\leq$, protože ty lze přímo převést na soustavu rovnic v kanonickém tvaru doplněním přídatných proměnných.

2.2.1 Výchozí stav jednofázové simplexové tabulky

Celý výpočet je přehledně organizován pomocí simplexové tabulky, která umožňuje systematické sledování postupu řešení. Po převedení omezení úlohy do soustavy rovnic a doplnění přídatných proměnných je rovněž nutné upravit účelovou funkci do anulované podoby. V takovém zápisu se všechny její členy nacházejí na levé straně a ve sloupci pravých stran je hodnota nula. Následně je možné v jednotlivých iteracích postupně přecházet k novým řešením, která zlepšují hodnotu účelové funkce. [1]

2.2.2 Test optimality

V každém kroku výpočtu simplexovou metodou dochází ke zlepšení nebo zachování hodnoty účelové funkce. Toto zlepšení závisí na tzv. **redukovaných cenách** (koeficientech vstupujících proměnných v řádku účelové funkce). Pokud je alespoň jeden z těchto koeficientů záporný (v případě maximalizace), nebo kladný (v případě minimalizace), znamená to, že existuje možnost zlepšit hodnotu účelové funkce zavedením příslušné proměnné do báze. Pokud se v řádku účelové funkce nevyskytuje žádný zlepšující koeficient, pak základní řešení již nelze dále zlepšit a je tedy optimální. Řešení aktuální iterace se z tabulky odečítá tak, že nezákladním (eliminovaným) proměnným se přiřadí hodnota nula, zatímco hodnoty základních proměnných odpovídají hodnotám ve sloupci pravé strany. [1][2]

V opačném případě se určí **klíčový (pivotní) sloupec**, tedy proměnná s největším záporným (při maximalizaci), resp. největším kladným (při minimalizaci) koeficientem, která vstoupí do báze. [13]

Formálně platí¹¹: [1]

- při **maximalizaci** je řešení optimální, pokud všechny redukované ceny splňují $f_k \geq 0, k \in N$,
- při **maximalizaci** je klíčový sloupec $f_k = \min f_j, f_j < 0, j \in N$,
- při **minimalizaci** je řešení optimální, pokud všechny redukované ceny splňují $f_k \leq 0, k \in N$,
- při **minimalizaci** je klíčový sloupec $f_k = \max f_j, f_j > 0, j \in N$,

kde:

N je množina indexů nezákladních proměnných (redukované cenové koeficienty),

záporné hodnoty f_k signalizují možnost zvýšení hodnoty účelové funkce,

kladné hodnoty f_k signalizují možnost snížení hodnoty účelové funkce.

2.2.3 Výběr klíčového (pivotního) prvku:

Po výběru klíčového sloupce následuje určení **klíčového řádku**. Pro všechny kladné prvky ve zvoleném klíčovém sloupci se vypočítají podíly hodnot pravé strany a odpovídajících koeficientů, na jejichž základě se určuje klíčový řádek¹².

$$t = \frac{b_i}{a_{ik}}, \quad a_{ik} > 0 \quad (2.2)$$

kde a_{ik} je kladný prvek v klíčovém sloupci a b_i odpovídající hodnota pravé strany. Vybere se řádek s **nejmenším nezáporným podílem**, čímž se zajistí zachování nezápornosti nového řešení. Průsečíkem klíčového sloupce a řádku je potom **klíčový prvek**, který slouží jako výchozí prvek pro provedení elementárních úprav vedoucích k sestavení nové simplexové tabulky v rámci následující iterace algoritmu. [1]

¹¹ Pokud hodnota klíčového sloupce vychází shodně pro více sloupců, lze zvolit libovolně z nich.

¹² Pokud minimální podíl vychází shodně pro více řádků, lze zvolit libovolně z nich.

2.2.4 Přepočít simplexové tabulky

Pro přepočít simplexové tabulky se využívá standardní Gaussova eliminační metoda. Cílem přepočtu je dosáhnout toho, aby klíčový sloupec obsahoval jednotkový vektor s hodnotou 1 na pozici klíčového prvku a nulami na ostatních místech. Zároveň je nezbytné zachovat jednotkový tvar ostatních sloupců odpovídajících základním proměnným.

Přepočít probíhá ve dvou krocích:

1. **Transformace klíčového řádku** se provede tak, že klíčový řádek se vydělí klíčovým prvkem, čímž se na pozici tohoto prvku vytvoří jednička.
2. **Transformace i-tého řádku** (včetně řádku účelové funkce) probíhá tak, že transformovaný klíčový řádek (z 1. kroků) vynásobíme hodnotou $-a_{ik}$ a přičteme k příslušnému i -tému řádku, nebo v případě řádku účelové funkce použijeme hodnotu $-f_k$.

Po realizaci simplexové tabulky následuje opětovný test optimality pro ověření, zda bylo dosaženo optimálního řešení. [1]

2.2.5 Řešení příkladu

Algoritmus řešení úlohy jednofázové simplexové metody lze shrnout do následujících bodů:¹³

1. **Sestavení výchozí simplexové tabulky** – doplněné o přídatné proměnné, včetně řádku účelové funkce v anulovaném tvaru;
2. **Test optimality**: Pokud jsou všechny redukované ceny $f_k \geq 0$ (pro maximalizaci), resp. $f_k \leq 0$ (pro minimalizaci), pak konec a odečtení řešení;
3. **Výběr klíčového sloupce**: Zvolí se proměnná s největším záporným f_k (pro maximalizaci), resp. s největším kladným f_k (pro minimalizaci);
4. **Výběr klíčového řádku**: $\min. \beta_i = \frac{b_i}{a_{ik}}$ z nezáporných;
5. **Přepočít tabulky**: Eliminace klíčového prvku pomocí ekvivalentních maticových úprav;
6. **Opakování**: Postup se opakuje od bodu 2, dokud není splněna podmínka optimality;

¹³ Převzato a upraveno ze zdroje [5]

K ilustrativnímu příkladu (1.6) doplníme účelovou funkci a zapíšeme jako úplnou úlohu lineárního programování. [1]

$$\max f = 20x_1 + 14x_2 \quad (2.3)$$

$$\text{za podmíněk: } 0,5x_1 + 0,25x_2 \leq 40$$

$$0,5x_1 + 0,5x_2 \leq 60$$

$$0,25x_2 \leq 25$$

$$x_j \geq 0$$

Kánonický tvar (1.10):

$$0,5x_1 + 0,25x_2 + x_3 = 40$$

$$0,5x_1 + 0,5x_2 + x_4 = 60$$

$$0,25x_2 + x_5 = 25$$

Tab. 5: Výchozí simplexová tabulka příkladu (2.3)

	x_1	x_2	x_3	x_4	x_5	b_i
	0,5	0,25	1	0	0	40
	0,5	0,5	0	1	0	60
	0	0,25	0	0	1	25
f	-20	-14	0	0	0	0

(zdroj: [1])

Tab. 5 představuje výchozí tvar simplexové tabulky, která obsahuje přehled základních proměnných, matici strukturálních koeficientů, vektor pravých stran a anulovaný tvar účelové funkce. Přípustné řešení obsažené v této tabulce je charakterizováno vektorem $x^1 = (0; 0; 40; 60; 25)$ a hodnota účelové funkce tohoto řešení je $f^1 = 0$, přičemž toto řešení odpovídá prvnímu řádku Tab. 3.

Klíčovým sloupcem se stává sloupec proměnné x_1 , pro kterou je redukováná cena -20 . To znamená, že zvýšením hodnoty této proměnné o jednotku dojde k navýšení hodnoty účelové funkce o 20 jednotek. Pro srovnání, u proměnné x_2 by byl tento přírůstek pouze 14 jednotek.

Po určení klíčového sloupce následuje výběr klíčového řádku. Ten se stanoví výpočtem podílů mezi hodnotami pravé strany a odpovídajícími kladnými koeficienty ve zvoleném sloupci, přičemž se vybírá nejmenší z těchto nezáporných podílů. Pro klíčový sloupec x_1 se získají následující podíly¹⁴:

$$t = \min \left[\frac{40}{0,5}, \frac{60}{0,5} \right] = 80 \quad (2.4)$$

Tab. 6 zachycuje průběh výpočtu simplexové metody od výchozího tvaru až po nalezení optimálního řešení. V první iteraci byl jako klíčový prvek určen prvek 0,5, který se nachází v průsečíku klíčového sloupce x_1 a klíčového řádku. Tento prvek byl vybrán jako nejmenší z podílů $t = b_i/a_{ik}$ mezi hodnotami pravé strany a kladnými prvky ve zvoleném sloupci. Po první iteraci došlo ke zlepšení hodnoty účelové funkce na 1 600. V následující iteraci byl jako klíčový prvek zvolen prvek 0,25 a výsledná hodnota účelové funkce po druhé iteraci činí 1 920.

Detailní výpočetní postup jednotlivých iterací, včetně výběru klíčových prvků a transformace tabulky, bude podrobně rozebrán v praktické části práce.

Tab. 6: Řešení simplexové metody příkladu (2.3)

	x_1	x_2	x_3	x_4	x_5	b_i	$t = b_i/a_{ik}$
Výchozí tvar	0,5	0,25	1	0	0	40	80
	0,5	0,5	0	1	0	60	120
	0	0,25	0	0	1	25	-
f	-20	-14	0	0	0	0	
1. iterace	1	0,5	2	0	0	80	160
	0	0,25	-1	1	0	20	80
	0	0,25	0	0	1	25	100
f	0	-4	40	0	0	1 600	
2. iterace	1	0	4	-2	0	40	
	0	1	-4	4	0	80	
	0	0	1	-1	1	5	
f	0	0	24	16	0	1 920	

(zdroj:[1])

¹⁴ Podíl pro třetí řádek není uveden, neboť by znamenal dělení nulou, což je matematicky nedefinované.

Z Tab. 6 lze odečíst, že výsledkem první iterace je řešení $x^2 = (80; 0; 0; 20; 25)$ s hodnotou účelové funkce $f^2 = 1\,600$. Následnou iterací byl dosažen odpovídající vektor proměnných optimálního řešení $x^{opt} = (40; 80; 0; 0; 5)$ s maximální hodnotou účelové funkce $f^{opt} = 1\,920$. Obě tato řešení, stejně jako ostatní přípustné řešení x^4 a x^5 , odpovídají přípustným řešením uvedeným v Tab. 3. Protože žádné další řešení z této tabulky nepřináší vyšší hodnotu účelové funkce, není z pohledu optimalizace nutné v iteracích pokračovat.

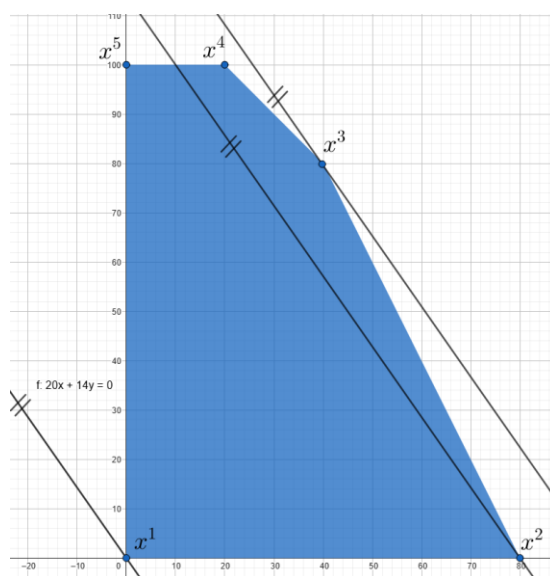
Důkazem je Tab. 7, ve které jsou po dosazení hodnot proměnných x_1 a x_2 do účelové funkce uvedeny odpovídající hodnoty funkce. Tato tabulka zároveň umožňuje přehledně posoudit, které z řešení dosahuje nejvyšší hodnoty účelové funkce.

Tab. 7: Výpočet hodnot účelové funkce pro jednotlivá přípustná řešení

Řešení	x_1	x_2	x_3	x_4	x_5	$f = 20x_1 + 14x_2$
x^1	0	0	40	60	25	0
x^2	80	0	0	20	25	1 600
x^3	40	80	0	0	5	1 920
x^4	20	100	5	0	0	1 800
x^5	0	100	15	10	0	1 400

(zdroj: vlastní)

Obrázek 4 znázorňuje průběh iterací simplexové metody, konkrétně postup přechodu mezi jednotlivými bazickými řešeními v rámci množiny přípustných řešení.



Obrázek 4: Grafické znázornění simplexové metody

(zdroj: vlastní)

2.3 Dvofázová simplexová metoda

Dvofázová simplexová metoda se uplatňuje v situacích, kdy omezení úlohy LP nejsou všechna vyjádřena ve formě nerovnic typu $\leq$, a proto nelze snadno získat výchozí základní řešení. Jeho nalezení tvoří samostatnou I. fázi výpočtu. Ve II. fázi pak dochází k optimalizaci původní účelové funkce a tato fáze již probíhá totožně s postupem popsáním u jednofázové metody.

Pro účely I. fáze je nutné rozšířit soustavu o pomocné proměnné, které slouží k vytvoření počátečního základního řešení v případě, kdy standardní jednotkové sloupce nelze vytvořit pomocí přídatných proměnných. Aby bylo možné tyto pomocné proměnné z výpočtu odstranit, zavádí se navíc **pomocná účelová funkce** f' , jejímž cílem je minimalizace součtu všech pomocných proměnných:

$$f' = \sum_i y_i \rightarrow \min = y_1 + \dots + y_i \rightarrow \min \quad (2.5)$$

Simplexová tabulka v I. fázi je tak rozšířena o pomocné proměnné a pracuje se s nově definovanou pomocnou účelovou funkcí a až po úspěšném odstranění všech pomocných proměnných je možné přejít k optimalizaci původní účelové funkce ve II. fázi výpočtu. [1]

2.3.1 Výchozí stav dvofázové simplexové tabulky

Při vytváření výchozího tvaru simplexové tabulky ve dvofázové metodě se úloha, podobně jako u jednofázové metody, převádí do soustavy rovnic. Jak již bylo uvedeno, výchozí tvar bude obsahovat nejen přídatné, ale i pomocné proměnné, jejichž zavedení závisí na konkrétním typu nerovnosti v omezeních.

Tab. 1 shrnuje způsob, jakým se určuje znaménko přídatných proměnných v závislosti na typu omezení. V případě nerovnosti typu $\geq$ je třeba přidat přídatnou proměnnou se záporným znaménkem (odebrat nadbytek) a současně pomocnou proměnnou s kladným znaménkem. V případě rovnosti se přídatná proměnná neuplatňuje a do systému se rovnou vkládá pomocná proměnná.

Pomocná účelová funkce se získá tak, že se ve všech rovnicích osamostatní pomocné proměnné, upravené rovnice se následně sečtou. Následným sečtením těchto rovnic vznikne již dříve definovaný výraz (2.5), který je dále převeden do anulovaného tvaru, aby mohl být zařazen do simplexové tabulky. [1]

2.3.2 Ověření přípustnosti řešení (I. fáze)

Po sestavení výchozí simplexové tabulky v I. fázi výpočtu probíhá minimalizace pomocné účelové funkce s cílem ověřit existenci přípustného řešení. Výpočet se řídí stejným postupem jako u jednofázové metody, pouze se optimalizuje pomocná účelová funkce f' a klíčový sloupec se určuje podle největšího kladného koeficientu f'_k .

Pokud se podaří minimalizovat hodnotu pomocné účelové funkce na $f' = 0$, pak vzhledem k nezápornosti všech proměnných musí být všechny pomocné proměnné rovny nule. To znamená, že bylo nalezeno výchozí přípustné řešení původní úlohy a výpočet může dále pokračovat ve II. fázi, tj. optimalizací původní účelové funkce. [1]

Naopak, je-li minimum $f' > 0$, nelze nalézt řešení, ve kterém by byly všechny pomocné proměnné nulové. V takovém případě úloha **nemá přípustné řešení**, neboť žádné z možných řešení nesplňuje všechny omezující podmínky úlohy.

2.3.3 Optimalizace původní účelové funkce (II. fáze)

Po úspěšném ukončení I. fáze, tedy po eliminaci všech pomocných proměnných a dosažení hodnoty pomocné účelové funkce $f' = 0$, následuje II. fáze výpočtu. V této části výpočet pokračuje podle pravidel jednofázové simplexové metody. Základní proměnné získané v I. fázi jsou zachovány a dále upravovány s cílem optimalizace původní účelové funkce. [1]

2.3.4 Řešení příkladu

Algoritmus řešení úlohy jednofázové simplexové metody lze shrnout do následujících bodů:¹⁵

1. **Sestavení výchozí simplexové tabulky** – doplnění o přídatné a pomocné proměnné, včetně řádku účelové funkce a pomocné účelové funkce v anulovaném tvaru;
2. **I. fáze – Výběr klíčového sloupce**: Zvolí se proměnná s největším kladným f'_k ;
3. **Výběr klíčového řádku**: $\min. \beta_i = \frac{b_i}{a_{ik}}$ z nezáporných;
4. **Přepočítání tabulky**: Eliminace klíčového prvku pomocí ekvivalentních maticových úprav;
5. **Test přípustnosti**: Postup se opakuje od bodu 2, dokud není dosaženo $f' = 0$, poté následuje přechod do II. fáze, jinak nemá přípustné řešení;

¹⁵ Převzato a upraveno ze zdroje [5]

6. **II. fáze – Test optimality:** Pokud jsou všechny redukováné ceny $f_k \geq 0$ (pro maximalizaci), resp. $f_k \leq 0$ (pro minimalizaci), pak konec a odečtení řešení;
 7. **Výběr klíčového sloupce:** Zvolí se proměnná s největším záporným f_k (pro maximalizaci), resp. s největším kladným f_k (pro minimalizaci);
 8. **Výběr klíčového řádku:** $\min. \beta_i = \frac{b_i}{a_{ik}}$ z nezáporných;
 9. **Přepočet tabulky:** Eliminace klíčového prvku pomocí ekvivalentních maticových úprav;
 10. **Opakování:** Postup se opakuje od bodu 6, dokud není splněna podmínka optimality;
- Z ukázkového příkladu (2.3) upravíme omezení tak, aby bylo možné ilustrovat dvoufázovou simplexovou metodu.

$$\max f = 20x_1 + 14x_2 \quad (2.6)$$

$$\text{za podmínek: } 0,5x_1 + 0,25x_2 \leq 40$$

$$0,5x_1 + 0,5x_2 \geq 60$$

$$0,25x_2 \geq 25$$

$$x_j \geq 0$$

Rozšířený kánonický tvar:

$$0,5x_1 + 0,25x_2 + x_3 = 40 \quad (2.7)$$

$$0,5x_1 + 0,5x_2 - x_4 + y_1 = 60$$

$$0,25x_2 - x_5 + y_2 = 25$$

Doplnění podle schématu (1.7):

$$\begin{pmatrix} 0,5 & 0,25 & 1 & 0 & 0 & 0 & 0 \\ 0,5 & 0,5 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0,25 & 0 & 0 & -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} 40 \\ 60 \\ 25 \end{pmatrix} \quad (2.8)$$

Z doplněné soustavy podle schématu (2.8) vyplývá, že základní řešení je určeno vektory $x^1 = (0; 0; 40; 0; 0)$ a $y^1 = (60; 25)$. Hodnoty ve vektoru x^1 odpovídají strukturálním a přídatným proměnným, zatímco vektor y^1 obsahuje hodnoty pomocných proměnných, které tvoří jednotkové sloupce v rozšířené matici. Dosazením hodnot vektoru x^1 do omezení zadané úlohy (2.6) zjistíme, že řešení nesplňuje všechny omezující podmínky původní úlohy LP, a tedy není přípustné.

Tab. 8: Hodnoty přídatných a pomocných proměnných

Omezení	Levá strana	Relace	Pravá strana	splněno	rozdíl	
					x	y
$0,5x_1 + 0,25x_2 \leq 40$	0	$\leq$	40	Ano	40	-
$0,5x_1 + 0,5x_2 \geq 60$	0	$\geq$	60	Ne	0	60
$0,25x_2 \geq 25$	0	$\geq$	25	Ne	0	25

(zdroj: upraveno podle [1])

Z přiložené tabulky lze odečíst, že pouze první omezující podmínka je v aktuálním základním řešení x^1 splněna, odpovídá nenulová hodnota přídatné proměnné $x_3 = 40$. Naproti tomu druhé a třetí omezení porušena jsou. Rozdíl mezi pravou a levou stranou je u druhé podmínky roven 60 a u třetí 25, což přímo odpovídá hodnotám příslušných pomocných proměnných $y_1 = 60$ a $y_2 = 25$. To znamená, že obě pomocné proměnné signalizují míru porušení příslušných omezení. [1]

Z uvedeného vyplývá, že přípustné řešení úlohy lze získat pouze tehdy, pokud budou všechny pomocné proměnné nabývat nulových hodnot, což je dosaženo minimalizací pomocné účelové funkce f' v I. fázi výpočtu.

Po převedení všech pomocných proměnných na jednu stranu rovnic a sečtením podle vztahu (2.5) vznikne pomocná účelová funkce. Ta je následně upravena do anulovaného tvaru pro zařazení do simplexové tabulky.

$$y_1 = 60 - 0,5x_1 - 0,5x_2 + x_4 \quad (2.9)$$

$$y_2 = 25 - 0,25x_2 + x_5$$

$$\begin{aligned} f' &= y_1 + y_2 = 60 - 0,5x_1 - 0,5x_2 + x_4 + 25 - 0,25x_2 + x_5 = \\ &= 85 - 0,5x_1 - 0,75x_2 + x_4 + x_5 \end{aligned}$$

V anulovaném tvaru má potom pomocná účelová funkce tuto konečnou podobu:

$$f' + 0,5x_1 + 0,75x_2 - x_4 - x_5 = 85$$

Výchozí simplexová tabulka doplněným o pomocné proměnné a řádek pomocné účelové funkce bude mít následující podobu:

Tab. 9: Výchozí simplexová tabulka pro dvoufázovou simplexovou metodu

	x_1	x_2	x_3	x_4	x_5	y_1	y_2	b_i
	0,5	0,25	1	0	0	0	0	40
	0,5	0,5	0	-1	0	1	0	60
	0	0,25	0	0	-1	0	1	25
f	-20	-14	0	0	0	0	0	0
f'	0,5	0,75	0	-1	-1	0	0	85

(zdroj: vlastní)

Klíčový prvek se určí podle předchozího metodického postupu a následně se provede přepočítání simplexové tabulky pomocí ekvivalentních maticových úprav, jak ukazuje Tab. 10.

Tab. 10: I. fáze – minimalizace pomocné účelové funkce f'

I. fáze	x_1	x_2	x_3	x_4	x_5	y_1	y_2	b_i	$t = b_i/a_{ik}$
Výchozí tvar	0,5	0,25	1	0	0	0	0	40	80
	0,5	0,5	0	-1	0	1	0	60	120
	0	0,25	0	0	-1	0	1	25	-
f	-20	-14	0	0	0	0	0	0	
f'	0,5	0,75	0	-1	-1	0	0	85	
1. iterace	0,5	0	1	0	1	0	-1	15	15
	0,5	0	0	-1	2	1	-2	10	5
	0	1	0	0	-4	0	4	100	-
f	-20	0	0	0	-56	0	56	1 400	
f'	0,5	0	0	-1	2	0	-3	10	
2. iterace	0,25	0	1	0,5	0	-0,5	0	10	
	0,25	0	0	-0,5	1	0,5	-1	5	
	1	1	0	-2	0	2	0	120	
f	-6	0	0	-28	0	28	0	1 680	
f'	0	0	0	0	0	-1	-1	0	

(zdroj: [1])

Po první iteraci byla hodnota pomocné účelové funkce rovna $f' = 10$ a v řádku f'_k se stále nachází kladný koeficient, což znamená, že minimalizace může pokračovat. Ve druhé iteraci se podařilo dosáhnout hodnoty $f' = 0$ a zároveň již žádný kladný koeficient v řádku f'_k nezbyvá. Lze tedy uzavřít, že úloha má přípustné řešení.

Tímto řešením je $x^3 = (0; 5; 10; 0; 5)$ a $y^3 = (0; 0)$ a zároveň hodnota původní účelové funkce dosahuje $f^3 = 1\,680$ a pomocná účelová funkce $f'^3 = 0$. Vzhledem k tomu, že

hodnota pomocné účelové funkce je nulová, představuje řešení x^3 současně výchozí základní řešení původní úlohy LP.

Nalezením výchozího přípustného řešení je ukončena I. fáze výpočtu dvoufázové simplexové metody. Pomocné proměnné, stejně jako pomocná účelová funkce, již dále nevstupují do výpočtu a nejsou nadále zvažovány. Ve II. fázi výpočtu je optimalizována původní účelová funkce. [1]

Tab. 11: Optimální řešení příkladu (2.6)

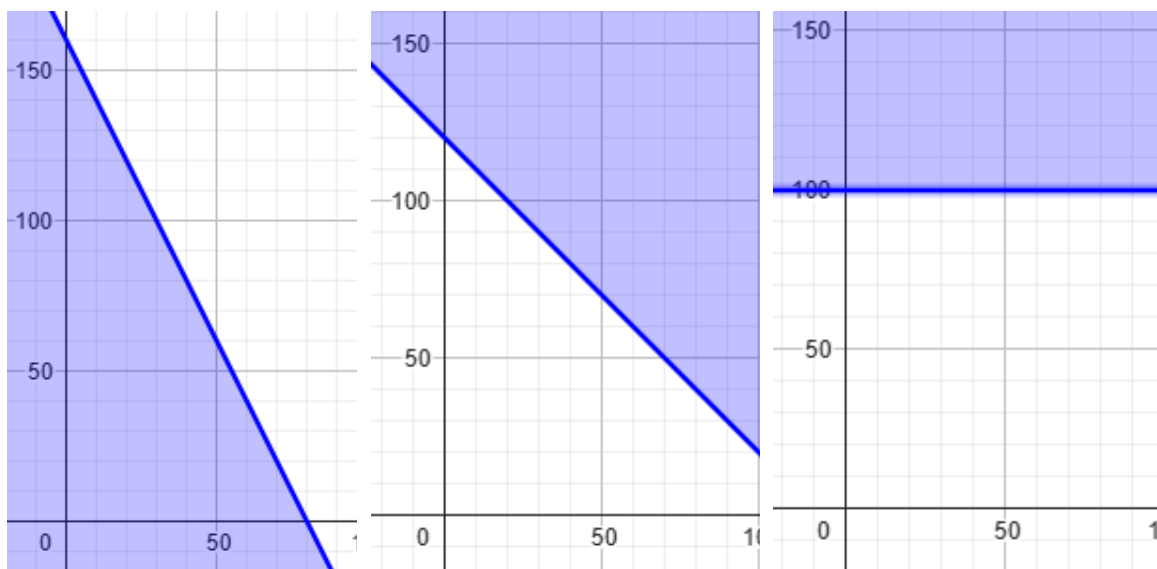
II. fáze	x_1	x_2	x_3	x_4	x_5	b_i	$t = b_i/a_{ik}$
2. iterace	0,25	0	1	0,5	0	10	20
	0,25	0	0	-0,5	1	5	-
	1	1	0	-2	0	120	-
f	-6	0	0	-28	0	1 680	
3. iterace	0,5	0	2	1	0	20	
	0,5	0	1	0	1	15	
	2	1	4	0	0	160	
f	8	0	56	0	0	2 240	

(zdroj: [1])

Po provedení přepočtu simplexové tabulky získáme nové řešení $x^4 = (0; 160; 0; 20; 15)$ s odpovídající hodnotou účelové funkce $f = 2\,240$ (viz Tab. 11). Tímto je výpočet ve II. fázi ukončen, neboť nalezené řešení je již optimální.

Grafické řešení:

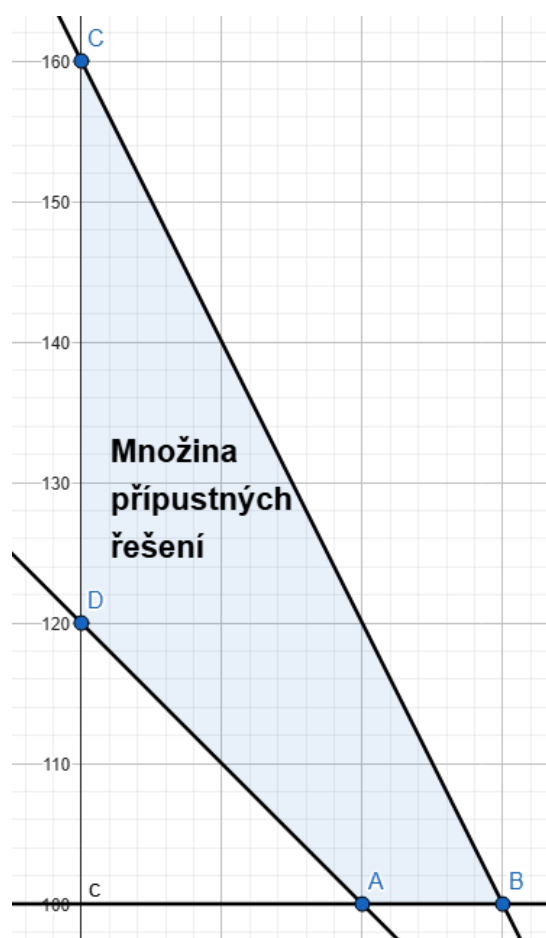
Grafické znázornění vychází z omezujících podmínek, které tvoří oblast přípustných řešení. Jednotlivá omezení jsou znázorněna graficky pomocí dosazení jedné proměnné rovné nule a dopočítání druhé, čímž vznikají přímky vymezující příslušné poloprostory. Výsledné znázornění je uvedeno na Obrázek 5, který zachycuje průběh každého omezení a jejich průnik tvoří množinu přípustných řešení v prvním kvadrantu.



$$0,5x_1 + 0,25x_2 \leq 40$$

$$0,5x_1 + 0,5x_2 \geq 60$$

$$0,25x_2 \geq 25$$



Obrázek 5: Grafické znázornění příkladu (2.6)

(zdroj: vlastní)

Obrázek 5 znázorňuje, že množina přípustných řešení tvoří konvexní polyedr vymezený vrcholy ABCD. Po těchto vrcholech následně postupuje simplexová metoda s cílem nalezení optimálního řešení. Následující Tab. 12 uvádí všechna základní řešení. U přípustných řešení je uveden také odpovídající vrchol konvexního polyedru a hodnota účelové funkce vypočtená dosazením.

Kánonický tvar (bez pomocných proměnných y_i):

$$0,5x_1 + 0,25x_2 + x_3 = 40 \quad (2.10)$$

$$0,5x_1 + 0,5x_2 - x_4 = 60$$

$$0,25x_2 - x_5 = 25$$

Počet základních řešení: $\binom{5}{3} = 10$

Tab. 12: Výpočet hodnot účelové funkce podle souřadnic

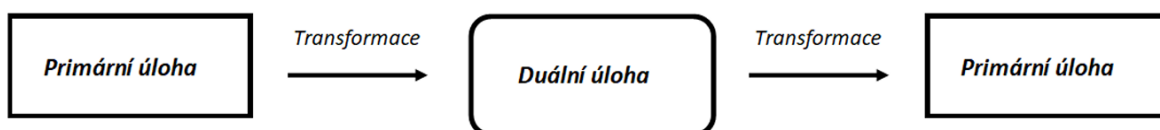
	x_1	x_2	x_3	x_4	x_5	Vrchol	$f = 20x_1 + 14x_2$
přípustné	20	100	5	0	0	A	1 800
přípustné	30	100	0	5	0	B	2 000
přípustné	0	160	0	20	15	C	2 240
přípustné	0	120	10	0	5	D	1 680
nepřípustné	0	0	40	60	-25		
nepřípustné	80	0	0	-20	-25		
nepřípustné	40	80	0	0	-5		
nepřípustné	120	0	-20	0	-25		
nepřípustné	0	100	15	-10	0		
nepřípustné	-	0	-	-	0		

(zdroj: vlastní)

2.4 Dualita v LP

Každá úloha LP má svou odpovídající duální úlohu (model), která je s ní úzce propojena. Obě tyto úlohy vycházejí ze stejných strukturních a omezujících koeficientů, avšak takovým způsobem, že pokud je jeden z těchto problémů maximalizační, pak druhý je minimalizační. Pokud existuje konečné optimální řešení obou úloh, pak platí, že hodnoty jejich účelových funkcí jsou si rovné. [12]

Úloha, která byla formulována jako první, se obvykle označuje jako primární a na jejím základě lze podle stanovených pravidel sestavit duální úlohu. Vztah mezi těmito dvěma úlohami je vzájemný, protože z každé duální úlohy lze zpětně odvodit odpovídající primární úlohu. To, která z úloh je označena jako primární a která jako duální, není určeno jejím matematickým zápisem. [14]



Obrázek 6: Ilustrace transformace mezi primární a duální úlohou

(zdroj:[15])

Duální úlohu lze podle způsobu zadání primární úlohy rozdělit na dva typy. Pokud jsou všechna omezení v primární úloze formulována ve tvaru nerovnic, hovoříme o tzv. **souměrném** (symetrickém) tvaru. V případě, že některá z omezení mají podobu rovnic, jedná se o tvar **nesouměrný** (nesymetrický). [14][15]

2.4.1 Symetrická duální úloha

Duální úlohu lze sestavit přímo pomocí obecně uznávaného vztahu mezi koeficienty účelové funkce, maticí omezení a pravou stranou.

Vzájemný vztah¹⁶: [12]

Primární úloha:	Duální úloha:	(2.11)
$c^T x \rightarrow \max$	$b^T y \rightarrow \min$	
$Ax \leq b$	$A^T y \geq c$	
$x \geq 0$	$y \geq 0$	

¹⁶ Pokud je primární úloha formulována jako maximalizační.

kde T značí operaci transpozice – řádky se stanou sloupci a sloupce řádky.

Maticové znázornění převodu duální úlohy z ukázkového příkladu (2.3):

Primární úloha:

$$\max f(x) = \underbrace{(20 \quad 14)}_{c^T} x$$

$$\underbrace{\begin{pmatrix} 0,5 & 0,25 \\ 0,5 & 0,5 \\ 0 & 0,25 \end{pmatrix}}_A x \leq \underbrace{\begin{pmatrix} 40 \\ 60 \\ 25 \end{pmatrix}}_b$$

$$x_j \geq 0$$

Duální úloha:

$$\min g(y) = \underbrace{(40 \quad 60 \quad 25)}_{b^T} y$$

$$\underbrace{\begin{pmatrix} 0,5 & 0,5 & 0 \\ 0,25 & 0,5 & 0,25 \end{pmatrix}}_{A^T} y \geq \underbrace{\begin{pmatrix} 20 \\ 14 \end{pmatrix}}_c$$

$$y_j \geq 0$$

Vizuální přehled převodu duální úlohy z ukázkového příkladu (2.3):

Primární úloha:

$$\max f(x) = 20x_1 + 14x_2$$

$$0,5x_1 + 0,25x_2 \leq 40$$

$$0,5x_1 + 0,5x_2 \leq 60$$

$$0,25x_2 \leq 25$$

$$x_j \geq 0$$

Duální úloha:

$$\min g(y) = 40y_1 + 60y_2 + 25y_3$$

$$0,5y_1 + 0,5y_2 \geq 20$$

$$0,25y_1 + 0,5y_2 + 0,25y_3 \geq 14$$

$$y_j \geq 0$$

Pokud některé omezující podmínky primární úlohy nejsou vyjádřeny v požadovaném standardizovaném tvaru, je nutné je před sestavením duální úlohy vhodně upravit. Nejčastěji se úprava provádí vynásobením daného omezení hodnotou -1 , čímž se dosáhne potřebného tvaru pro převod mezi primární a duální úlohou.

Znázornění postupu úpravy omezení vycházející z ukázkového příkladu (2.6):

Primární úloha:

(2.12)

$$\max f = 20x_1 + 14x_2$$

$$0,5x_1 + 0,25x_2 \leq 40$$

$$0,5x_1 + 0,5x_2 \geq 60 \xrightarrow{-1} -0,5x_1 - 0,5x_2 \leq -60$$

$$0,25x_2 \geq 25 \xrightarrow{-1} -0,25x_2 \leq -25$$

$$x_j \geq 0$$

Duální úloha:

$$\min g = 40y_1 - 60y_2 - 25y_3$$

$$0,5y_1 - 0,5y_2 \geq 20$$

$$0,25y_1 - 0,5y_2 - 0,25y_3 \geq 14$$

$$y_j \geq 0$$

2.4.2 Nesymetrická duální úloha

Na rozdíl od symetrického případu, kdy lze duální úlohu sestavit přímo na základě standardního vztahu, vyžaduje nesymetrická úloha úpravu rovnic na nerovnice, aby bylo možné určit odpovídající duální model. Po této úpravě lze následně využít postup pro symetrickou duální formulaci.

Pro řešení nesymetrické duální úlohy lze využít dva přístupy řešení. V rámci ukázkového příkladu (2.3) bude jeden z omezujících vztahů nahrazen rovností, aby bylo možné ilustrovat vznik nesymetrické duální úlohy.

$$\begin{aligned} \max f &= 20x_1 + 14x_2 \\ 0,5x_1 + 0,25x_2 &= 40 \\ 0,5x_1 + 0,5x_2 &\leq 60 \\ 0,25x_2 &\leq 25 \\ x_j &\geq 0 \end{aligned} \tag{2.13}$$

1. přístup:

Rovnost v prvním omezení lze nahradit dvojicí nerovností ve tvaru $\leq$ a $\geq$, čímž se převede na dvě samostatné podmínky. [9]

Primární úloha:	Duální úloha:	(2.14)
$\max f(x) = 20x_1 + 14x_2$	$\min g(y) = 40y_1 - 40y_2 + 60y_3 + 25y_4$	
$0,5x_1 + 0,25x_2 \leq 40$	$0,5y_1 - 0,5y_2 + 0,5y_3 \geq 20$	
$(0,5x_1 + 0,25x_2 \geq 40) * (-1)$	$0,25y_1 - 0,25y_2 + 0,5y_3 + 0,25y_4 \geq 14$	
$0,5x_1 + 0,5x_2 \leq 60$		
$0,25x_2 \leq 25$		
$x_j \geq 0$	$y_j \geq 0$	

2. přístup:

V tomto přístupu se k i -tému omezení formulovanému jako rovnost, v našem příkladu první, přiřadí příslušná duální proměnná, která je nazývána jako volná proměnná (URS). Taková proměnná nemusí být nezáporná, a proto se při dalším výpočtu zavádí jako rozdíl dvou nezáporných proměnných, jak ilustruje vztah (1.5).

$$x_j = x_j^+ - x_j^-, \quad x_j^+, x_j^- \geq 0$$

Při zavedení volné proměnné nelze v základním přípustném řešení připustit, aby obě složky x_j^+ a x_j^- byly současně kladné. Každá URS proměnná tak musí v řešení spadat do jedné ze tří situací: [9]

1. $x_j^+ > 0, x_j^- = 0$:

proměnná x_j je kladná, výsledek odpovídá $x_j = x_j^+$.

2. $x_j^+ = 0, x_j^- \geq 0$

proměnná x_j je záporná, výsledek odpovídá $x_j = -x_j^-$.

3. $x_j^+ = 0, x_j^- = 0$:

$x_j = 0$, proměnná v řešení nenabývá žádné hodnoty.

Primární úloha:

Duální úloha:

(2.15)

$$\max f(x) = 20x_1 + 14x_2$$

$$\min g(y) = 40y_1 + 60y_2 + 25y_3$$

$$0,5x_1 + 0,25x_2 = 40$$

$$0,5y_1 + 0,25y_2 \geq 40$$

$$0,5x_1 + 0,5x_2 \leq 60$$

$$0,25y_1 + 0,5y_2 + 0,25y_3 \geq 60$$

$$0,25x_2 \leq 25$$

$$x_j \geq 0$$

$$y_1 \text{ URS}, y_2 \geq 0, y_3 \geq 0$$

Přehled vzájemných vztahů mezi prvky primární a duální úlohy shrnuje Tab. 13. Ta poskytuje systematické srovnání jednotlivých složek obou modelů – účelové funkce, pravé strany, omezení i typu proměnných – a ukazuje, jak si jednotlivé komponenty v rámci duality navzájem odpovídají.

Tab. 13: Vztah mezi primární a duální úlohou

Primární úloha	Duální úloha
Maximalizovat	Minimalizovat
Minimalizovat	Maximalizovat
Koeficienty účelové funkce	Prvky pravé strany omezení
Prvky pravé strany omezení	Koeficienty účelové funkce
i -tý řádek matice omezení	i -tý sloupec matice omezení
j -tý sloupec matice omezení	j -tý řádek matice omezení
i -té omezení typu $\leq$	i -tá proměnná $y_i \geq 0$
i -té omezení typu $\geq$	i -tá proměnná $y_i \leq 0$ ¹⁷
i -té omezení typu $=$	i -tá proměnná volná (<i>URS</i>)
j -tá proměnná $x_j \geq 0$	j -té omezení typu $\geq$
j -tá proměnná $x_j \leq 0$ ¹⁸	j -té omezení typu $\leq$
j -tá proměnná volná (<i>URS</i>)	j -té omezení typu $=$

(zdroj: [9])

2.5 Odečtení simplexové tabulky

Kromě hodnot strukturních a přídatných proměnných lze z optimální simplexové tabulky vyčíst i další informace, které mají zásadní význam pro interpretaci výsledků. Mezi tyto informace patří zejména **stínové ceny** a **redukované ceny**, které poskytují přímou vazbu mezi strukturou omezení a hodnotou účelové funkce. [1]

Přehledné určení stínových a redukovaných cen v rámci simplexové tabulky znázorňuje Tab. 14.

¹⁷ Pro zachování nezápornosti se proměnné ≤ 0 v značí s opačným znaménkem ($-y$).

¹⁸ Pro zachování nezápornosti se proměnné ≤ 0 v značí s opačným znaménkem ($-x$).

Tab. 14: Určení stínových a redukováných cen

Strukturní proměnné				Přídavné proměnné				Pravá strana
x_1	x_2	...	x_n	x_{n+1}	x_{n+2}	...	x_{n+m}	b_i
Strukturní koeficienty				Strukturní koeficienty				Hodnoty základních proměnných
f_1	f_2	...	f_n	u_1	u_2	...	u_{n+m}	Hodnota účel. funkce
Redukované ceny				Stínové ceny				

(zdroj: [1])

2.5.1 Stínové ceny

Stínové ceny (neboli duální hodnoty) představují hodnoty přiřazené jednotlivým omezujícím podmínkám a vyjadřují, o kolik se změní hodnota účelové funkce při jednotkové změně pravé strany daného omezení. Při řešení duální úlohy pak hodnota účelové funkce odpovídá součinu vektoru stínových cen a vektoru pravých stran primární úlohy, podle vztahu: [1]

$$y = b_1 u_1 + b_2 u_2 + \dots + b_m u_m \quad (2.16)$$

Práci se stínovými cenami si demonstrováme na ukázkovém příkladu (2.3), využijeme při tom poslední iteraci simplexové metody uvedenou v Tab. 6.

Tab. 15: Optimální řešení příkladu (2.3)

	x_1	x_2	x_3	x_4	x_5	b
	1	0	4	-2	0	40
	0	1	-4	4	0	80
	0	0	1	-1	1	5
f	0	0	24	16	0	1 920

(zdroj: [1])

Stínové ceny přiřazené jednotlivým omezením jsou: $u_1 = 24$, $u_2 = 16$, $u_3 = 0$.

Optimální hodnota duální úlohy musí být 1 920, neboť podle principu duálního řešení platí, že obě úlohy sdílejí stejnou optimální hodnotu účelové funkce.

Výpočet podle vztahu (2.16):

$$y = b_1 u_1 + b_2 u_2 + b_3 u_3 = 40 * 24 + 80 * 16 + 5 * 0 = 1\,920 \quad (2.17)$$

Pro ilustraci praktického významu stínových cen bude v následující části ukázáno, jak se změní hodnota účelové funkce při úpravě prvků ve vektoru pravých stran. Konkrétně se zaměříme na upravené zadání ukázkového příkladu (2.3), ve kterém došlo ke změně hodnot omezujících podmínky.

$$\max f = 20x_1 + 14x_2 \quad (2.18)$$

$$0,5x_1 + 0,25x_2 \leq 41$$

$$0,5x_1 + 0,5x_2 \leq 58$$

$$0,25x_2 \leq 30$$

$$x_j \geq 0$$

Pravé strany omezení byly upraveny tak, že k první bylo přičteno +1, ke druhé -2 a ke třetí +5 jednotek. Dopad těchto změn na hodnotu účelové funkce lze vyjádřit pomocí vztahu:

$$f = 1920 + u_1 - 2u_2 + 5u_3 = 1920 + 24 - 2 * 16 + 5 * 0 = 1912 \quad (2.19)$$

Tab. 16: Optimální řešení dopadu změn pravých stran příkladu (2.18)

	x_1	x_2	x_3	x_4	x_5	b
	1	0	4	-2	0	48
	0	1	-4	4	0	68
	0	0	1	-1	1	13
f	0	0	24	16	0	1912

(zdroj: vlastní)

2.5.2 Redukované ceny

V rámci testu optimality při řešení simplexové metody redukované ceny představují hodnoty přiřazené ke **strukturním proměnným**. V případě maximalizační úlohy se jedná o potřebné zvýšení tohoto koeficientu, zatímco v případě minimalizační úlohy je nutné, aby došlo k jeho snížení. [1]

Pro interpretaci redukovaných cen z_j je vhodné si uvědomit, že jejich hodnotu lze vyjádřit jako rozdíl:

$$z_j = u^T a_j - c_j \quad (2.20)$$

kde:

a_j – j -tý sloupec strukturních koeficientů A (omezení),

$u^T a_j$ – skalární součin vektoru stínových cen s vektorem a_j , který reprezentuje celkové ocenění zdrojů potřebných pro realizaci j -tého procesu,

c_j – vyjadřuje přínos této proměnné (například zisk).

Redukovaná cena z_j poskytuje informaci o efektivitě zařazení dané proměnné do optimální báze. Situaci shrnuje Tab. 17, na jejímž základě lze rozlišit tři možné případy.

Tab. 17: Interpretace redukované ceny v závislosti na typu úlohy

Případy	Maximalizace	Minimalizace
$z_j > 0$	Proměnná není efektivní pro zařazení do optimální báze. ¹⁹	Proměnná může zlepšit řešení
$z_j < 0$	Proměnná může zlepšit řešení	Proměnná není efektivní pro zařazení do optimální báze. ²⁰
$z_j = 0$	Proměnná je efektivní pro zařazení do optimální báze	

(zdroj: vlastní zpracování podle [1])

Pro ilustraci práce s redukovanými cenami rozšíříme ukázkový příklad (2.3) o novou strukturní proměnnou x_3 včetně příslušných koeficientů v omezujících podmínkách. [1]

$$\max f = 20x_1 + 14x_2 + 12x_3 \quad (2.21)$$

$$0,5x_1 + 0,25x_2 + 0,2x_3 \leq 40$$

$$0,5x_1 + 0,5x_2 + 0,3x_3 \leq 60$$

$$0,25x_2 + 0,5x_3 \leq 30$$

$$x_j \geq 0$$

¹⁹ Koeficient c_j by musel být zvýšen minimálně o z_j , aby se stala efektivní.

²⁰ Koeficient c_j by musel být zvýšen minimálně o z_j , aby se stala efektivní.

Pro rozhodnutí, zda zařadit novou proměnnou, se vypočítá její redukovaná cena.

Ocenění zdrojů u proměnné x_3 je:

$$u^T a_3 = 0,2u_1 + 0,3u_2 + 0,5u_3 = 0,2 * 24 + 0,3 * 16 + 0,5 * 0 = 9,6 \quad (2.22)$$

Následně se redukovaná cena vypočítá jako

$$z_3 = 9,6 - 12 = -2,4 \quad (2.23)$$

Hodnota redukované ceny z_3 ukazuje, že je efektivní zařadit proměnnou x_3 do báze.

2.6 Možnosti zakončení úloh LP

Při řešení úloh LP pomocí simplexové metody může výpočet skončit několika způsoby v závislosti na vlastnostech modelu i vstupních datech. Výše uvedený Obrázek 2 představuje jednotlivé možné scénáře zakončení výpočtu. V následujícím textu si přiblížíme, jak k těmto situacím dochází a jak je lze identifikovat v simplexové tabulce.

2.6.1 Úloha s jediným optimálním řešením

Jediné optimální řešení nastává tehdy, jsou-li všechny redukované ceny z_j u nezákladních proměnných v simplexové tabulce kladné (při maximalizaci), resp. záporné (při minimalizaci). To značí, že žádná z nezákladních proměnných nemůže vstoupit do báze bez zhoršení hodnoty účelové funkce. V důsledku toho je dosažené řešení jednoznačně optimální. [1]

2.6.2 Úloha má alternativní optimální řešení

Alternativní optimální řešení nastává tehdy, pokud kromě optimální báze existují i další řešení se stejnou hodnotou účelové funkce. Pokud je při výběru klíčového sloupce zvolen některý z těch, který není součástí báze a zároveň nulovou hodnotou v řádku účelové funkce, může dojít k nalezení alternativního optimálního řešení. [1]

Z ukázkového příkladu (2.3) bude pro účely demonstrace upravena účelová funkce, aby bylo možné ilustrovat vznik alternativního optimálního řešení.

$$\max f = 20x_1 + 20x_2 \quad (2.24)$$

$$0,5x_1 + 0,25x_2 \leq 40$$

$$0,5x_1 + 0,5x_2 \leq 60$$

$$0,25x_2 \leq 25$$

Optimální řešení spolu s jeho alternativní variantou je znázorněno v Tab. 18.

Tab. 18: Alternativní optimální řešení

	x_1	x_2	x_3	x_4	x_5	b
	1	0	4	-2	0	40
	0	1	-4	4	0	80
	0	0	1	-1	1	5
f	0	0	0	40	0	2 400
	1	0	0	2	-4	20
	0	1	0	0	4	100
	0	0	1	-1	1	5
	0	0	0	40	0	2 400

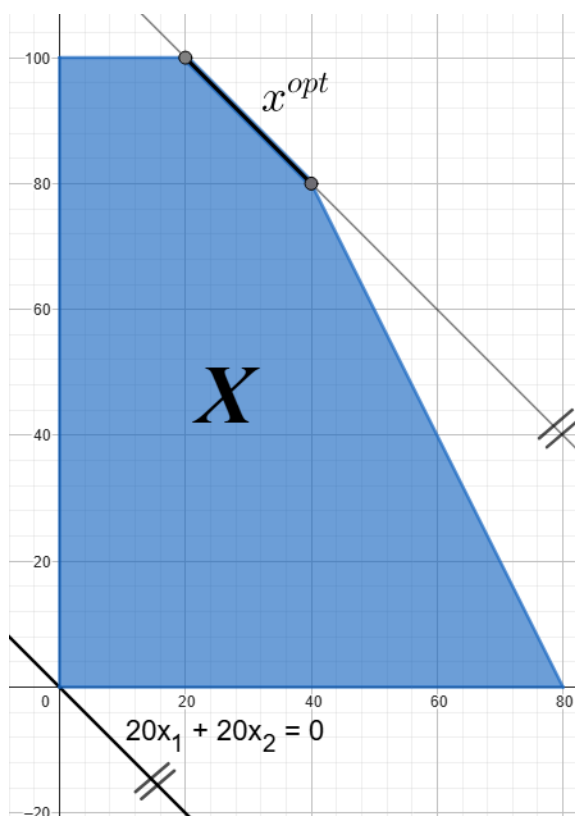
(zdroj:[1])

V důsledku změny se změní simplexová tabulka. Z Tab. 18 je patrné, že optimální řešení je $x^1 = (40; 80; 0; 0; 5)$ a zároveň koeficient u nezákladní proměnné x_3 v účelovém řádku je nulový. Zvolením x_3 jako klíčového sloupce, dojde k výměně báze a získáme nové optimální řešení $x^2 = (20; 100; 5; 0; 0)$, které má stejnou hodnotu účelové funkce.

Jak je pozorovat z Obrázek 7, optimální řešení příkladu (2.24) není reprezentováno jedním vrcholem množiny přípustných řešení, ale leží na její hraně. Konkrétně se jedná o úsečku mezi dvěma extrémními body, kde proměnná x_1 může nabývat hodnot z intervalu $[20; 40]$, zatímco odpovídající hodnoty proměnné x_2 náleží intervalu $[80; 100]$, přičemž ze strukturálního řešení v bázi dále vyplývá, že mezi proměnnými platí vztah:

$$x_1 + x_2 = 120 \quad (2.25)$$

Možné kombinace těchto hodnot a odpovídající hodnoty účelové funkce uvádí Tab. 19



Obrázek 7: Alternativní optimální řešení

(zdroj: vlastní)

Tab. 19: Možné kombinace hodnot x_1 a x_2

x_1	x_2	$f = 20x_1 + 20x_2$
25	95	2 400
30	90	2 400
32	88	2 400
39	81	2 400

(zdroj: vlastní)

V praxi jsou alternativní optimální řešení užitečná, protože si můžeme vybrat z více možností, aniž by došlo ke zhoršení hodnoty účelové funkce. Pokud daný příklad představuje situaci výrobního mixu, může být výhodné nabízet na trhu dva produkty místo jednoho. [15]

2.6.3 Úloha nemá omezenou hodnotu účelové funkce

Tento typ zakončení výpočtu nastává v případě, kdy koeficienty ve klíčovém sloupci jsou záporné, nebo není možné výběr klíčového řádku kvůli dělení nulou. Taková situace značí, že lze neomezeně zvyšovat (nebo snižovat) hodnotu účelové funkce bez porušení omezení, a tedy optimalizační úloha nemá omezené řešení.²¹

Zvažme následující jednoduchý ukázkový příklad ze zdroje [1]:

$$\max f = x_1 + 2x_2 \quad (2.26)$$

$$\text{za podmíněk: } x_1 - x_2 \leq 12$$

$$-2x_1 + x_2 \leq 8$$

$$x_j \geq 0$$

Tab. 20 simplexové metody pro příklad (2.26) zachycuje průběh řešení, ve kterém dojde k ukončení výpočtu z důvodu neomezené hodnoty účelové funkce.

Tab. 20: Neomezená hodnota účelové funkce

	x_1	x_2	x_3	x_4	b
	-1	0	1	1	20
	-2	1	0	1	8
f	-3	0	0	2	16

Neomezený prostor řešení může naznačovat, že model je špatně sestaven. Nejčastěji to bývá způsobeno tím, že některá klíčová omezení nebyla do modelu zahrnuta. [15]

2.6.4 Úloha nemá přípustné řešení

Tato situace představuje poslední možný způsob zakončení výpočtu a souvisí s využitím dvoufázové simplexové metody. V I. fázi této metody je cílem minimalizovat pomocnou účelovou funkci, jejímž účelem je zjistit nepřípustnost úlohy. Pokud se pomocnou účelovou funkcí podaří dosáhnout hodnoty nula, znamená to, že všechny nepřípustné složky byly odstraněny a bylo nalezeno přípustné řešení původní úlohy.²² [1]

²¹ Anglický se nazývá *unbounded solution*

²² Anglický se nazývá *infeasible solution*

II PRAKTICKÁ ČÁST

3 STUDIJNÍ OPORA

V této části se zaměříme na konkrétní příklady úloh LP a budou řešeny jak maximalizační, tak minimalizační varianty.

Cílem kapitoly je nejen poskytnout soubor řešených úloh, ale zároveň vytvořit studijní oporu, která neslouží pouze jako formální výklad, ale také jako přehledný průvodce řešením jednotlivých typů úloh od zadání až po výsledek.

Veškeré tabulky uvedené v této kapitole jsou **vlastní tvorbou** a byly zpracovány za účelem ilustrace a podpory výkladu řešených úloh.

3.1 Ilustrativní řešené příklady

Tato podkapitola obsahuje pět příkladů úloh LP, které demonstrují použití simplexové metody pro maximalizační i minimalizační úlohy s různými typy omezení ($\leq$, $\geq$, $=$) a jejich kombinacemi. Příklady byly zvoleny tak, aby pokrývaly typické situace, se kterými se lze v rámci LP běžně setkat, od základních úloh až po případy vyžadující použití dvoufázové metody.

První tři příklady budou rozpracovány podrobně, aby byly zřejmé veškeré kroky a odlišnosti v postupu řešení. U čtvrtého a pátého příkladu bude věnována pozornost především výpočetní části, protože základní principy byly objasněny v předchozích řešeních.

Příklady budou zahrnovat tyto kroky:

1. Převod omezení do kanonického tvaru
2. Sestavení výchozí tvaru simplexové tabulky
3. Výběr klíčového prvku a řešení iteracemi simplexové metody
4. Odečtení řešení a formulace duální úlohy

Přehled řešených příkladů:

- **Příklad 1:** Maximalizační úloha s omezeními ve tvaru $\leq$
- **Příklad 2:** Minimalizační úloha s omezeními ve tvaru $\geq$
- **Příklad 3:** Maximalizační úloha s omezeními ve tvaru $=$
- **Příklad 4:** Maximalizační úloha s kombinovanými omezeními
- **Příklad 5:** Minimalizační úloha s kombinovanými omezeními

3.1.1 Maximalizační úloha s omezeními ve tvaru $\leq$

Vlastní příklad pro procvičování:

$$\max f(x) = 3x_1 + 5x_2 \quad (3.1)$$

$$\text{za podmínek: } 3x_1 + 2x_2 \leq 10$$

$$x_2 \leq 15$$

$$4x_1 \leq 20$$

$$x_j \geq 0$$

Pokud jsou všechna omezení ve tvaru $\leq$, není nutné přidávat umělé proměnné, stačí zavést pouze přídatné proměnné, které umožní převést nerovnice do kanonického tvaru rovnice.

Celkový počet proměnných v úloze můžeme určit tak, že sečteme počet základních proměnných v účelové funkci f a počet omezení. Každé omezení rozšíříme o jednu přídatnou proměnnou, která se v dané rovnici objeví s koeficientem 1.

Celkový počet proměnných: $2 + 3 = 5$

Kanonický tvar:

$$3x_1 + 2x_2 + x_3 = 10$$

$$x_2 + x_4 = 15$$

$$4x_1 + x_5 = 20$$

Účelová funkce f se zapisuje do posledního řádku simplexové tabulky, vynásobením všech koeficientů -1.

Tab. 21: Výchozí tvar simplexové tabulky příkladu (3.1)

	x_1	x_2	x_3	x_4	x_5	b_i
	3	2	1	0	0	10
	0	1	0	1	0	15
	4	0	0	0	1	20
f	-3	-5	0	0	0	0

Po sestavení výchozí simplexové tabulky určíme klíčový sloupec a klíčový řádek, abychom mohli zahájit iterace. Při maximalizační úloze je klíčovým sloupcem ten, který obsahuje nejmenší záporný koeficient v řádku účelové funkce. Klíčový řádek pak určujeme jako ten s nejmenší kladnou hodnotou, podílem pravé strany (b) a odpovídajícího prvku v klíčovém sloupci.

Tab. 22: Nalezení klíčového sloupce a řádku z Tab. 21

		x_1	x_2	x_3	x_4	x_5	b_j	$t = b_i/a_{mn}$
1. ř.		3	2	1	0	0	10	5
2. ř.		0	1	0	1	0	15	15
3. ř.		4	0	0	0	1	20	-
4. ř.	f	-3	-5↑	0	0	0	0	-

Klíčová hodnota se nachází na průsečíku klíčového sloupce a řádku (hodnota 2). Iteraci zahájíme tak, že celý klíčový řádek vydělíme hodnotou 2. Tím získáme v následující tabulce transformovaný klíčový řádek, ve kterém je hodnota na místě původního klíčového prvku 1.

Následně provedeme transformaci všech řádků tak, aby ve sloupci, kde se nachází klíčová proměnná (v tomto případě x_2), byly ve výsledné tabulce nuly. Tento postup odpovídá Gaussově eliminaci.

Tab. 23: První iterace Tab. 22

1. iterace		x_1	x_2	x_3	x_4	x_5	b_i
5. ř.		3/2	1	1/2	0	0	5
6. ř.		-3/2	0	-1/2	1	0	10
7. ř.		4	0	0	0	1	20
8. ř.	f	9/2	0	5/2	0	0	25

Detailní způsob výpočtu 1. iterace (Tab. 22 – Tab. 23)

- Transformovaný klíčový řádek: 5. ř. = 1. ř. / 2
- 6. ř. = $-1 * 5. ř. + 2. ř.$
- 7. ř. = $0 * 5. ř. + 3. ř.$
- Řádek účelové funkce f : 8. ř. = $5 * 5. ř. + 4. ř.$

V řádku účelové funkce již není žádné záporné číslo, což znamená, že jsme dosáhli optimálního řešení. Nastane-li situace, kdy by se v řádku záporná čísla stále nacházela,

museli bychom opakovat postup nalezení klíčového sloupce a provádět další iterace až do chvíle, kdy budou všechny hodnoty v tomto řádku nezáporné.

Výsledné hodnoty proměnných se určují na základě jednotkových sloupců ve vektoru proměnných:

- Pokud daný sloupec obsahuje právě jednu hodnotu 1 a ostatní hodnoty jsou 0 (tj. tvoří jednotkový vektor), potom hodnota příslušné proměnné odpovídá hodnotě ve sloupci b na tom řádku, kde se nachází jednička.
- Pokud sloupec tuto podmínku nesplňuje, považuje se odpovídající proměnná za nulovou.
- Hodnota účelové funkce f je uvedena v pravém dolním rohu tabulky ve sloupci b .

Výsledné řešení příkladu:

$x^{opt} = (0; 5; 0; 10; 20)$, kde hodnoty v bázi jsou $x_2 = 5$, $x_3 = 10$, $x_4 = 20$;

Redukovaná cena proměnné $x_1 = 4,5$ je kladná, zařazení do báze by nezlepšilo hodnotu účelové funkce;

Stínové ceny (duální hodnoty): $u_1 = 2,5$; $u_2 = 0$; $u_3 = 0$

Úloha má jediné optimální řešení;

Optimální hodnota účelové funkce je $f = 25$;

Kontrola správnosti:

$$\max f(x) = 3x_1 + 5x_2 \rightarrow 25 = 3 \cdot 0 + 5 \cdot 5 \rightarrow 25 = 25$$

$$3x_1 + 2x_2 \leq 10 \rightarrow 3 \cdot 0 + 2 \cdot 5 \leq 10 \rightarrow 10 \leq 10 \text{ (rovnost)}$$

$$x_2 \leq 15 \rightarrow 5 \leq 15 \text{ (nerovnost)}^{23};$$

$$4x_1 \leq 20 \rightarrow 4 \cdot 0 \leq 20 \rightarrow 0 \leq 20 \text{ (nerovnost)}^{24}$$

Pro formulaci duální úlohy z primární využijeme vztahu (2.11).

Primární:

Duální:

$$\begin{aligned} \max f(x) &= c^T x \\ Ax &\leq b \\ x_i &\geq 0 \end{aligned}$$

$$\begin{aligned} \min g(y) &= b^T y \\ A^T y &\geq c \\ y_i &\geq 0 \end{aligned}$$

²³ $x_3 = 10$ představuje nevyužitou kapacitu a po doplnění bude rovnost ($15 = 15$).

²⁴ $x_4 = 10$ představuje nevyužitou kapacitu a po doplnění bude rovnost ($20 = 20$).

Po dosazení hodnot ze zadání a transpozice matic získáváme:

$$c^T = (3, 5) \rightarrow c = \begin{pmatrix} 3 \\ 5 \end{pmatrix}$$

$$A = \begin{pmatrix} 3 & 2 \\ 0 & 1 \\ 4 & 0 \end{pmatrix} \rightarrow A^T = \begin{pmatrix} 3 & 0 & 4 \\ 2 & 1 & 0 \end{pmatrix}$$

$$b = \begin{pmatrix} 10 \\ 15 \\ 20 \end{pmatrix} \rightarrow b^T = (10 \quad 15 \quad 20)$$

Formulace duálního problému:

$$\min g(y) = 10y_1 + 15y_2 + 20y_3$$

$$3y_1 + 4y_3 \geq 3$$

$$2y_1 + y_2 \geq 5$$

$$y_j \geq 0$$

3.1.2 Minimalizační úloha s omezeními ve tvaru $\geq$

Pro řešení příkladu lze využít duální řešení předchozí úlohy (3.1):

$$\min g(y) = 10y_1 + 15y_2 + 20y_3 \tag{3.2}$$

$$\text{za podmíněk: } 3y_1 + 4y_3 \geq 3$$

$$2y_1 + y_2 \geq 5$$

$$y_j \geq 0$$

Pokud jsou některá omezení ve tvaru $\geq$, není možné je přímo převést do kanonického tvaru pouhým přidáním přidavných proměnných. V tomto případě je nutné zavést umělé proměnné²⁵, které umožní sestavit výchozí tvar simplexové tabulky. Tím vzniká potřeba využít dvoufázovou simplexovou metodu, kde v její I. fázi je řešena pomocná účelová funkce sloužící k odstranění umělých proměnných z báze.

Celkový počet proměnných v úloze určíme tak, že k základním proměnným z účelové funkce g přičteme počet omezení a umělé proměnné. Těchto umělých proměnných bývá právě tolik, kolik je nerovností typu $\geq$. Ke každému omezení přidáváme, stejně jako v předchozí úloze, odpovídající koeficienty. V případě omezení typu $\geq$ se ke přidatné proměnné přidává koeficient -1 a koeficient 1 u proměnné umělé.

²⁵ Umělé proměnné jsou zde označeny jako z

Celkový počet proměnných: $3 + 2 + 2 = 7$

Kanonický tvar:

$$3y_1 + 4y_3 - y_4 + z_1 = 3$$

$$2y_1 + y_2 - y_5 + z_2 = 5$$

Účelová funkce g se tentokrát zapisuje do předposledního řádku simplexové tabulky, vynásobením všech koeficientů -1 a do posledního řádku se přidává pomocná funkce g' v anulovaném tvaru podle vztahu (2.5).

Kde z_1 a z_2 označují zavedené umělé proměnné, které jsou vyjádřeny z kanonického tvaru:

$$z_1 = 3 - 3y_1 - 4y_3 + y_4$$

$$z_2 = 5 - 2y_1 + y_2 + y_5$$

Anulovaný tvar pomocné účelové funkce: $g' + 5y_1 + y_2 + 4y_3 - y_4 - y_5 = 8$

Tab. 24: Výchozí tvar simplexové tabulky příkladu (3.2)

	y_1	y_2	y_3	y_4	y_5	z_1	z_2	b_i
	3	0	4	-1	0	1	0	3
	2	1	0	0	-1	0	1	5
g	-10	-15	-20	0	0	0	0	0
g'	5	1	4	-1	-1	0	0	8

Podobně jako v předchozím příkladu je po sestavení tabulky určen klíčový sloupec a řádek. U minimalizační úlohy je sloupec volen podle **největšího kladného** koeficientu v řádku pomocné funkce, řádek podle nejmenšího kladného podílu pravé strany a hodnot v klíčovém sloupci.

Tab. 25: Nalezení klíčového sloupce a řádku Tab. 24

		y_1	y_2	y_3	y_4	y_5	z_1	z_2	b_j	$t = b_i/a_{mn}$
1. ř.		3	0	4	-1	0	1	0	3	1
2. ř.		2	1	0	0	-1	0	1	5	2,5
3. ř.	g	-10	-15	-20	0	0	0	0	0	-
4. ř.	g'	5↑	1	4	-1	-1	0	0	8	-

Klíčový prvek se určí jako průsečík příslušného sloupce a řádku, upraví se na jedničku a ostatní hodnoty ve sloupci se vynulují pomocí Gaussovy eliminace. I. fáze pokračuje, dokud hodnota pomocné účelové funkce není rovna nule.

Tab. 26: Výpočet první fáze příkladu (3.2)

1. iterace		y_1	y_2	y_3	y_4	y_5	z_1	z_2	b_j	$t = b_i/a_{mn}$
5. ř.		1	0	4/3	-1/3	0	1/3	0	1	-
6. ř.		0	1	-8/3	2/3	-1	-2/3	1	3	3
7. ř.	g	0	-15	-20/3	-10/3	0	10/3	0	10	-
8. ř.	g'	0	1↑	-8/3	2/3	-1	-5/3	0	3	-
2. iterace										
9. ř.		1	0	4/3	-1/3	0	1/3	0	1	-
10. ř.		0	1	-8/3	2/3	-1	-2/3	1	3	-
11. ř.	g	0	0	-140/3	20/3	-15	-20/3	15	55	-
12. ř.	g'	0	0	0	0	0	-1	-1	0	-

Detailní způsob výpočtu 1. iterace (Tab. 25 – Tab. 26):

- Transformovaný klíčový řádek: 5. ř. = 1. ř. / 3
- 6. ř. = $-2 * 5. \text{ř.} + 2. \text{ř.}$
- Řádek účelové funkce g : 7. ř. = $10 * 5. \text{ř.} + 3. \text{ř.}$
- Řádek pomocné funkce g' : 8. ř. = $-5 * 5. \text{ř.} + 4. \text{ř.}$

Detailní způsob výpočtu 2. iterace (pouze z Tab. 26):

- Transformovaný klíčový řádek: 10. ř. = 6. ř. / 1
- 9. ř. = $0 * 10. \text{ř.} + 5. \text{ř.}$
- Řádek účelové funkce g : 11. ř. = $15 * 10. \text{ř.} + 7. \text{ř.}$
- Řádek pomocné funkce g' : 12. ř. = $-1 * 10. \text{ř.} + 8. \text{ř.}$

Jakmile je hodnota pomocné účelové funkce rovna nule, je úloha považována za přípustnou a I. fáze dvoufázové simplexové metody je tímto dokončena. Pomocné proměnné a pomocná účelová funkce již dále nehraje roli ve výpočtu a může být z tabulky vynechána.

Po ukončení první fáze následuje přechod ke II. fázi, v níž se pokračuje v iteracích s minimalizační verzí původní účelové funkce, a to stejným způsobem jako ve fázi první.

Tab. 27: Výpočet druhé fáze příkladu (3.2)

2. iterace		y_1	y_2	y_3	y_4	y_5	b_j	$t = b_i/a_{mn}$
9. ř.		1	0	4/3	-1/3	0	1	-
10. ř.		0	1	-8/3	2/3	-1	3	4,5
11. ř.	g	0	0	-140/3	20/3↑	-15	55	-

3. iterace							
13. ř.		1	1/2	0	0	-1/2	5/2
14. ř.		0	3/2	-4	1	-3/2	9/2
15. ř.	g	0	-10	-20	0	-5	25

Detailní způsob výpočtu 3. iterace (pouze z Tab. 27):

- Transformovaný klíčový řádek: 14. ř. = 10. ř. * 3/2
- 13. ř. = 1/3 * 14. ř. + 9. ř.
- Řádek účelové funkce g : 15. ř. = $-20/3$ * 14. ř. + 11. ř.

Výsledné řešení příkladu:

$y^{opt} = (\frac{5}{2}; 0; 0; \frac{9}{2}; 0)$, kde hodnoty v bázi jsou $y_1 = 2,5; y_4 = 4,5$;

Redukované ceny proměnných y_2 a y_3 jsou kladné, zařazení do báze by nezlepšilo hodnotu účelové funkce;

Stínové ceny (duální hodnoty): $u_1 = 0, u_2 = -5$;

Úloha má jediné optimální řešení;

Optimální hodnota účelové funkce je $g = 25$;

Kontrola správnosti:

$$\min f = 10x_1 + 15x_2 + 20x_3 \rightarrow 25 = 10 * \frac{5}{2} + 0 + 0 \rightarrow 25 = 25;$$

$$3x_1 + 4x_3 \geq 3 \rightarrow 3 * \frac{5}{2} + 4 * 0 \geq 3 \rightarrow 7,5 \geq 3 \text{ (nerovnost)}^{26};$$

$$2x_1 + x_2 \geq 5 \rightarrow 2 * \frac{5}{2} \leq 5 \rightarrow 5 \geq 5 \text{ (rovnost)};$$

Převod na duální úlohu:

$$c = \begin{pmatrix} 3 \\ 5 \end{pmatrix} \rightarrow c^T = (3, 5)$$

$$A^T = \begin{pmatrix} 3 & 0 & 4 \\ 2 & 1 & 0 \end{pmatrix} \rightarrow A = \begin{pmatrix} 3 & 2 \\ 0 & 1 \\ 4 & 0 \end{pmatrix}$$

$$b^T = (10 \quad 15 \quad 20) \rightarrow b = \begin{pmatrix} 10 \\ 15 \\ 20 \end{pmatrix}$$

²⁶ $y_4 = 4,5$ představuje překročení a po odečtení bude rovnost ($3 = 3$).

Tab. 13 ukazuje, že omezení typu $\geq$ v primární úloze odpovídá duální proměnné $y_i \leq 0$. Pokud chceme hodnotu této proměnné převést na kladnou, můžeme ji vynásobit -1 , což lze aplikovat i na stínovou cenu $u_2 = -5 \xrightarrow{-1} 5$.

Protože se jedná o duální variantu úlohy řešené v příkladu (3.1) pomocí simplexové metody, platí mezi oběma modely přímé vazby:

- Duální proměnné y_i odpovídá stínové ceně u_m z primární úlohy (3.1):

$$y_1 = u_1 = 2,5$$

- Stínová cena u_m odpovídá základní proměnné x_i z primární úlohy (3.1):

$$u_2 = x_2 = 5$$

- Optimální hodnota účelové funkce je v obou úlohách shodná:

$$g = f = 25$$

Duální úloha představuje zpětnou transformaci primární úlohy (3.1), jak znázorňuje Obrázek 6;

Převod na duální úlohu (3.2):

Primární úloha (3.1):

$$\max f(x) = 3x_1 + 5x_2$$

$$\max f(x) = 3x_1 + 5x_2$$

$$3x_1 + 2x_2 \leq 10$$

$$3x_1 + 2x_2 \leq 10$$

$$x_2 \leq 15$$

$$x_2 \leq 15$$

$$4x_1 \leq 20$$

$$4x_1 \leq 20$$

$$x_j \geq 0$$

$$x_j \geq 0$$

3.1.3 Maximalizační úloha s omezeními ve tvaru =

Vlastní příklad pro procvičování:

$$\max f(x) = 15x_1 + 10x_2 \quad (3.3)$$

$$\text{za podmíněk: } 5x_1 + 8x_2 = 12$$

$$5x_1 + 15x_2 = 20$$

$$x_j \geq 0$$

V tomto příkladu se vyskytují pouze omezení ve tvaru rovnosti. Tato omezení nelze převést do kanonického tvaru pomocí přídatných proměnných, protože rovnost již nemá

směr nerovnosti, který by bylo třeba upravovat. Z tohoto důvodu se v úloze nezavádějí žádné přídatné proměnné, ale výhradně umělé, které umožní vytvořit výchozí simplexovou tabulku s jednotkovými vektory.

Pro řešení je opět nutné využít dvoufázovou simplexovou metodu, kde v I. fázi slouží pomocná účelová funkce k odstranění umělých proměnných z báze.

Celkový počet proměnných se určuje jako součet počtu proměnných z účelové funkce a počtu rovností, přičemž počet umělých proměnných odpovídá právě počtu rovnostních omezení. Podobně jako v předchozím příkladu se ke každému rovnostnímu omezení přidává umělá proměnná s koeficientem 1

Celkový počet proměnných: $2 + 3 = 5$

Kanonický tvar:

$$5x_1 + 8x_2 + y_1 = 15$$

$$5x_1 + 15x_2 + y_2 = 20$$

Pomocná účelová funkce se sestavuje stejným způsobem jako v předchozím příkladu, minimalizace pomocné účelové funkce v anulovaném tvaru pomocí vztahu:

$$f' = y_1 + y_2 \rightarrow \min$$

Tab. 28: Výchozí tvar simplexové tabulky příkladu (3.3)

	x_1	x_2	y_1	y_2	b_i
	5	8	1	0	12
	5	15	0	1	20
f	-15	-10	0	0	0
f'	10	23	0	0	32

Výpočet probíhá standardním způsobem – po úpravě klíčového řádku se ostatní řádky transformují tak, aby vznikla jednička v klíčovém prvku a nuly ve zbytku sloupce. Iterace v I. fázi pokračují, dokud se v hodnota pomocné účelové funkce nerovná 0.

Tab. 29: Nalezení klíčového sloupce a řádku Tab. 28

		x_1	x_2	y_1	y_2	b_j	$t = b_i/a_{mn}$
1.		5	8	1	0	12	1,5
2.		5	15	0	1	20	1,33
3.	f	-15	-10	0	0	0	-
4.	f'	10	23↑	0	0	32	-

Tab. 30: Výpočet první fáze

1. iterace		x_1	x_2	y_1	y_2	b_j	$t = b_i/a_{mn}$
5.		7/3	0	1	-8/15	4/3	0,57
6.		1/3	1	0	1/15	4/3	4
7.	f	-35/3	0	0	2/3	40/3	-
8.	f'	7/3↑	0	0	-23/15	4/3	-
2. iterace							
9.		1	0	3/7	-8/35	4/7	-
10.		0	1	-1/7	1/7	8/7	-
11.	f	0	0	5	-2	20	-
12.	f'	0	0	-1	-1	0	-

Detailní způsob výpočtu 1. iterace (Tab. 29 – Tab. 30):

- Transformovaný klíčový řádek: 6. ř. = 2. ř. / 15
- 5. ř. = $-8 * 6. \text{ř.} + 1. \text{ř.}$
- Řádek účelové funkce f : 7. ř. = $10 * 6. \text{ř.} + 3. \text{ř.}$
- Řádek pomocné funkce f' : 8. ř. = $-23 * 6. \text{ř.} + 4. \text{ř.}$

Detailní způsob výpočtu 2. iterace (Tab. 30):

- Transformovaný klíčový řádek: 9. ř. = 5. ř. * (3/7)
- 10. ř. = $-1/3 * 9. \text{ř.} + 6. \text{ř.}$
- Řádek účelové funkce f : 11. ř. = $35/3 * 9. \text{ř.} + 7. \text{ř.}$
- Řádek pomocné funkce f' : 12. ř. = $-7/3 * 9. \text{ř.} + 8. \text{ř.}$

Na základě výstupu po druhé iteraci lze si všimnout, že řádek pomocné účelové funkce f' , již neobsahuje žádné kladné hodnoty kromě sloupce umělých proměnných, což značí ukončení první fáze dvoufázové simplexové metody.

Tab. 31: Druhá fáze a optimální hodnota

	x_1	x_2	y_1	y_2	b_i
	1	0	3/7	-8/35	4/7
	0	1	-1/7	1/7	8/7
f	0	0	5	-2	20

Z Tab. 31 je patrné, že v řádku původní účelové funkce f , kromě sloupce umělých proměnných, se nenachází žádná záporná hodnota. To znamená, že jsou splněny podmínky optimality pro druhou fázi výpočtu a není třeba pokračovat v dalších iteracích. Pro určení stínových cen se ponechává sloupec s umělými proměnnými.

Výsledné řešení příkladu:

$x^{opt} = (\frac{4}{7}; \frac{8}{7})$, kde nenulové hodnoty jsou $x_1 \approx 0,571$; $x_2 \approx 1,142$

Stínové ceny (duální hodnoty): $u_1 = 5$, $u_2 = -2$

Úloha má jediné optimální řešení;

Optimální hodnota účelové funkce je $f = 20$.

Kontrola správnosti:

$$\max f = 15x_1 + 10x_2 \rightarrow 20 = 15 * \frac{4}{7} + 10 * \frac{8}{7} \rightarrow 20 = 20$$

$$5x_1 + 8x_2 = 12 \rightarrow 5 * \frac{4}{7} + 8 * \frac{8}{7} = 12 \rightarrow 12 = 12 \text{ (rovnost)}$$

$$5x_1 + 15x_2 = 20 \rightarrow 5 * \frac{4}{7} + 15 * \frac{8}{7} \rightarrow 20 = 20 \text{ (rovnost)}$$

Převod na duální úlohu:

Duální úloha není jednoznačná, protože všechna omezení v primární úloze jsou rovnosti. Jde o nesouměrnou dualitu, kdy chybí běžný vztah mezi typem omezení a znaménkem duálních proměnných. Formulace duální úlohy je možné řešit dvěma způsoby:

1. způsob:

Řešení spočívá v tom, že rovnosti lze nahradit dvěma nerovnostmi ($\leq$ a $\geq$), což odpovídá právě tomu, že jejich duální proměnná není vázána znaménkovým omezením. Pro získání správného směru nerovnosti v kontextu duální formulace stačí upravit nevhodné omezení vynásobením celého vztahu konstantou -1 .

$$\max f = 15x_1 + 10x_2$$

$$5x_1 + 8x_2 \leq 12$$

$$5x_1 + 8x_2 \geq 12 \xrightarrow{-1} -5x_1 + 8x_2 \leq -12$$

$$5x_1 + 15x_2 \leq 20$$

$$5x_1 + 15x_2 \geq 20 \xrightarrow{-1} -5x_1 - 15x_2 \leq -20$$

$$x_j \geq 0$$

Takto je zachován formát primární úlohy odpovídající standardní souměrné dualitě.

$$c = \begin{pmatrix} 12 \\ -12 \\ 20 \\ -20 \end{pmatrix} \rightarrow c^T = (12 \quad -12 \quad 20 \quad -20)$$

$$A^T = \begin{pmatrix} 5 & 8 \\ -5 & -8 \\ 5 & 15 \\ -5 & -15 \end{pmatrix} \rightarrow A = \begin{pmatrix} 5 & -5 & 5 & -5 \\ 8 & -8 & 15 & -15 \end{pmatrix}$$

$$b^T = (15 \quad 10) \rightarrow b = \begin{pmatrix} 15 \\ 10 \end{pmatrix}$$

Formulace duální úlohy:

$$\min g(y) = 12y_1 - 12y_2 + 20y_3 - 20y_4$$

$$5y_1 - 5y_2 + 5y_3 - 5y_4 \geq 15$$

$$8y_1 - 8y_2 + 15y_3 - 15y_4 \geq 10$$

$$y_j \geq 0$$

2. způsob:

Duální úlohu lze sestavit přímo podle vztahu s primární úlohou (Tab. 13), kde omezení v primární odpovídá proměnná v duální a naopak. Rovnosti v primární úloze vedou k volným duálním proměnným (URS) a nezápornost primárních proměnných znamená, že příslušná duální omezení mají tvar $\geq$.

Na základě těchto pravidel lze přímo sestavit duální úlohu bez nutnosti převádět rovnosti na dvě nerovnosti:

$$\min g(y) = 12y_1 + 20y_2$$

$$5y_1 + 5y_2 \geq 15$$

$$8y_1 + 15y_2 \geq 10$$

$$y_1, y_2 \text{ URS}$$

Proměnná URS se reprezentují jako rozdíl dvou nezáporných proměnných: $y_j = y_j^+ - y_j^-$

kde:

- y_j^+ označuje **kladnou složku** proměnné
- y_j^- označuje **zápornou složku** proměnné.

A při dosazení dostaneme:

$$\min g(y) = 12(y_j^+ - y_j^-) + 20(y_j^+ - y_j^-)$$

$$5(y_j^+ - y_j^-) + 5(y_j^+ - y_j^-) \geq 15$$

$$8(y_j^+ - y_j^-) + 15(y_j^+ - y_j^-) \geq 10$$

Model sestavený druhým způsobem je zcela ekvivalentní výsledku získanému prvním postupem.

3.1.4 Maximalizační úloha s kombinovanými omezeními

Vlastní příklad pro procvičování:

$$\max f(x) = 6x_1 + x_2 + 2x_3 \quad (3.4)$$

$$\text{za podmínek: } -x_1 + 3x_2 + 5x_3 \leq 6$$

$$x_1 - 3x_2 + 6x_3 = 6$$

$$x_1 + x_2 + 2x_3 \geq 1$$

$$x_j \geq 0$$

Tab. 32: Simplexová metoda příkladu (3.4)

		x_1	x_2	x_3	x_4	x_5	y_1	y_2	b_j	$t = b_i/a_{mn}$
1.		-1	3	5	1	0	0	0	6	1,2
2.		1	-3	6	0	0	1	0	6	1
3.		1	1	2	0	-1	0	1	1	0,5
4.	f	-6	-1	2	0	0	0	0	0	-
5.	f'	2	-2	8↑	0	-1	0	0	7	-
6.		-7/2	1/2	0	1	5/2	0	-5/2	7/2	1,4
7.		-2	-6	0	0	3	1	-3	3	1
8.		1/2	1/2	1	0	-1/2	0	1/2	1/2	-
9.	f	-7	-2	0	0	1	0	-1	-1	-
10.	f'	-2	-6	0	0	3↑	0	-4	3	-
11.		-11/6	11/2	0	1	0	-5/6	0	1	-
12.		-2/3	-2	0	0	1	1/3	-1	1	-
13.		1/6	-1/2	1	0	0	1/6	0	1	6
14.	f	-19/3↑	0	0	0	0	-1/3	0	-2	-
15.	f'	0	0	0	0	0	-1	-1	0	-
16.		0	0	11	1	0	1	0	12	-
17.		0	-4	4	0	1	1	-1	5	-
18.		1	-3	6	0	0	1	0	6	-

19.	f	0	-19↑	38	0	0	6	0	36	-
-----	-----	---	------	----	---	---	---	---	----	---

Zápis konkrétních hodnot do tabulky probíhal následujícím způsobem Tab. 32:

- 1. – 5. ř.: Výchozí tvar simplexové tabulky
- 1. iterace (6. – 10. ř.)
 - 8.ř. = 3.ř. / 2
 - 6.ř. = $-5 * 8.ř. + 1.ř.$
 - 7.ř. = $-6 * 8.ř. + 2.ř.$
 - 9.ř. = $-2 * 8.ř. + 4.ř.$
 - 10.ř. = $-8 * 8.ř. + 5.ř.$
- 2. iterace (11. – 15. ř.)
 - 12.ř. = 7.ř. / 3
 - 11.ř. = $-\frac{5}{2} * 12.ř. + 6.ř.$
 - 13.ř. = $\frac{1}{2} * 12.ř. + 8.ř.$
 - 14.ř. = $-1 * 12.ř. + 9.ř.$
 - 15.ř. = $-3 * 12.ř. + 10.ř.$
- 3. iterace (16. – 19. ř.)
 - 18.ř. = 13.ř. * 6
 - 16.ř. = $\frac{11}{6} * 18.ř. + 11.ř.$
 - 17.ř. = $\frac{2}{3} * 18.ř. + 12.ř.$
 - 19.ř. = $\frac{19}{3} * 18.ř. + 14.ř.$

Úloha je bez omezení (*unbounded solution*), neboť v dané iteraci nelze určit klíčový řádek. Hodnota účelové funkce tak může růst do nekonečna, aniž by došlo k porušení některého z omezení.

3.1.5 Minimalizační úloha s kombinovanými omezeními

$$\min f = 8x_1 + 5x_2 + x_3 \quad (3.5)$$

$$\text{za podmínek: } x_1 + 3x_2 + x_3 \leq 8$$

$$2x_1 + 2x_2 = 3$$

$$2x_1 + 4x_2 + 3x_3 \geq 5$$

$$x_{1,2} \geq 0, \quad x_3 \text{ URS}$$

Tab. 33: Simplexová metoda příkladu (3.5)

		x_1	x_2	x_3^+	x_3^-	x_4	x_5	y_1	y_2	b_j	$t = b_i/a_{mn}$
1.		1	3	1	-1	1	0	0	0	8	2,66
2.		2	2	0	0	0	1	0	0	6	1,5
3.		2	4	3	-3	0	-1	0	1	5	1,25
4.	f	-8	-5	-1	1	0	0	0	0	0	-
5.	f'	0	6↑	3	-3	0	-1	0	0	8	-
6.		-1/2	0	-5/4	5/4	1	3/4	0	-3/4	17/4	3,4
7.		1	0	-3/2	3/2	0	1/2	1	-1/2	1/2	0,33
8.		1/2	1	3/4	-3/4	0	-1/4	0	1/4	5/4	-
9.	f	-11/2	0	11/4	-11/4	0	-5/4	0	5/4	25/4	-
10.	f'	1	0	-3/2	3/2↑	0	1/2	0	-3/2	1/2	-
11.		-4/3	0	0	0	1	1/3	-1/6	-	23/6	-
12.		2/3	0	-1	1	0	1/3	2/3	-	1/3	-
13.		1	1	0	0	0	0	1/2	-	3/2	-
14.	f	-11/3	0	0	0	0	-1/3	11/6	-	43/6	-

Zápis konkrétních hodnot do tabulky probíhal následujícím způsobem Tab. 33:

- 1. – 5. ř.: Výchozí tvar simplexové tabulky
- 1. iterace (6. – 10. ř.)
 - 8.ř. = 3.ř. / 4
 - 6.ř. = -3 * 8.ř. + 1.ř.
 - 7.ř. = -2 * 8.ř. + 2.ř.
 - 9.ř. = 5 * 8.ř. + 4.ř.
 - 10.ř. = -6 * 8.ř. + 5.ř.

- 2. iterace (11. – 14. ř.)

$$\circ \quad 12.\text{ř.} = 7.\text{ř.} * \frac{2}{3}$$

$$\circ \quad 11.\text{ř.} = -\frac{5}{4} * 12.\text{ř.} + 6.\text{ř.}$$

$$\circ \quad 13.\text{ř.} = \frac{3}{4} * 12.\text{ř.} + 8.\text{ř.}$$

$$\circ \quad 14.\text{ř.} = \frac{11}{4} * 12.\text{ř.} + 9.\text{ř.}$$

Výsledné řešení příkladu:

$$x_2 = \frac{3}{2}; \quad x_3 = 0 - \frac{1}{3} = -\frac{1}{3}; \quad x_4 = \frac{23}{6}; \quad x_1 = x_5 = 0;$$

Redukovaná cena proměnné x_1 je záporná, její zařazení do báze by nezlepšilo hodnotu účelové funkce;

$$\text{Stínové ceny (duální hodnoty): } u_1 = 0, u_2 = \frac{11}{6}, u_3 = -\frac{1}{3}$$

Úloha má jediné optimální řešení;

$$\text{Optimální hodnota účelové funkce je } f = \frac{43}{6} \approx 7,17.$$

Kontrola správnosti:

$$\min f = 8x_1 + 5x_2 + x_3 \rightarrow \frac{43}{6} = 0 + 5 * \frac{3}{2} - \frac{1}{3} \rightarrow \frac{43}{6} = \frac{43}{6}$$

$$x_1 + 3x_2 + x_3 \leq 8 \rightarrow 0 + 3 * \frac{3}{2} - \frac{1}{3} \leq 8 \rightarrow 4,16 \leq 8 \text{ (nerovnost)}^{27}$$

$$2x_1 + 2x_2 = 3 \rightarrow 0 + 2 * \frac{3}{2} = 3 \rightarrow 3 = 3 \text{ (rovnost)}$$

$$2x_1 + 4x_2 + 3x_3 \geq 5 \rightarrow 0 + 4 * \frac{3}{2} + 3 * \left(-\frac{1}{3}\right) \rightarrow 5 \geq 5 \text{ (rovnost)}$$

Duální model:

$$\max g(y) = -8y_1 + 3y_2 + 5y_3$$

$$\text{za podmínek: } -y_1 + 2y_2 + 4y_3 \leq 8$$

$$-3y_1 + 2y_2 + 4y_3 \leq 5$$

$$-2y_1 + 3y_3 = 1$$

$$y_{1,3} \geq 0, \quad y_2 \text{ URS}$$

²⁷ $x_4 = \frac{23}{6}$ představuje nevyužitou kapacitu a po doplnění bude rovnost ($8 = 8$).

4 APLIKACE METOD V PRAXI

Cílem této podkapitoly je ukázat, jak lze metody LP aplikovat při řešení praktických problémů. V mnoha oblastech rozhodování, kde je nutné pracovat s omezenými zdroji či optimalizovat provozní procesy, slouží lineární modely jako efektivní prostředek pro nalezení nejlepšího možného řešení.

Sestavení takového modelu představuje základní předpoklad pro další řešení úlohy. Zadaný problém je nutné převést do matematické formy, v níž účelová funkce vyjadřuje cíl optimalizace. Omezující podmínky pak zachycují reálná omezení, zahrnující jak samotné nerovnosti či rovnosti, tak i podmínky nezápornosti jednotlivých proměnných. [17]

Na vybraných typech úloh si ukážeme, jak lze metody LP uplatnit v konkrétní praxi.

- problém alokace zdrojů
- nutriční problém
- optimalizace portfolia
- dělení materiálu
- dopravní problém

Optimální řešení bylo nalezeno pomocí vlastního programu implementovaného v jazyce Python, jehož popis je uveden v podkapitole 5.2.

4.1 Problém alokace zdrojů

Ukázková úloha byla převzata a přeložena ze zdroje [18].

Farmář plánuje využít svá pole k pěstování kukuřice a pšenice. Kukuřice má výnos 113,5 bušlů na akr a prodává se za 3,15 dolarů za bušl. Pšenice má výnos 35,8 bušlů na akr a prodává se za 4,45 dolarů za bušl.

Omezení farmáře jsou následující:

1. Celková plocha kukuřice a pšenice nesmí přesáhnout 120 akrů.
2. Kukuřice musí být zasazena na alespoň 20 akrech, nejvýše však na 80 akrech.
3. Pšenice musí být zasazena na alespoň 30 akrech.

Úkolem je zjistit, kolik akrů kukuřice a pšenice má farmář zasadit, aby maximalizoval svůj příjem.

Tab. 34: Přehled parametrů pro pěstování kukuřice a pšenice

Plodina	Výnos (bušlů/akr)	Cena (\$/bušel)	Min akrů	Max akrů
Kukuřice	113,5	3,15	20	80
Pšenice	35,8	4,45	30	-
Celková plocha [akr]	-	-	120	

(zdroj: [18])

Sestavení matematického modelu úlohy:

Pro účely sestavení matematického modelu definujeme následující proměnné:

- x_1 : počet akrů osázených kukuřicí
- x_2 : počet akrů osázených pšenicí

Účelová funkce vyjadřuje celkový zisk farmáře z prodeje obou plodin a jejím cílem je maximalizace tohoto zisku. Jednotkový zisk z jednoho akru každé plodiny je určen jako součin jejího výnosu a prodejní ceny. Účelovou funkci lze sestavit ve tvaru:

$$\max f(x) = x_1(113,5 * 3,15) + x_2(35,8 * 4,45) = 357,525x_1 + 159,31x_2$$

Omezující podmínky získáme na základě specifikovaných požadavků farmáře na využití půdy:

$$x_1 + x_2 \leq 120$$

$$20 \leq x_1 \leq 80$$

$$x_2 \geq 30$$

V některých případech vedou k intervalovým podmínkám, které je nutné pro potřeby LP rozdělit na dvě samostatné lineární nerovnice. Z podmínky omezení $20 \leq x_1 \leq 80$, což odpovídá požadavku, aby bylo oseto alespoň 20 a nejvýše 80 akrů kukuřicí, je třeba vzorec přepsat ve formátu dvou nerovností:

$$x_1 \geq 20$$

$$x_1 \leq 80$$

Rozhodovací proměnné představují počet akrů osázených jednotlivými plodinami, nemohou nabývat záporných hodnot. Platí tedy:

Optimální řešení:

$$x^{opt} = (80; 40; 0; 60; 0; 10)$$

$$f^{opt} = 34\,974,40 \$$$

Pro každý vrchol F_i konvexní přípustné oblasti lze dopočítat hodnotu účelové funkce. Tab. 36 uvádí výsledky pro všechny vrcholy a tučně je zvýrazněna ta hodnota, která je maximální.

Tab. 36: Vyhodnocení zisku pro jednotlivé vrcholy přípustné oblasti

Vrchol	Souřadnice (x_1, x_2)	Účelová funkce	Celkový zisk (\$)
F_1	(20,30)	$357,525 \cdot 20 + 159,31 \cdot 30$	11 929,80
F_2	(80,30)	$357,525 \cdot 80 + 159,31 \cdot 30$	33 381,30
F_3	(80,40)	$357,525 \cdot 80 + 159,31 \cdot 40$	34 947,40
F_4	(20,100)	$357,525 \cdot 20 + 159,31 \cdot 100$	23 081,50

(zdroj: vlastní)

Na základě hodnot uvedených v Tab. 36 lze odečíst, že maximálního zisku dosáhne farmář při volbě varianty F_3 , tedy pokud osází 80 akrů kukuřicí a 40 akrů pšenicí. Celkový zisk v tomto případě činí 34 974,40 \$, což představuje optimální řešení dané úlohy (Tab. 35).

Hodnoty přídatných proměnných v bázi ukazují, že $x_4 = 60$, což znamená, že kukuřice dosáhla hodnoty 80, tedy přesně horní hranice 3. omezení (4.1.3), zároveň byla o 60 jednotek nad minimem stanoveným 2. omezením (4.1.2). Podobně hodnota $x_6 = 10$ ukazuje, že pšenice dosáhla hodnoty 40, což znamená překročení minimální požadované výměry stanovené v omezení (4.1.4) o 10 akrů.

Stínové ceny naznačují, že z pohledu optimalizace je výhodnější zaměřit se na úpravu 3. omezení (4.1.3). Do budoucna by farmář mohl zvážit rozšíření maximální plochy určené pro pěstování kukuřice o 10 akrů, které jsou v současném řešení využity pro pšenici. Tímto krokem by bylo možné zvýšit celkový zisk o $198,215 \cdot 10 = 1\,982,15 \$$, přičemž ostatní omezení by zůstala nezměněna.

4.2 Nutriční problém

Ukázková úloha byla převzata a upravena ze zdroje [1].

Denní dávka výživy pro jistou skupinu dospělých osob by měla mít energetickou hodnotu minimálně 12 000 kJ, měla by obsahovat minimálně 80 g bílkovin, 15 mg železa,

6 000 jednotek vitamínu A a 200 mg vitamínu C. Pro zabezpečení uvedených požadavků je k dispozici devět základních druhů potravin. Jejich složení z hlediska uvedených výživových komponent (vždy na 100 g dané potraviny) a jejich cena v Kč za 100 g jsou uvedeny v následující tabulce. Formulujte úlohu lineárního programování tak, aby denní skladba výživy byla co nejlevnější a aby byly uspokojeny všechny výše uvedené požadavky.

Tab. 37: Výživové hodnoty a ceny jednotlivých potravin

Potravina	Energie [kJ]	Bílkoviny [g]	Železo [mg]	Vitamín A [jednotka]	Vitamín C [mg]	Cena
Vepřové maso	1 200	18,4	3,1	20	0	12,00
Máslo	3 000	0,6	0,2	2 500	0	11,20
Chléb	1 160	7,2	0,8	0	0	1,50
Brambory	300	1,6	0,6	40	10	1,20
Jablka	240	0,0	0,5	60	2	1,50
Sýr eidam	1 250	31,2	0,6	1 100	0	10,60
Kuře	650	20,2	1,5	0	0	6,00
Jogurt	450	7,0	0,2	260	0	4,50
Jahody	150	0,0	0,8	60	60	12,00

(zdroj:[1])

Sestavení matematického modelu úlohy:

Pro účely sestavení matematického modelu definujeme následující proměnné:

- x_1 : množství vepřového masa ve 100 g potravy
- x_2 : množství másla ve 100 g potravy
- x_3 : množství chleba ve 100 g potravy
- x_4 : množství brambor ve 100 g potravy
- x_5 : množství jablek ve 100 g potravy
- x_6 : množství sýru eidam ve 100 g potravy
- x_7 : množství kuřete ve 100 g potravy
- x_8 : množství jogurtu ve 100 g potravy
- x_9 : množství jahod ve 100 g potravy

Cílem je minimalizovat celkovou cenu denní stravy. Cena každé potraviny je uvedena (Tab. 37) za 100 g. Účelová funkce tedy nabývá tvaru:

$$\min f(x) = 12x_1 + 11,2x_2 + 1,5x_3 + 1,2x_4 + 1,5x_5 + 10,6x_6 + 6x_7 + 4,5x_8 + 12x_9$$

Omezení vyplývají z požadavků na minimální denní příjem živin:

Minimálně 12 000 kJ energetické hodnoty:

$$1\,200x_1 + 3\,000x_2 + 1\,160x_3 + 300x_4 + 240x_5 + 1\,260x_6 + 650x_7 + 450x_8 + 150x_9 \geq 12\,000$$

Minimálně 80 g bílkovin:

$$18,4x_1 + 0,6x_2 + 7,2x_3 + 1,6x_4 + 31,2x_6 + 20,2x_7 + 7x_8 \geq 80$$

Minimálně 15 mg železa:

$$3,1x_1 + 0,2x_2 + 0,8x_3 + 0,6x_4 + 0,5x_5 + 0,6x_6 + 1,5x_7 + 0,2x_8 + 0,8x_9 \geq 15$$

Minimálně 6 000 jednotek vitamínu A:

$$20x_1 + 2\,500x_2 + 40x_4 + 60x_5 + 1\,100x_6 + 260x_8 + 60x_9 \geq 6\,000$$

Minimálně 200 mg vitamínu C:

$$10x_4 + 2x_5 + 60x_9 \geq 200$$

Rozhodovací proměnné představují množství potravin, nemohou nabývat záporných hodnot:

$$x_i \geq 0 \text{ pro } i = 1, 2, \dots, 9$$

Formulace slovní úlohy jako lineární model:

$$\begin{aligned} \min f(x) = & 12x_1 + 11,2x_2 + 1,5x_3 + 1,2x_4 + \\ & 1,5x_5 + 10,6x_6 + 6x_7 + 4,5x_8 + 12x_9 \end{aligned} \quad (4.2)$$

za podmínek:

$$\begin{aligned} 1\,200x_1 + 3\,000x_2 + 1\,160x_3 + 300x_4 + 240x_5 + 1\,260x_6 + 650x_7 \\ + 450x_8 + 150x_9 \geq 12\,000 \end{aligned} \quad (4.2.1)$$

$$18,4x_1 + 0,6x_2 + 7,2x_3 + 1,6x_4 + 31,2x_6 + 20,2x_7 + 7x_8 \geq 80 \quad (4.2.2)$$

$$\begin{aligned} 3,1x_1 + 0,2x_2 + 0,8x_3 + 0,6x_4 + 0,5x_5 + 0,6x_6 + 1,5x_7 + 0,2x_8 \\ + 0,8x_9 \geq 15 \end{aligned} \quad (4.2.3)$$

$$20x_1 + 2\,500x_2 + 40x_4 + 60x_5 + 1\,100x_6 + 260x_8 + 60x_9 \geq 6\,000 \quad (4.2.4)$$

$$10x_4 + 2x_5 + 60x_9 \geq 200 \quad (4.2.5)$$

$$x_i \geq 0$$

Tab. 38: Optimální řešení úlohy nutričního problému

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	b
	4,10	0	1	0	0,56	0	1,71	0,09	-3,86	0	0,02	-1,47	0	0,08	2,66
	-0,36	0	0	0	-0,14	1	0,25	0,20	0,59	0	-0,04	0,34	0	-0,01	0,89
	3606	0	0	0	540,4	0	1321	-40,4	-3077	1	30,5	-1723	-1,07	72,88	3270,71
	0,17	1	0	0	0,08	0	-0,11	0,02	-0,33	0	0,02	-0,15	0	0,01	1,69
	0	0	0	1	0,20	0	0	0	6	0	0	0	0	-0,1	20
f	-7,8	0	0	0	-1	0	-2	-2,05	-8,05	0	-0,18	-0,26	0	-0,06	56,35

(zdroj: vlastní)

Optimální řešení:

$$x^{opt} = (0; 1,69; 2,66; 20; 0; 0,89; 0; 0; 0; 3270,71; 0; 0; 0; 0; 0; 0)$$

$$f^{opt} = 56,35 \text{ Kč}$$

Ze zadání úlohy vyplývá, že hledané hodnoty odpovídají množství v přepočtu na 100 g dané potraviny.

$$x_i^{opt} * 100, \quad i = 1, 2, \dots, 9$$

$$x^{opt} = (0; 169; 266; 2000; 0; 89; 0; 0; 0; 3270,71; 0; 0; 0; 0; 0; 0)$$

$$\text{Celkové množství stravy: } 169 + 266 + 2000 + 89 = 2\,524 \text{ g}$$

Optimální složení denní výživy zahrnuje pouze máslo, chléb, brambory a sýr eidam. Po přepočtu na hmotnostní množství odpovídá výsledné řešení přibližně 169 g másla, 266 g chleba, 2 kg brambor a 89 g eidamu. Celkové množství stravy činí 2 524 g. Energetický příjem přesahuje stanovené minimum o 3 270,71 kJ, což potvrzuje, že zvolená kombinace potravin je z hlediska výživy dostatečná. Celkové náklady na tuto skladbu odpovídají částce 56,35 Kč.

Záporné redukováné ceny u vepřového masa, jablek, kuřecího masa, jogurtu a jahod naznačují, že tyto potraviny nejsou v porovnání s ostatními položkami dostatečně výhodné, jak z hlediska ceny, tak výživové hodnoty. Model proto upřednostňuje jiné potraviny, které dokážou požadované výživové složky pokrýt efektivněji a levněji.

4.3 Optimalizace portfolia

Ukázková úloha byla převzata ze zdroje [16].

Navrhnete optimální skladbu investičního portfolia, chcete-li investovat jeden milion Kč a máte-li na výběr z pěti druhů cenných papírů, u kterých je znám očekávaný roční výnos (v %) a též byla odhadnuta jejich rizikovost (koeficient z celočíselné stupnice 0–10 od nejnižší k nejvyšší rizikovosti), jak popisuje následující tabulka:

Tab. 39: Parametry cenných papírů [19]

Cenný papír	Očekávaný roční výnos (%)	Koeficient rizikovosti (0-10)
CP1	6,5	1
CP2	7	3
CP3	9	5
CP4	11	7
CP5	15	10

(zdroj: [19])

Přitom investor požaduje, aby CP1 nebo CP2 byly nakoupeny alespoň v hodnotě 400 000Kč, CP5 nejvýše v hodnotě 300 000Kč a celková míra rizika portfolia, definovaná jako vážený průměr koeficientů rizikovosti jednotlivých variant, nesmí přesáhnout 4.

Sestavení matematického modelu úlohy:

Pro účely sestavení matematického modelu definujeme následující proměnné:

- x_1 : Kapitál investovaný do CP1 (v Kč)
- x_2 : Kapitál investovaný do CP2 (v Kč)
- x_3 : Kapitál investovaný do CP3 (v Kč)
- x_4 : Kapitál investovaný do CP4 (v Kč)
- x_5 : Kapitál investovaný do CP5 (v Kč)

Cílem investora je maximalizace celkového ročního výnosu z investice. Očekávaný výnos každého cenného papíru je definován v procentech z investované částky (Tab. 39), a proto lze celkový výnos vyjádřit jako vážený součet dílčích výnosů jednotlivých investic. Účelovou funkci lze sestavit ve tvaru:

$$\max f(x) = 0,065x_1 + 0,07x_2 + 0,09x_3 + 0,11x_4 + 0,15x_5$$

Omezující podmínky vycházejí z požadavků investora na rozložení portfolia, míru rizika a celkovou výši investice:

K dispozici 1 000 000 Kč:

$$x_1 + x_2 + x_3 + x_4 + x_5 = 1\,000\,000$$

Investice do CP1 nebo CP2 musí činit alespoň 400 000 Kč:

$$x_1 + x_2 \geq 400\,000$$

Investice do CP5 musí činit nejvýše 300 000 Kč:

$$x_5 \leq 300\,000$$

Vážený průměr koeficientů rizikovosti jednotlivých variant, nesmí přesáhnout 4:

$$\frac{x_1 + 3x_2 + 5x_3 + 7x_4 + 10x_5}{x_1 + x_2 + x_3 + x_4 + x_5} \leq 4$$

po úpravě:

$$x_1 + 3x_2 + 5x_3 + 7x_4 + 10x_5 \leq 4(x_1 + x_2 + x_3 + x_4 + x_5)$$

$$x_1 + 3x_2 + 5x_3 + 7x_4 + 10x_5 \leq 4x_1 + 4x_2 + 4x_3 + 4x_4 + 4x_5$$

$$-3x_1 - x_2 + x_3 + 3x_4 + 6x_5 \leq 0$$

Rozhodovací proměnná představuje částku investovanou do daného cenného papíru v Kč, nemůže nabývat záporných hodnot:

$$x_i \geq 0 \text{ pro } i = 1, 2, 3, 4, 5$$

Formulace slovní úlohy jako lineární model:

$$\max f(x) = 0,065x_1 + 0,07x_2 + 0,09x_3 + 0,11x_4 + 0,15x_5 \quad (4.3)$$

za podmínek:

$$x_1 + x_2 + x_3 + x_4 + x_5 = 1\,000\,000 \quad (4.3.1)$$

$$x_1 + x_2 \geq 400\,000 \quad (4.3.2)$$

$$x_5 \leq 300\,000 \quad (4.3.3)$$

$$-3x_1 - x_2 + x_3 + 3x_4 + 6x_5 \leq 0 \quad (4.3.4)$$

$$x_i \geq 0$$

Tab. 40: Optimální řešení úlohy portfolia

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	b
	0	-0,333	0,333	0	0	1	0,5	-0,166	250 000
	1	0,666	0,333	0	0	0	0,5	-0,166	650 000
	0	0	0	0	1	0	1	0	300 000
	0	0,333	0,666	1	0	0	-1,5	0,166	50 000
f	0	0,01	0,005	0	0	0	0,0175	0,0075	92 750

(zdroj: vlastní)

Optimální řešení:

$$x^{opt} = (650\,000; 0; 0; 50\,000; 300\,000; 250\,000; 0; 0)$$

$$f^{opt} = 92\,750 \text{ Kč}$$

Z výsledku vyplývá, že optimální investiční portfolio se skládá z cenných papírů CP1 v hodnotě 650 000 Kč, CP4 ve výši 50 000 Kč a CP5 ve výši 300 000 Kč. Částka 250 000 Kč představuje překročení minimální požadované investice do CP1 a CP2. Takto sestavené portfolio přináší maximální možný očekávaný roční výnos ve výši 92 750 Kč.

Kladné redukované ceny u CP2 a CP3 naznačují, že za daných omezení nejsou tyto investice efektivní vzhledem k poměru očekávaného výnosu a rizika, a proto nejsou zahrnuty v optimálním portfoliu.

4.4 Dělení materiálů

Ukázková úloha byla převzata ze zdroje [14].

Podnik na předmětnou zakázku potřebuje tyče T1 a T2 o délce 80 cm a 90 cm. Vedení rozhodlo, že se podnik zbaví skladových zásob. Ve skladu jsou tyče délky 150 cm a 180 cm, a to v množství 200 a 240 kusů. Podnik potřebuje tyče T1 v množství aspoň 340 kusů a tyče T2 v počtu aspoň 300 kusů. Je nutné vyřešit, jakým způsobem nejvhodněji rozřezat tyče, aby řezný odpad byl co nejmenší.

Sestavení matematického modelu úlohy:

Pro sestavení řezného modelu je nejprve nutné identifikovat všechny možné způsoby, jak lze danou skladovou tyč délky 150 cm a 180 cm rozřezat na požadované délky výrobků, tedy 80 cm a 90 cm, a zároveň určit množství vzniklého odpadu.

- způsoby řezu 150cm tyče lze získat:

1. 1 ks tyče T1, odpad 70 cm
2. 1 ks tyče T2, odpad 60 cm

- způsoby řezu 180cm tyče lze získat:

1. 2 ks tyče T1, odpad 20 cm
2. 1 ks tyče T1 a T2, odpad 10 cm
3. 2 ks tyče T2, odpad 0 cm

Tab. 41: Možné způsoby řezu tyčí

	Tyč délky 150 cm		Tyč délky 180 cm		
Způsob řezu	1.	2.	1.	2.	3.
Počet T1 [ks]	1	0	2	1	0
Počet T2 [ks]	0	1	0	1	2
Odpad [cm]	70	60	20	10	0

(zdroj: [14])

Pro účely sestavení matematického modelu definujeme následující proměnné:

- x_1 : počet tyčí délky 150 cm rozřezaných 1. způsobem (v cm)
- x_2 : počet tyčí délky 150 cm rozřezaných 2. způsobem (v cm)
- x_3 : počet tyčí délky 180 cm rozřezaných 1. způsobem (v cm)
- x_4 : počet tyčí délky 180 cm rozřezaných 2. způsobem (v cm)
- x_5 : počet tyčí délky 180 cm rozřezaných 3. způsobem (v cm)

Cílem je minimalizace celkového řezného odpadu vznikajícího při rozřezávání skladových tyčí. Množství odpadu závisí na zvoleném způsobu řezu každé tyče (Tab. 41), a proto jej lze vyjádřit jako vážený součet dílčích odpadů. Účelovou funkci lze sestavit ve tvaru:

$$\min f(x) = 70x_1 + 60x_2 + 20x_3 + 10x_4$$

Omezující podmínky vycházejí z požadavků podniku na minimální množství požadovaných výrobků, dostupné skladové zásoby a omezení daná délkou jednotlivých tyčí:

Tyče o délce 150 cm jsou v počtu 200 kusů:

$$x_1 + x_2 \leq 200$$

Tyče o délce 180 cm jsou v počtu 240 kusů:

$$x_3 + x_4 + x_5 \leq 240$$

Počet potřebných tyčí T1 je aspoň 340 ks:

$$x_1 + 0x_2 + 2x_3 + x_4 + 0x_5 \geq 340$$

Počet potřebných tyčí T2 je aspoň 300 ks:

$$0x_1 + x_2 + 0x_3 + x_4 + 2x_5 \geq 300$$

Rozhodovací proměnná udává počet tyčí rozřezaných daným způsobem, nemůže nabývat záporných hodnot:

$$x_i \geq 0, \quad i = 1, 2, 3, 4, 5$$

Formulace slovní úlohy jako lineární model:

$$\min f(x) = 70x_1 + 60x_2 + 20x_3 + 10x_4 \quad (4.4)$$

za podmínek:

$$x_1 + x_2 \leq 200 \quad (4.4.1)$$

$$x_3 + x_4 + x_5 \leq 240 \quad (4.4.2)$$

$$x_1 + 2x_3 + x_4 \geq 340 \quad (4.4.3)$$

$$x_2 + x_4 + 2x_5 \geq 300 \quad (4.4.4)$$

$$x_i \geq 0$$

Tab. 42: Optimální řešení úlohy dělení materiálů

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	b
	0	0	0	0	0	1	2	1	1	40
	0	0,5	0	0,5	1	0	0	0	-0,5	150
	0	-0,5	1	0,5	0	0	1	0	0,5	90
	1	1	0	0	0	0	-2	-1	-1	160
f	0	0	0	0	0	0	-120	-70	-60	13 000

(zdroj: vlastní)

Optimální řešení:

$$x^{opt} = (160; 0; 90; 0; 150; 40; 0; 0; 0)$$

$$f^{opt} = 13\,000$$

Z výsledků vyplývá, že optimální rozřezání zahrnuje využití 160 kusů tyčí o délce 150 cm prvním způsobem řezu a u tyčí o délce 180 cm pak 90 kusů třetím a 40 kusů pátým způsobem řezu. Zvolená kombinace pokrývá požadavky na výrobu. Celkové množství vzniklého odpadu v tomto řešení činí 13 000 cm.

Na základě provedeného simplexového postupu bylo zjištěno, že existuje alternativní optimální řešení. Možnost alternativního optimálního řešení naznačuje poslední řádek simplexové tabulky Tab. 42, kde se pod proměnnými x_2 a x_4 , které nejsou součástí aktuální báze, nacházejí nulové hodnoty v řádku účelové funkce. To znamená, že jejich vstup do báze nezmění hodnotu cílové funkce a vede k dalšímu optimálnímu řešení.

Tab. 43 prezentuje příklady alternativních optimálních řešení, která všechna vedou ke stejné hodnotě účelové funkce $f = 1\,300$. Jedná se o různé kombinace hodnot strukturních proměnných, jež představují různé způsoby dosažení optimálního cíle za dodržení všech omezení modelu.

V tabulce jsou záměrně uvedeny pouze hodnoty strukturních proměnných, zatímco přídatné proměnné nejsou zobrazeny, protože jejich hodnota není pro účel této analýzy podstatná.

Tab. 43: Alternativní optimální řešení úlohy dělení materiálů

	x_1	x_2	x_3	x_4	x_5	b
	0	0	0	0	0	40
	0	0,5	0	0,5	1	150
	0	-0,5	1	0,5	0	90
	1	1	0	0	0	160
f	0	0	0	0	0	13 000
	0	0	0	0	0	40
	-0,5	0	0	0,5	1	70
	0,5	0	1	0,5	0	170
	1	1	0	0	0	160
f	0	0	0	0	0	13 000
	0	0	0	0	0	40
	-1	0	0	1	2	140

	1	0	1	0	-1	100
	1	1	0	0	0	160
<i>f</i>	0	0	0	0	0	13 000

(zdroj: vlastní)

Alternativní optimální řešení:

$$x^2 = (0; 160; 170; 0; 70);$$

$$x^3 = (0; 160; 100; 140; 0);$$

4.5 Dopravní problém

Ukázková úloha byla převzata ze zdroje [1].

Společnost Multicomp, s.r.o. má v ČR tři střediska (Plzeň, Pardubice, Olomouc), ve kterých montuje osobní počítače. Kapacita těchto středisek je 330, 150 a 220 ks počítačů měsíčně. Tyto počítače jsou distribuovány smluvním odběratelům v Brně, Praze, Ostravě a Liberci. Podle smluv dodá firma jednotlivým odběratelům postupně 180, 250, 160 a 110 ks počítačů. Distribuční náklady mezi středisky a odběrateli byly vykalkulovány na 1 ks počítače ve výši, která je zřejmá z Tab. 44 – uvedené hodnoty jsou ve stovkách Kč.

Tab. 44: Distribuční náklady mezi středisky a odběrateli (v stovkách Kč/ks)

	Brno	Praha	Ostrava	Liberec	Kapacity
Plzeň	11	4	17	9	330
Pardubice	6	7	10	8	150
Olomouc	3	9	5	12	220
Požadavky	180	250	160	110	Celkem = 700

(zdroj:[1])

Zápis dopravního modelu ve formě sumací poskytuje oproti předchozímu způsobu formulace větší přehlednost a stručnost: [1]

$$z = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij}$$

za podmínek:

$$\begin{aligned}\sum_{j=1}^n x_{ij} &= a_i, \quad i = 1, 2, \dots, m \\ \sum_{i=1}^m x_{ij} &= b_j, \quad j = 1, 2, \dots, n \\ x_{ij} &\geq 0, \quad i = 1, 2, \dots, m; j = 1, 2, \dots, n\end{aligned}$$

V zájmu přehlednosti matematického modelu zavedeme značení, v němž jednotlivá výrobní střediska označíme jako S_i , kde $i = 1, 2, 3$ a odběratele symbolem O_j , kde $j = 1, 2, 3, 4$. Každá proměnná x_{ij} pak bude reprezentovat počet kusů zboží, přepravovaných ze střediska S_i k odběrateli O_j .

Pro účely sestavení matematického modelu definujeme následující proměnné:

- x_{11} : Náklady ze střediska S_1 k odběrateli O_1 (v 100 Kč/ks)
- x_{12} : Náklady ze střediska S_1 k odběrateli O_2 (v 100 Kč/ks)
- x_{13} : Náklady ze střediska S_1 k odběrateli O_3 (v 100 Kč/ks)
- x_{14} : Náklady ze střediska S_1 k odběrateli O_4 (v 100 Kč/ks)
- x_{21} : Náklady ze střediska S_2 k odběrateli O_1 (v 100 Kč/ks)

Obdobně lze pokračovat i pro ostatní proměnné x_{ij} , kde index $i = 1, 2, 3$ označuje středisko a $j = 1, 2, 3, 4$ příslušného odběratele.

Matematický model této dopravní úlohy má za cíl minimalizovat celkové náklady na přepravu zboží ze středisek S_i ke konečným odběratelům O_j . Náklady přepravy mezi jednotlivými dvojicemi středisko–odběratel jsou uvedeny v Tab. 44. Celkové náklady lze vyjádřit jako součet dílčích nákladů přiřazených jednotlivým přepravním trasám. Účelovou funkci lze sestavit ve tvaru:

$$\begin{aligned}\min f(x) &= 11x_{11} + 4x_{12} + 17x_{13} + 9x_{14} + \\ &6x_{21} + 7x_{22} + 10x_{23} + 8x_{24} + \\ &3x_{31} + 9x_{32} + 5x_{33} + 12x_{34}\end{aligned}$$

V rámci formulace matematického modelu přepravní úlohy je nezbytné zohlednit kapacitní omezení jednotlivých středisek. Pro každé středisko S_i sledujeme celkové

množství zboží expedovaného k jednotlivým odběratelům O_j , přičemž toto množství nesmí překročit dostupnou kapacitu střediska.

Kapacitní omezení pro sklad S_i lze matematicky formulovat následovně:

$$x_{11} + x_{12} + x_{13} + x_{14} = 330$$

$$x_{21} + x_{22} + x_{23} + x_{24} = 150$$

$$x_{31} + x_{32} + x_{33} + x_{34} = 220$$

Je nutné zohlednit rovněž požadavky jednotlivých odběratelů. U každého odběratele O_j sledujeme součet všech dodávek ze středisek S_i , přičemž tento objem musí přesně odpovídat jeho specifikovanému požadavku.

Požadavky omezení pro prodejny P_j lze matematicky formulovat následovně:

$$x_{11} + x_{21} + x_{31} = 180$$

$$x_{12} + x_{22} + x_{32} = 250$$

$$x_{13} + x_{23} + x_{33} = 160$$

$$x_{14} + x_{24} + x_{34} = 110$$

Rozhodovací proměnná představující distribuci zboží, nemůže nabývat záporných hodnot:

$$x_{ij} \geq 0, \quad i = 1, 2, 3; j = 1, 2, 3, 4$$

Formulace slovní úlohy jako lineární model:

$$\begin{aligned} \min f(x) = & 11x_{11} + 4x_{12} + 17x_{13} + 9x_{14} + \\ & 6x_{21} + 7x_{22} + 10x_{23} + 8x_{24} + \\ & 3x_{31} + 9x_{32} + 5x_{33} + 12x_{34} \end{aligned}$$

za podmínek:

$$x_{11} + x_{12} + x_{13} + x_{14} = 330$$

$$x_{21} + x_{22} + x_{23} + x_{24} = 150$$

$$x_{31} + x_{32} + x_{33} + x_{34} = 220$$

$$x_{11} + x_{21} + x_{31} = 180$$

$$x_{12} + x_{22} + x_{32} = 250$$

$$x_{13} + x_{23} + x_{33} = 160$$

$$x_{14} + x_{24} + x_{34} = 110$$

$$x_{ij} \geq 0, \quad i = 1, 2, 3; j = 1, 2, 3, 4$$

Tab. 45: Optimální řešení úlohy dopravního problému

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	b
	0	1	0	0	0	1	0	0	0	1	0	0	250
	1	0	1	0	1	0	1	0	0	-1	0	-1	120
	0	0	1	0	0	0	1	0	0	0	1	0	160
	0	0	-1	0	0	0	-1	0	1	1	0	1	60
	1	0	1	1	0	-1	0	0	0	-1	0	0	80
	-1	0	-1	0	0	1	0	1	0	1	0	1	30
	0	0	0	0	0	0	0	0	0	0	0	0	0
f	-4	0	-8	0	0	-4	-2	0	0	-9	0	-7	3 660

(zdroj: vlastní)

Optimální řešení:

$$x^{opt} = (0; 250; 0; 80; 120; 0; 0; 30; 60; 0; 160; 0)$$

$$f^{opt} = 3\,660$$

Z výsledku vyplývá, že optimální plán distribuce počítačů zahrnuje dodávku z Plzně 250 ks do Prahy a 80 ks do Liberce, z Pardubic 120 ks do Brna a 30 ks do Liberce a z Olomouce 60 ks do Brna a 160 ks do Ostravy. Takto navržená distribuce plně pokrývá požadavky jednotlivých odběratelů a zároveň nepřekračuje kapacitní možnosti jednotlivých středisek. Celkové přepravní náklady v tomto případě činí 3 660, tedy 366 000 Kč, přičemž hodnoty v tabulce jsou uvedeny ve stovkách korun.

Tab. 46: Optimální rozložení dodávek mezi střediska a odběrateli

	Brno	Praha	Ostrava	Liberec	Kapacity
Plzeň	0	250	0	80	330
Pardubice	120	0	0	30	150
Olomouc	60	0	160	0	220
Požadavky	180	250	160	110	Celkem = 700

(zdroj: vlastní)

5 IMPLEMENTACE METODY V SOFTWARE

Tato kapitola se zaměřuje na softwarovou podporu řešení úloh LP. Nejprve je představen přehled dostupných nástrojů, které slouží k formulaci a řešení optimalizačních problémů. Následuje návrh a popis vlastní implementace vytvořené v programovacím jazyce Python, včetně uživatelského rozhraní, způsobu zadávání vstupních dat a zobrazení výstupních výsledků.

5.1 Přehled existujících softwarových nástrojů pro LP

V současnosti existuje široká škála softwarových nástrojů, které umožňují efektivní řešení úloh LP. Tyto nástroje se liší nejen svými výpočetními schopnostmi, ale také uživatelským rozhraním, dostupností a vhodností pro různé typy uživatelů. Mezi ně patří například GAMS, Microsoft Excel (s doplňkem Řešitel), AIMMS, MATLAB, WolframAlpha a spousta dalších. [7]

Pro účely této práce byly zvoleny nástroje **GAMS** a **Microsoft Excel**, a to především z důvodu jejich dostupnosti, ale také z důvodu osobního zájmu o porovnání různých přístupů k řešení lineárních úloh. Možnost využití jazyka MATLAB byla zvažována, avšak po konzultaci s vedoucím práce nebyla dále rozvíjena.

Pro účely porovnání softwarových nástrojů bude využit jednotný testovací příklad, který bude zadáván ve všech nástrojích ve stejné podobě. Díky tomu bude možné objektivně porovnat způsob zadávání vstupních dat, práci s modelem a způsob prezentace výstupních výsledků.

Testovací příklad:

$$\max f = 12x_1 + 25x_2 + 5x_3 \quad (5.1)$$

$$\text{za podmíněk: } x_1 + x_2 + x_3 \leq 10$$

$$4x_1 + 3x_2 + 2x_3 = 20$$

$$x_1 + x_2 + 10x_3 \geq 8$$

$$x_1, x_2 \geq 0 \quad x_3 \text{ URS}$$

5.1.1 GAMS

GAMS (General Algebraic Modeling System) je vysoce úrovnový modelovací systém pro matematické programování a optimalizaci. Systém se skládá z vlastního kompilátoru modelovacího jazyka a široké škály integrovaných řešících nástrojů. [20]

Program je dostupný ke stažení na oficiálních stránkách (www.gams.com), kde je možné si zvolit mezi bezplatnou demoverzí a plnou placenou verzí.

Proměnná x_3 nemůže být deklarována jako URS, a proto je vyjádřena jako rozdíl dvou nezáporných proměnných: $x_3 = x_{3poz} - x_{3neg}$. Proměnné jsou deklarovány v bloku *Variables* a nezápornost se nastavuje pomocí *Positive Variables*. Cílová funkce a omezení jsou zapsány pomocí rovnic, GAMS využívá specifické operátory:

=e= pro rovnost

=l= pro nerovnost $\leq$

=g= pro nerovnost $\geq$

Model je následně řešen jako lineární program pomocí klíčového slova *Solve* a výsledné hodnoty proměnných se zobrazují příkazem *Display*.

Ukázka modelu v jazyce GAMS:

```

1  * Deklarace rozhodovacích a pomocných proměnných používaných v modelu
2  * x1 a x2 jsou základní proměnné
3  * x3poz a x3neg proměnné pro URS proměnnou x3 (x3 = x3poz - x3neg)
4  * f je cílová proměnná
5  Variables
6      x1, x2,
7      x3poz, x3neg,
8      f;
9
10 * Podmínky všech proměnných
11 Positive Variables x1, x2, x3poz, x3neg;
12
13 * Deklarace rovnic (cílová funkce a omezení)
14 Equations
15     omezeni1
16     omezeni2
17     omezeni3
18     ucelovaf;
19
20 * =e= pro =, =l= pro ≤, =g= pro ≥
21 omezeni1.. x1 + x2 + x3poz - x3neg =l= 10;
22 omezeni2.. 4*x1 + 3*x2 + 2*x3poz - 2*x3neg =e= 20;
23 omezeni3.. x1 + x2 + 10*x3poz - 10*x3neg =g= 8;
24 ucelovaf.. f =e= 12*x1 + 25*x2 + 5*x3poz - 5*x3neg;
25
26 * Definice modelu (názvu) se všemi rovnicemi
27 Model LP_model /
28     omezeni1
29     omezeni2
30     omezeni3
31     ucelovaf/;
32
33 * Řešení modelu jako LP s cílem maximalizace f
34 Solve LP_model using lp maximizing f;
35
36 * Výstup hodnot proměnných
37 Display x1.l, x2.l, x3poz.l, x3neg.l, f.l;
38 |

```

Obrázek 9: Zadání modelu v GAMS

(zdroj: vlastní)

VARIABLE x1.L	=	0.000
VARIABLE x2.L	=	6.571
VARIABLE x3poz.L	=	0.143
VARIABLE x3neg.L	=	0.000
VARIABLE f.L	=	165.000

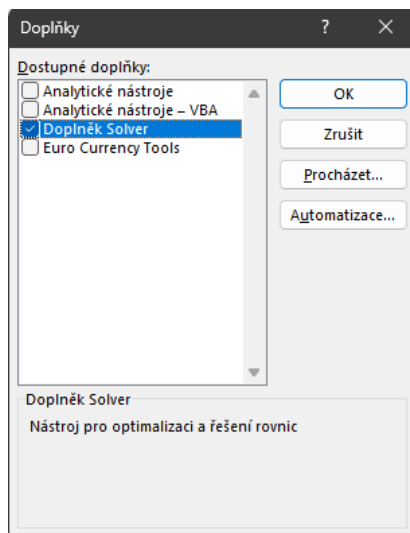
Obrázek 10: Výstup modelu v GAMS

(zdroj: vlastní)

5.1.2 MS Excel Řešitel

Microsoft Excel nabízí vestavěný nástroj **Řešitel** (*Solver*), který umožňuje řešit optimalizační úlohy, včetně úloh LP. Tento nástroj je běžně dostupný jako doplněk a je možné jej aktivovat v nastavení Excelu. Pro aktivaci doplňku je nutné jej nejprve zavést prostřednictvím nabídky Možnosti. Postup je následující:

1. V hlavním panelu Excelu klikněte na Soubor → Možnosti → Doplňky.
2. Ve spodní části okna nastavte Spravovat: Doplňky Excelu a klikněte na Přejít....
3. V zobrazeném dialogovém okně zaškrtněte Doplňek Solver a potvrďte výběr kliknutím na OK [21]



Obrázek 11: Dialogové okno doplňku v Excelu

(zdroj: vlastní)

Obrázek 12 znázorňuje list pro zadání LP modelu v prostředí Microsoft Excel. Horní část tabulky (řádky 1 až 7) slouží k vizuálnímu znázornění struktury modelu. Tato část je pouze orientační a neobsahuje žádné výpočetní vzorce.

	A	B	C	D	E	F	G
1	Proměnné:	x1	x2	x3+	x3-		
2							
3	Účelová funkce: max f=	12	25	5	-5		
4							
5	Za podmínek:	1	1	1	-1	<=	10
6		4	3	2	-2	=	20
7		1	1	10	-10	>=	8
8							
9	Výsledné proměnné:	x1	x2	x3+	x3-		
10							
11							
12	Hodnota účelové funkce:						
13	0						
14							
15	Kontrola správnosti:	Levá strana	Znak nerovnosti	Pravá strana			
16		0	<=	10			
17		0	=	20			
18		0	>=	8			

Obrázek 12: Přepis testovacího příkladu do listu Excel

(zdroj: vlastní)

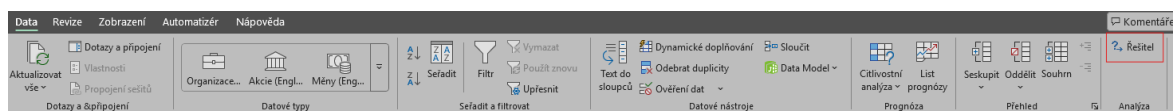
Pro zajištění funkčnosti výpočtů byly do jednotlivých buněk přidány vzorce sloužící k automatickému vyhodnocení hodnot účelové funkce i jednotlivých omezení:

- Buňky B10 až E10 po spuštění Řešitele automaticky uloží hodnoty optimalizovaných rozhodovacích proměnných.
- Buňka A13 obsahuje vzorec pro výpočet hodnoty účelové funkce:

$$=B3*B10 + C3*C10 + D3*D10 + E3*E10$$
- Buňky B16, B17, B18 vyhodnocují levé strany jednotlivých omezení pomocí vzorců:

$$=B5*B9 + C5*C9 + D5*D9 + E5*E9, =B6*B9 + C6*C9 + D6*D9 + E6*E9,$$

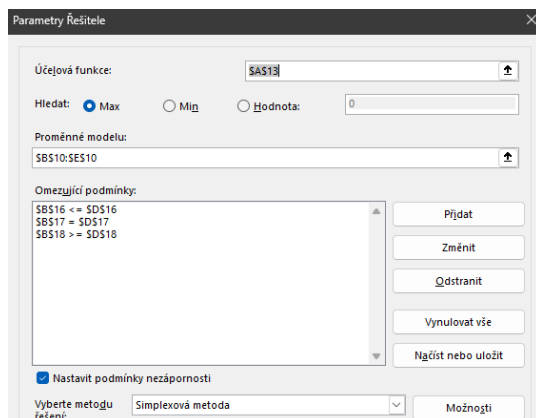
$$=B7*B9 + C7*C9 + D7*D9 + E7*E9$$
- Buňky D15, D16, D17 obsahují pouze pravé strany omezení.



Obrázek 13: Spuštění doplňku Řešitel

(zdroj: vlastní)

Po kliknutí na možnost **Řešitel** se zobrazí dialogové okno, ve kterém je možné zadat cílovou buňku (účelovou funkci), určit typ optimalizační úlohy (maximalizace nebo minimalizace), nastavit podmínky a metodu řešení. Na Obrázek 14 je znázorněno konkrétní vyplnění parametrů pro řešení testovacího příkladu.



Obrázek 14: Parametry Řešitele

(zdroj: vlastní)

Výsledné hodnoty optimalizovaných proměnných se automaticky zobrazily ve vybraných buňkách (B10:E10), které byly v okně Řešitele definovány jako rozhodovací proměnné. Hodnota účelové funkce je následně dopočítána ve vyhrazené buňce (A12) pomocí vzorce.

Proměnná x_3 , která je bez znaménkového omezení, musela být rozdělena na dvě nezáporné složky x_3^+ a x_3^- .

Ve spodní části tabulky je vytvořena kontrolní sekce, která ověřuje splnění všech zadaných omezení. Hodnoty na levé straně odpovídají pravé straně nerovností a rovnic, což potvrzuje, že nalezené řešení je platné a model byl správně nastaven.

	A	B	C	D	E	F	G
1	Proměnné:	x1	x2	x3+	x3-		
2							
3	Účelová funkce: max f=	12	25	5	-5		
4							
5	Za podmínek:	1	1	1	-1	<=	10
6		4	3	2	-2	=	20
7		1	1	10	-10	>=	8
8							
9	Výsledné proměnné:	x1	x2	x3+	x3-		
10		0	6,571428571	0,142857143	0		
11							
12	Hodnota účelové funkce:						
13	165						
14							
15	Kontrola správnosti:	Levá strana	Znak nerovnosti	Pravá strana			
16		6,714285714	<=	10			
17		20	=	20			
18		8	>=	8			

Obrázek 15: Výstup modelu v Excel Řešiteli

(zdroj: vlastní)

5.2 Implementace vlastního softwarového řešení

V rámci této kapitoly je představeno jednoduché softwarové řešení vytvořené pro účely procvičování metod LP. Výsledná aplikace má podobu interaktivní kalkulačky, která kromě zadávání vstupních dat umožňuje také zobrazení jednotlivých kroků výpočtu pomocí vestavěného náhledu na simplexovou tabulku.

Aplikace byla implementována v programovacím jazyce Python (verze alespoň 3.8) s využitím knihovny **PyQt5** pro tvorbu grafického uživatelského rozhraní. Pro vizualizaci byla použita knihovna **matplotlib**, konkrétně její modul **pyplot**, který umožňuje jednoduché vykreslování grafických výstupů ve 2D prostoru.

Uvedeny budou pouze vybrané úseky zdrojového kódu, které demonstrují klíčové funkce a **principy fungování simplexové metody**. Vzhledem k délce některých funkcí jsou v textu práce prezentovány pouze nejdůležitější části. Kompletní zdrojový kód je uveden v příloze práce.

5.2.1 Programovací jazyk Python

Python je interpretovaný, objektově orientovaný a vysoce úroňový programovací jazyk s dynamickým typovým systémem. Díky podpoře modulů a balíčků je vhodný pro tvorbu přehledně strukturovaných a znovupoužitelných aplikací. Python je široce využíván jak pro rychlý vývoj aplikací, tak jako skriptovací jazyk pro propojování různých komponent. [22]

Je volně dostupný a lze jej stáhnout²⁸ ve formě instalačních balíčků na oficiálních stránkách (www.python.org).

5.2.2 Výchozí tvar simplexové tabulky

Při sestavování výchozí simplexové tabulky byly implementovány tři klíčové funkce:

- Funkce *expand_variables* zajišťuje podmínku bez znaménkového omezení na dvojici složek.
- Funkce *initialize_simplex_table* následně vytváří výchozí simplexovou tabulku, pomocí přidání přídatných a umělých proměnných podle typu omezení ($\leq$, $\geq$, $=$). Každému omezení přiřazuje 1 nebo -1 tam, kde má být, a připravuje i řádek pomocné účelové funkce.

²⁸ Při instalaci je nutné zaškrtnout „Add python.exe to Path“ pro instalování balíčku v příkazovém řádku.

- Funkce *adjust_auxiliary_objective_function* provádí úpravu pomocné účelové funkce tak, že sčítá příslušné řádky tabulky. Výsledkem je anulovaný tvar pomocné účelové funkce.

Kód 1: Rozšíření URS proměnné

```

1. def expand_variables(values, conditions):
2.     """ Rozšíření proměnných pro URS """
3.     expanded = []
4.     for i, val in enumerate(values):
5.         if conditions[i] == "URS":
6.             expanded.extend([val, -val])
7.         else:
8.             expanded.append(val)
9.     return expanded

```

Kód 2: Přidávání přídatných a pomocných proměnných

```

1. def initialize_simplex_table(obj_function, constraints, conditions):
2.     """ Inicializace simplexové tabulky """
3.     # [předchozí část kódu vynechána]
4.     for lhs, operator, rhs in [(c[:-2], c[-2], c[-1]) for c in init_constraints]:
5.         row = lhs + [0] * (num_slack + num_artificial) + [rhs]
6.
7.         if operator == "<=":
8.             row[slack_idx] = 1
9.             slack_idx += 1
10.
11.        elif operator == ">=":
12.            row[slack_idx] = -1
13.            row[artificial_idx] = 1
14.            auxiliary_row[artificial_idx] = -1
15.            slack_idx += 1
16.            artificial_idx += 1
17.
18.        elif operator == "=":
19.            row[artificial_idx] = 1
20.            auxiliary_row[artificial_idx] = -1
21.            artificial_idx += 1
22.
23.        init_simplex_table.append(row)

```

Kód 3: Anulovaný tvar pomocné účelové funkce

```

1. def adjust_auxiliary_objective_function(init_simplex_table):
2.     """ Sečtení příslušných řádků s pomocnou účelovou funkcí """
3.
4.     adjusted_table = [row[:] for row in init_simplex_table]
5.     auxiliary_row = adjusted_table[-1]
6.
7.     # Sečtení -1 a 1 v příslušných sloupcích
8.     for i in range(len(auxiliary_row)):
9.         if auxiliary_row[i] == -1:
10.            for row in adjusted_table[:-1]:
11.                if row[i] == 1:
12.                    for j in range(len(auxiliary_row)):
13.                        auxiliary_row[j] += row[j]
14.
15.     return adjusted_table
16.

```

5.2.3 Nalezení klíčového prvku

Metody *find_pivot_column_maximize* a *find_pivot_column_minimize* slouží k určení klíčového sloupce v závislosti na typu optimalizační úlohy. Obě využívají téměř shodnou logiku a liší se pouze použitím funkce *min()* a *max()*. V obou případech dochází k prohledávání posledního řádku simplexové tabulky, kde se identifikuje příslušný sloupec pro další iteraci.

Naopak metoda *find_pivot_row* provádí poměrový test, pomocí kterého se určuje klíčový řádek. Poměr mezi hodnotou pravé strany a hodnotou v klíčovém sloupci se vypočítá pro každý řádek a uloží se do seznamu.

Kód 4: Nalezení pivotního prvku

```

1. def find_pivot_column_maximize(table, obj_function, constraints, conditions):
2.     """ Nalezení pivotu ve sloupci pro maximalizaci """
3.
4.     simplex_table, num_artificial = initialize_simplex_table(obj_function, constraints,
5. conditions)
6.     last_row = table[-1][: -1]
7.     min_value = min(last_row[:len(simplex_table[-1]) - num_artificial - 1])
8.     return last_row.index(min_value) if min_value < 0 else -1
9.
10. def find_pivot_column_minimize(table, obj_function, constraints, conditions):
11.     """ Nalezení pivotu ve sloupci pro minimalizaci """
12.
13.     simplex_table, num_artificial = initialize_simplex_table(obj_function, constraints,
14. conditions)
15.     last_row = table[-1][: -1]
16.     max_value = max(last_row[:len(simplex_table[-1]) - num_artificial - 1])
17.     return last_row.index(max_value) if max_value > 0 else -1
18.
19. def find_pivot_row(table, pivot_column):
20.     """ Nalezení pivotového řádku """
21.
22.     pivot_row_ratios = []
23.
24.     for row in table[: -1]:
25.         # Jen pokud pivotní hodnota > 0 a zároveň pravá strana >= 0
26.         if row[pivot_column] > 0:
27.             ratio = row[-1] / row[pivot_column]
28.             # Zajistíme, že 0 nebude vybráno jako pivot
29.             if ratio > 0:
30.                 pivot_row_ratios.append(ratio)
31.             else:
32.                 pivot_row_ratios.append(float('inf'))
33.         else:
34.             pivot_row_ratios.append(float('inf'))
35.
36.     return pivot_row_ratios
37.

```

5.2.4 Nalezení klíčového prvku

Funkce *gauss_pivoting_simplex* aktualizuje simplexovou tabulku po výběru klíčového prvku. Nejprve se klíčový řádek vydělí klíčovou hodnotou a následně jsou ostatní řádky upraveny pomocí Gaussovy eliminace tak, aby ve sloupci pivotu vznikly nuly. Každý řádek je převeden podle vzorce.

Kód 5: Gaussova eliminační metoda

```

1. def gauss_pivoting_simplex(table, pivot_row, pivot_column):
2.     """ Aktualizace simplexové tabulky """
3.
4.     # Vydělení pivotního řádku pivotní hodnotou
5.     pivot = table[pivot_row][pivot_column]
6.     table[pivot_row] = [pivot_row_value / pivot for pivot_row_value in table[pivot_row]]
7.
8.     # Gaussova eliminace (Nový řádek = starý řádek - kl.faktor * pivotní řádek)
9.     for i, row in enumerate(table):
10.         if i != pivot_row:
11.             factor = row[pivot_column]
12.             table[i] = [row[j] - factor * table[pivot_row][j] for j in range(len(row))]
13.
14.     return table

```

5.2.5 Přepoččet simplexové tabulky

Průběh simplexové metody je realizován pomocí funkcí *simplex_iteration_maximize* a *simplex_iteration_minimize*, které zajišťují iterační řešení úloh. Obě funkce využívají dříve definované funkce a jsou řízeny pomocí cyklu *while* s tzv. *walrus operátorem* (*:=*), jenž umožňuje současné přiřazení i kontrolu hodnoty v podmínce. Iterace končí ve chvíli, kdy jsou splněna kritéria optimality.

Kód 6: Průběh simplexové metody pro maximalizaci²⁹

```

1. def simplex_iteration_maximize(table, obj_function, constraints, conditions):
2.     """ Iterace simplexové metody pro maximalizaci """
3.
4.     # Opakování, dokud existuje záporná hodnota v posledním řádku
5.     while (pivot_column := find_pivot_column_maximize(table, obj_function, constraints,
6. conditions)) != -1:
7.         ratios = find_pivot_row(table, pivot_column)
8.         # Kontrola nekonečných hodnot
9.         if all(r == float('inf') for r in ratios):
10.             return simplex_table, None
11.
12.         pivot_row = ratios.index(min(ratios))
13.         table = gauss_pivoting_simplex(table, pivot_row, pivot_column)
14.         simplex_table += format_simplex_table(table, constraints, conditions) + "\n"
15.         last_iteration = table
16.
17.     return simplex_table, last_iteration

```

²⁹ Pro minimalizaci obsahuje podmínku *if all(value < 1e-10 for value in table[-1]): table.pop()*, k odstranění pomocné účelové funkce

5.2.6 Jednofázová a dvoufázová metoda

Rozhodnutí se opírá ve funkci *simplex_method* o kontrolu hodnot posledního řádku výchozí simplexové tabulky, zda obsahuje pouze nuly. Pokud ano, lze přistoupit rovnou k optimalizaci. Pokud ne, je nutné nejprve upravit pomocnou účelovou funkci a zahájit první fázi pomocí dvoufázové metody.

Kód 7: Spuštění metody

```
1. def simplex_method(optimization_type, obj_function, constraints, conditions):
2.     """ Spuštění simplexové metody """
3.     # [předchozí část kódu vynechána]
4.     single_phase = False
5.     if all(value == 0 for value in init_table[-1]):
6.         init_table.pop()
7.         single_phase = True
8.
9.     output += format_simplex_table(init_table, constraints, conditions) + "\n"
10.
11.    # Jednofázová metoda
12.    if single_phase:
13.        output += format_simplex_table(init_table, constraints, conditions) + "\n"
14.        args = (init_table, obj_function, constraints, conditions)
15.        if optimization_type == "max f":
16.            simplex_table, last_iteration = simplex_iteration_maximize(*args)
17.        else:
18.            simplex_table, last_iteration = simplex_iteration_minimize(*args)
19.
20.    # Dvoufázová metoda
21.    else:
22.        adjusted_table = adjust_auxiliary_objective_function(init_table)
23.        output += format_simplex_table(adjusted_table, constraints, conditions) + "\n"
24.        args = (adjusted_table, obj_function, constraints, conditions)
25.        simplex_table, last_iteration = simplex_iteration_minimize(*args)
26.        output += simplex_table
27.
28.        if optimization_type == "max f":
29.            simplex_table, last_iteration = simplex_iteration_maximize(*args)
30.        else:
31.            simplex_table, last_iteration = simplex_iteration_minimize(*args)
```

5.2.7 Pomocné funkce

V rámci implementace byly rovněž vytvořeny pomocné funkce, které neslouží k samotnému řešení úlohy, ale zajišťují formátování výstupu, práci s názvy proměnných nebo předzpracování vstupu. Tyto funkce nejsou součástí výpočetního jádra a mají spíše podpůrnou či estetickou roli. Pomocné funkce jsou:

- *format_fraction* – převod desetinných čísel na zlomkový zápis pro lepší čitelnost výstupu
- *invert_inequalities* – úprava znamének v omezeních tak, aby pravé strany obsahovaly nezáporné hodnoty
- *generate_table_header* – sestavení hlavičky simplexové tabulky na základě typu proměnných a omezení,
- *format_simplex_table* – převod číselné tabulky do přehledné textové reprezentace.

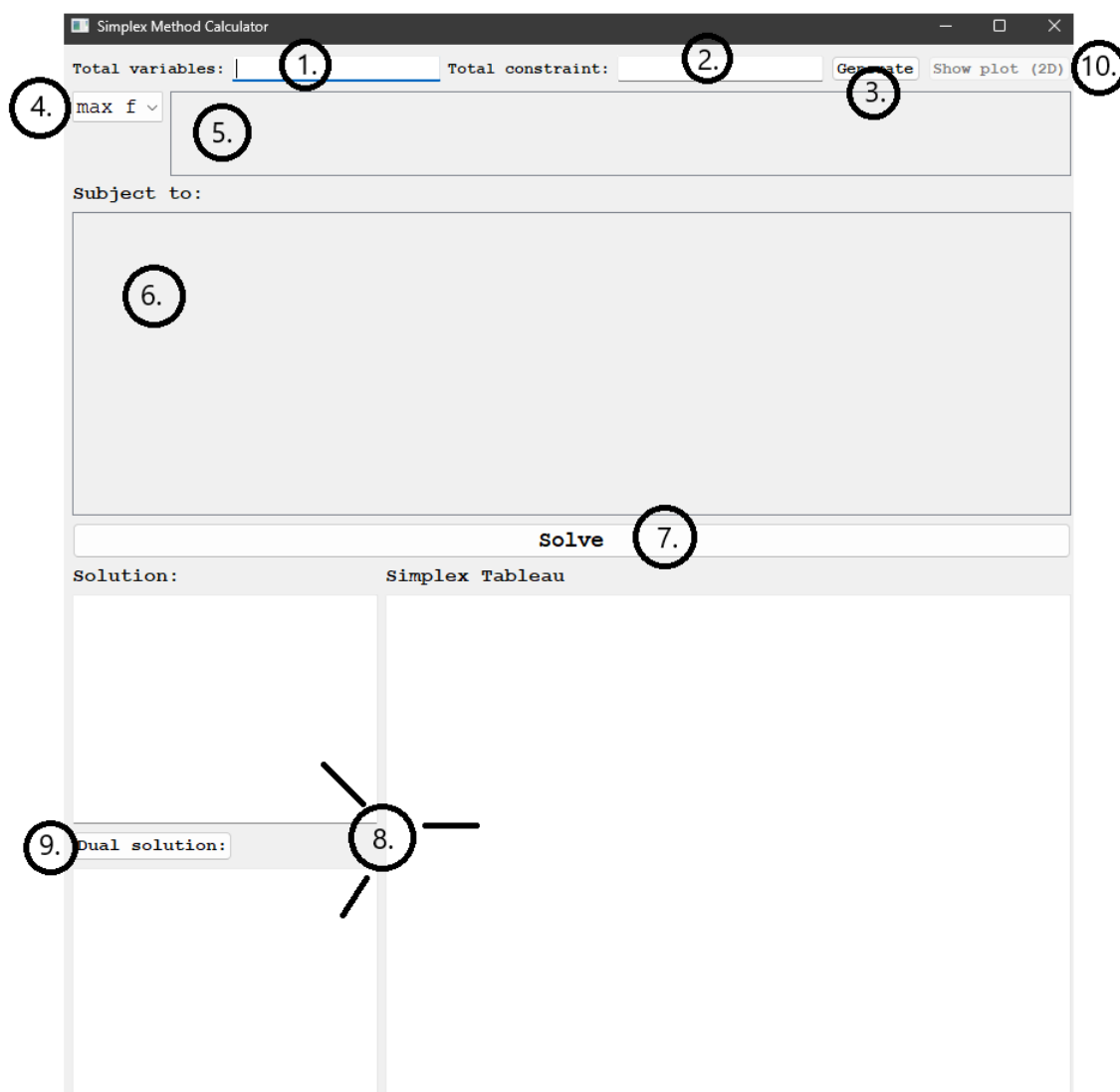
5.2.8 Grafické rozhraní

Pro vytvoření grafického uživatelského rozhraní (dále jako GUI) byla využita knihovna **PyQt5**, která poskytuje vazbu mezi programovacím jazykem **Python** a frameworkem **Qt**. Díky své modulární architektuře a nástrojům, jako je Qt Designer, umožňuje PyQt5 efektivní návrh a implementaci uživatelského rozhraní. [23]

Pro použití knihovny PyQt5 je potřeba mít nainstalovaný Python ve verzi 3.8 nebo vyšší. Pokud již Python nainstalovaný máte, můžete knihovnu PyQt5 nainstalovat pomocí nástroje *pip* v příkazovém řádku:

```
pip install PyQt5
```

```
pip install matplotlib
```



Obrázek 16: Vzhled GUI programu

(zdroj: vlastní)

GUI tvoří deset základních prvků, které jsou znázorněny na Obrázek 16:

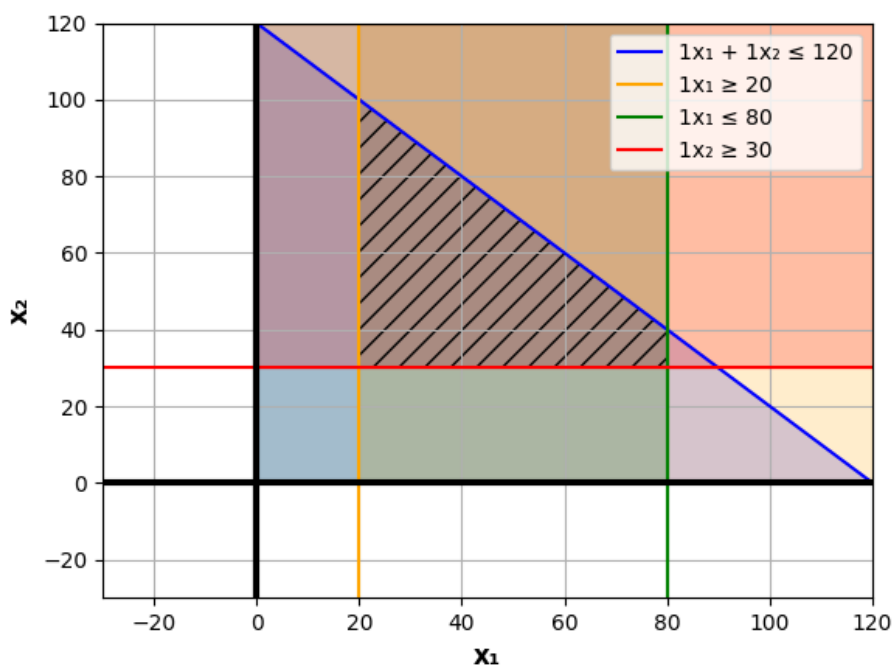
1. **Pole pro zadání počtu rozhodovacích proměnných** – na základě této hodnoty se dynamicky vygeneruje odpovídající počet vstupních polí. I když není počet technicky omezen, z důvodu výpočetní náročnosti se nedoporučuje zadávat příliš vysoké hodnoty;
2. **Pole pro zadání počtu omezení** – podobně jako u proměnných se dynamicky vytvoří příslušný počet řádků pro zadání podmínek;
3. **Tlačítko „Generate“** – slouží ke spuštění generování vstupních polí pro zadání úlohy;

4. **Volba typu úlohy** – uživatel zde určuje, zda se jedná o maximalizační nebo minimalizační úlohu;
5. **Okno pro zadání cílové funkce** – zobrazuje a umožňuje editaci koeficientů rozhodovacích proměnných;
6. **Okno pro zadání omezujících podmínek** – slouží k zadání koeficientů jednotlivých podmínek, typu nerovnosti ($\leq$, $=$, $\geq$), pravé strany a zároveň umožňuje určit podmínky pro každou proměnnou;
7. **Tlačítko „Solve“** – spustí výpočet pomocí simplexové metody;
8. **Výstupní pole** – zobrazí výsledky výpočtu, konkrétně hodnoty rozhodovacích proměnných primární úlohy, řešení odpovídající duální úlohy a jednotlivé iterace simplexové tabulky;
9. **Tlačítko „Dual solution“** – provede převod úlohy na její duální formu a zobrazí odpovídající zápis;
10. **Tlačítko „Show plot (2D)“** – vizualizuje grafické řešení úlohy, je-li zadán přesně dvourozměrný model (tedy právě dvě proměnné);

5.2.9 Grafické řešení v GUI

Pro vizualizaci grafického řešení úloh se dvěma proměnnými byla použita knihovna **Matplotlib**, konkrétně její modul **pyplot**, který umožňuje jednoduché vykreslování grafů. Tento modul nabízí širokou škálu funkcí pro práci s grafy, včetně vykreslení nerovností, průsečíků a hranic řešení, což je vhodné pro přehledné znázornění přípustné oblasti i optimálního bodu.

Pro ilustraci grafického řešení byla použita úloha alokace zdrojů (4.). Tento příklad byl zvolen záměrně, neboť obsahuje právě dvě strukturní proměnné, což umožňuje vizualizaci řešení v rovinném prostoru. Díky tomu lze přehledně znázornit oblast přípustných řešení, jednotlivá omezení i optimální řešení úlohy. Pro zobrazení grafického výstupu je nutné stisknout tlačítko **Show plot (2D)** (10.)



Obrázek 17: Grafické řešení v GUI

(zdroj: vlastní)

5.2.10 Zadávání a výstup v GUI

Na Obrázek 18 je vidět, že ke každému vytvořenému poli pro zadání hodnot byla přiřazena odpovídající proměnná x_n , a každé omezení bylo pro lepší přehlednost očíslováno podle své pozice v zadání.

Obrázek 18: Zadávání testovacího příkladu v GUI

(zdroj: vlastní)

Výstup řešení zadané úlohy, který zahrnuje výsledky primární úlohy, průběh simplexové metody a odpovídající duální řešení, jsou znázorněny na Obrázek 19.

Solution:

x1 = 0

x2 = 6.57

x3 = 0.14

x4 = 3.29

x5 = 0

x6 = 0

f = 165

Dual solution:

min g(y) = 10y1 + 20y2 + -8y3

y1 = 0, y2 = 8.75, y3 = 1.25

min g = 165

1y1 + 4y2 + -1y3 ≥ 12

1y1 + 3y2 + -1y3 ≥ 25

1y1 + 2y2 + -10y3 = 5

y1 ≥ 0, y2 URS, y3 ≥ 0

Simplex Tableau

-23/2

-49/2

0

0

0

-1/2

0

1/2

4

19/5

14/5

0

0

0

1/5

0

-6/5

92/5

x1

x2

x3'

x3''

x4

x5

y1

y2

b

0

9/38

0

0

1

1/19

-9/38

-1/19

92/19

1

14/19

0

0

0

1/19

5/19

-1/19

92/19

0

1/38

1

-1

0

-2/19

-1/38

2/19

6/19

0

-609/38

0

0

0

2/19

115/38

-2/19

1134/19

x1

x2

x3'

x3''

x4

x5

y1

y2

b

0

9/2

0

0

19

1

-9/2

-1

92

1

1/2

0

0

-1

0

1/2

0

0

0

1/2

1

-1

2

0

-1/2

0

10

0

-33/2

0

0

-2

0

7/2

0

50

x1

x2

x3'

x3''

x4

x5

y1

y2

b

-9

0

0

0

28

1

-9

-1

92

2

1

0

0

-2

0

1

0

0

-1

0

1

-1

3

0

-1

0

10

33

0

0

0

-35

0

20

0

50

x1

x2

x3'

x3''

x4

x5

y1

y2

b

-9/28

0

0

0

1

1/28

-9/28

-1/28

23/7

19/14

1

0

0

0

1/14

5/14

-1/14

46/7

-1/28

0

1

-1

0

-3/28

-1/28

3/28

1/7

87/4

0

0

0

0

5/4

35/4

-5/4

165

x1

x2

x3'

x3''

x4

x5

y1

y2

b

Obrázek 19: Výstup testovacího příkladu v GUI

(zdroj: vlastní)

Z Obrázek 10, Obrázek 15 a Obrázek 19, které reprezentují výstup jednotlivých programu, je patrné, že výsledné hodnoty strukturálních proměnných i účelové funkce jsou ve všech případech shodné až na drobné rozdíly v počtu zobrazovaných desetinných míst. Po zaokrouhlení lze tedy optimální řešení považovat za totožné:

$$x^{opt} = (0; 6,57; 0,14)$$

$$f^{opt} = 165$$

5.2.11 Dualita v GUI

Po stisknutí tlačítka **Dual solution** (9.) se automaticky přepíší hodnoty v polích pro zadání počtu proměnných (1.), počtu omezení (2.), polí pro zadávání proměnných (5.) i omezujících podmínek (6.).³⁰, aby odpovídaly duální úloze. Tento převod je znázorněn na Obrázek 20.

³⁰ Totéž platí i pro případy, kdy je proměnná bez omezení znaménkem (URS).

Simplex Method Solver

Total variables: 3 Total constraint: 3 Generate Show plot (2D)

min f

10 x1 20 x2 -8 x3

Subject to:

1 x1 4 x2 -1 x3 ≥ 12 (1)

1 x1 3 x2 -1 x3 ≥ 25 (2)

1 x1 2 x2 -10 x3 = 5 (3)

x1 ≥ 0 x2 URS x3 ≥ 0

Solve

Obrázek 20: Převod úlohy na její duální formu v GUI

(zdroj: vlastní)

5.2.12 Chybové hlášení

Závěrem k výstupnímu zobrazení aplikace je třeba uvést, že obsahuje také základní kontrolní podmínky vstupu, které upozorňují uživatele na případné chyby ve formulaci úlohy. Tyto výstupy slouží k lepší orientaci uživatele a eliminaci chyb při práci s aplikací.

V případě nesprávně zadané účelové funkce:

- **The objective function is not written correctly.**

Při chybně zadaných omezeních:

- **The constraints are not written correctly.**

V případě, že optimalizační úloha nemá omezené řešení:

- **Unbounded solution – no valid pivot row.**

Pokud není možné duální úlohu vytvořit, například z důvodu nesplnění strukturálních předpokladů, aplikace informuje hlášením:

- **Dual problem is not available.**

ZÁVĚR

Tato práce se zabývala lineárním programováním, konkrétně simplexovou metodou, a jejím praktickým využitím při řešení optimalizačních úloh. Přestože lineární programování představuje pouze dílčí oblast širšího rámce operačního výzkumu, jedná se o velmi důležitý a široce využívaný nástroj v mnoha oblastech, od průmyslu až po ekonomii.

V teoretické části byla podrobně rozebrána struktura úloh lineárního programování, jejich klasifikace a základní pojmy, které jsou klíčové pro pochopení fungování optimalizačních algoritmů. Během zpracování této části jsem narazil na několik témat, která nebyla v běžně dostupné literatuře dostatečně rozvedena. Hlubším rešeršním úsilím se mi podařilo nalézt doplňující odborné zdroje, ze kterých jsem mohl čerpat a tím práci vhodně doplnit o chybějící souvislosti.

Praktická část ukázala konkrétní příklady řešení různých typů úloh. Dále byly představeny vybrané softwarové nástroje pro řešení lineárních úloh a detailně popsána implementace vlastního řešení v programovacím jazyce Python, včetně grafického rozhraní, které umožňuje intuitivní zadávání a vizualizaci řešení simplexové metody.

Zajímavostí této práce je, že jsem začal nejprve s implementací softwarového řešení, kde jsem využil především znalosti získané v předmětu optimalizační metody. Teoretická část mi pak pomohla lépe porozumět jednotlivým principům simplexové metody, duality i možnostem zakončení úloh, a zároveň rozšířila mé znalosti o pojmy a souvislosti, které mi dříve nebyly známy.

Věřím, že vytvořený softwarový nástroj může posloužit jako praktická pomůcka pro studenty i zájemce o tuto oblast a zároveň jako pevný základ pro další rozvoj.

Závěrem lze shrnout, že tato práce naplnila svůj cíl propojit teoretické poznatky s praktickou aplikací simplexové metody. Proces tvorby softwarového nástroje mi umožnil nejen lépe pochopit principy lineárního programování, ale také si ověřit jejich fungování na reálných úlohách. Díky tomu jsem získal hlubší vhled do tématu a rozšířil své dovednosti v oblasti optimalizačních metod i programování.

SEZNAM POUŽITÉ LITERATURY

- [1] JABLONSKÝ, Josef. *Operační výzkum: kvantitativní modely pro ekonomické rozhodování*. 3. vyd. Praha: Professional Publishing, 2007. ISBN 978-80-86946-44-3.
- [2] PROKOP, Roman. *Lineární programování*. Zlín: FAI Univerzita Tomáše Bati.
- [3] ENCYCLOPAEDIA BRITANNICA, INC. *Linear programming*. Online. ENCYCLOPAEDIA BRITANNICA, INC. Encyclopaedia Britannica. 2025. Dostupné z: <https://www.britannica.com/science/linear-programming-mathematics>. [cit. 2025-04-17].
- [4] STANFORD UNIVERSITY. *George B. Dantzig, operations research professor, dies at 90*. Online. STANFORD UNIVERSITY. Stanford News. 2005. Dostupné z: <https://news.stanford.edu/stories/2005/05/george-b-dantzig-operations-research-professor-dies-90>. [cit. 2025-04-17].
- [5] PROKOP, Roman. *Metody optimalizace*. Výukové materiály. Zlín: FAI Univerzita Tomáše Bati, [n.d].
- [6] PEKAŘ, Libor. *Optimalizace*. Výukové materiály. Zlín: FAI Univerzita Tomáše Bati, 2013.
- [7] HRABEC, Dušan. *Matematika v bezpečnostních technologiích*. Zlín: FAI Univerzita Tomáše Bati, [n.d].
- [8] KREISEL, Tomáš. *Lineární programování a jeho využití ve vybraných úlohách*. Online, Bakalářská práce, vedoucí Mgr. Jana Řezníčková, Ph.D. Zlín: FAI Univerzita Tomáše Bati, 2019. Dostupné z: https://digilib.k.utb.cz/bitstream/handle/10563/44568/kreisel_2019_dp.pdf. [cit. 2025-04-20].
- [9] FAREED, Hiba G. *Operation Research*. Online, Výukové materiály. [n.d], 2022. Dostupné z: https://uomustansiriyah.edu.iq/media/lectures/6/6_2022_01_08!08_06_42_P_M.pdf. [cit. 2025-05-10].
- [10] ATTARD, Natalie a SKLENAR, Jaroslav. *Linear Programming*. Online, Výukové materiály. [n.d]. Dostupné z: https://is.muni.cz/www/jkrejci/Linear_Programming.pdf. [cit. 2025-05-22].

- [11] UNIVERSITY OF CALIFORNIA SAN DIEGO. *Linear Programming Notes V: Problem Transformations*. Online, Výukové materiály. [n.d]. [cit. 2025-04-28]. Dostupné z: <https://econweb.ucsd.edu/~jsobel/172aw02/notes5.pdf>
- [12] LUENBERGER, David G a YE, Yinyu. *Linear and nonlinear programming*. Third Edition. New York: Springer, 2014. ISBN 978-0387-74502-2.
- [13] DEMEL, Jiří. *Operační výzkum*. Online, Výukové materiály. Praha: Fakulta stavební ČVUT v Praze, 2022. Dostupné z: <https://kix.fsv.cvut.cz/~demel/ped/ov/ov.pdf>. [cit. 2025-05-08].
- [14] LINDA, Bohdan a Josef VOLEK. *Lineární programování*. 6. Vydání., opravené a doplněné. Pardubice: Univerzita Pardubice, 2016. ISBN 978-80-7560-018-9.
- [15] PLESNÍK, Ján; DUPAČOVÁ, Jitka a VLACH, Milan. *Lineárne programovanie*. Bratislava: Alfa, 1990. ISBN 80-05-00679-9.
- [16] AWAD, Feraas. *Linear Programming Lecture Notes for Math 373*. Online, Výukové materiály. Philadelphia: Philadelphia University, [n.d] Dostupné z: <https://www.philadelphia.edu.jo/academics/fawad/uploads/lp-fawad-0323.pdf>. [cit. 2025-04-15].
- [17] FIŠNAROVÁ, Simona. *Základy lineárního programování*. Online, Výukové materiály. Brno: Ústav matematiky Mendelovy univerzity v Brně, 2012. Dostupné z: <https://user.mendelu.cz/fisnarov/imt/prednasky/lp.pdf>. [cit. 2025-04-15]
- [18] JACKSON, BJ. *Linear Programming*. Online, Výukové materiály. University of Georgia, 2002. Dostupné z: <https://jwilson.coe.uga.edu/EMT668/EMAT6680.2002/Jackson/EMAT%206000/topic%205/topic%205-lin%20program.html>. [cit. 2025-04-15].
- [19] KUBIŠOVA, Andrea. *Operační výzkum*. Online, Výukové materiály. Jihlava: Vysoká škola polytechnická Jihlava, Katedra matematiky, 2014. Dostupné z: https://is.muni.cz/th/dxuqz/P1_skripta.pdf. [cit. 2025-04-15].
- [20] GAMS DEVELOPMENT CORP. *GAMS Language*. Online. GAMS.com. [n.d]. Dostupné z: <https://www.gams.com/products/gams/gams-language/>. [cit. 2025-04-15].

- [21] MICROSOFT. *Load the Solver Add-in in Excel*. Online. Microsoft Support. [n.d]. Dostupné z: <https://support.microsoft.com/en-us/office/load-the-solver-add-in-in-excel-612926fc-d53b-46b4-872c-e24772f078ca>. [cit. 2025-04-15].
- [22] PYTHON SOFTWARE FOUNDATION. *What is Python? Executive Summary*. Online. Python.org. [n.d]. Dostupné z: <https://www.python.org/doc/essays/blurb/>. [cit. 2025-04-15].
- [23] *PyQt5*. Online. PYTHON SOFTWARE FOUNDATION. PyPI. [n.d]. Dostupné z: <https://pypi.org/project/PyQt5/>. [cit. 2025-04-15].

SEZNAM OBRÁZKŮ

Obrázek 1: Množina přípustných řešení příkladu (1.6)	19
Obrázek 2: Možnosti zakončení výpočtu.....	20
Obrázek 3: Hrubé schéma simplexové metody	24
Obrázek 4: Grafické znázornění simplexové metody	30
Obrázek 5: Grafické znázornění příkladu (2.6)	37
Obrázek 6: Ilustrace transformace mezi primární a duální úlohy	39
Obrázek 7: Alternativní optimální řešení.....	49
Obrázek 8: Grafické znázornění úlohy alokaci zdrojů.....	71
Obrázek 9: Zadání modelu v GAMS	87
Obrázek 10: Výstup modelu v GAMS	88
Obrázek 11: Dialogové okno doplňku v Excelu	88
Obrázek 12: Přepis testovacího příkladu do listu Excel	89
Obrázek 13: Spuštění doplňku Řešitel	89
Obrázek 14: Parametry Řešitele.....	90
Obrázek 15: Výstup modelu v Excel Řešiteli	90
Obrázek 16: Vzhled GUI programu.....	97
Obrázek 17: Grafické řešení v GUI	99
Obrázek 18: Zadávání testovacího příkladu v GUI	99
Obrázek 19: Výstup testovacího příkladu v GUI.....	100
Obrázek 20: Převod úlohy na její duální formu v GUI.....	101

SEZNAM TABULEK

Tab. 1: Doplnování přídatných a pomocných proměnných.....	17
Tab. 2: Průsečíky jednotlivých omezení s osami soustavy souřadnic	18
Tab. 3: Základní řešení ekvivalentní soustavy rovnic	22
Tab. 4 Výchozí tvar simplexové tabulky	25
Tab. 5: Výchozí simplexová tabulka příkladu (2.1)	28
Tab. 6: Řešení simplexové metody příkladu (2.3).....	29
Tab. 7: Výpočet hodnot účelové funkce pro jednotlivá přípustná řešení	30
Tab. 8: Hodnoty přídatných a pomocných proměnných.....	34
Tab. 9: Výchozí simplexová tabulka pro dvoufázovou simplexovou metodu	35
Tab. 10: I. fáze – minimalizace pomocné účelové funkce f'	35
Tab. 11: Optimální řešení příkladu (2.6)	36
Tab. 12: Výpočet hodnot účelové funkce podle souřadnic.....	38
Tab. 13: Vztah mezi primární a duální úlohou	43
Tab. 14: Určení stínových a redukovaných cen.....	44
Tab. 15: Optimální řešení příkladu (2.3)	44
Tab. 16: Optimální řešení dopadu změn pravých stran příkladu (2.20)	45
Tab. 17: Interpretace redukované ceny v závislosti na typu úlohy.....	46
Tab. 18: Alternativní optimální řešení	48
Tab. 19: Možné kombinace hodnot x_1 a x_2	49
Tab. 20: Neomezená hodnota účelové funkce	50
Tab. 21: Výchozí tvar simplexové tabulky příkladu (3.1).....	53
Tab. 22: Nalezení klíčového sloupce a řádku z Tab. 21	54
Tab. 23: První iterace Tab. 22.....	54
Tab. 24: Výchozí tvar simplexové tabulky příkladu (3.2).....	57
Tab. 25: Nalezení klíčového sloupce a řádku Tab. 24.....	57
Tab. 26: Výpočet první fáze příkladu (3.2)	58
Tab. 27: Výpočet druhé fáze příkladu (3.2).....	58
Tab. 28: Výchozí tvar simplexové tabulky příkladu (3.3).....	61
Tab. 29: Nalezení klíčového sloupce a řádku Tab. 28.....	61
Tab. 30: Výpočet první fáze	62
Tab. 31: Druhá fáze a optimální hodnota	62
Tab. 32: Simplexová metoda příkladu (3.4)	65
Tab. 33: Simplexová metoda příkladu (3.5)	67
Tab. 34: Optimální řešení úlohy alokaci zdrojů	71

Tab. 35: Vyhodnocení zisku pro jednotlivé vrcholy přípustné oblasti	72
Tab. 36: Výživové hodnoty a ceny jednotlivých potravin	73
Tab. 37: Optimální řešení úlohy nutričního problému.....	75
Tab. 38: Parametry cenných papírů [19].....	76
Tab. 39: Optimální řešení úlohy portfolia.....	78
Tab. 40: Možné způsoby řezu tyčí	79
Tab. 41: Optimální řešení úlohy dělení materiálů.....	80
Tab. 42: Alternativní optimální řešení úlohy dělení materiálů	81
Tab. 43: Distribuční náklady mezi středisky a odběrateli (v stovkách Kč/ks)	82
Tab. 44: Optimální řešení úlohy dopravního problému.....	85
Tab. 45: Optimální rozložení dodávek mezi středisky a odběrateli.....	85

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

LP	Lineární programování
Tab.	Tabulka
max	Maximální
min	Minimální
URS	Unrestricted in sign
opt	Optimální
tj.	To jest
tzv	Takzvaný
I.	První
II.	Druhá
resp.	Respektive
ř.	Řádek
\$	Dolar
kJ	Kilojoul
g	Gram
mg	Miligram
Kč	Koruna česká
%	Procent
cm	Centimetr
ks	Kus
s.r.o	Společnost s ručením omezeným

SEZNAM PŘÍLOH

Příloha P I: Simplex method

Příloha P II: Simplex method Qt Designer

PŘÍLOHA P I: SIMPLEX METHOD

Obsahem první přílohy je zdrojový kód programu pro řešení úloh lineárního programování pomocí simplexové metody.

PŘÍLOHA P II: SIMPLEX METHOD QT DESIGNER

Obsahem druhé přílohy je soubor s grafickým návrhem uživatelského rozhraní aplikace vytvořeným v nástroji Qt Designer.