

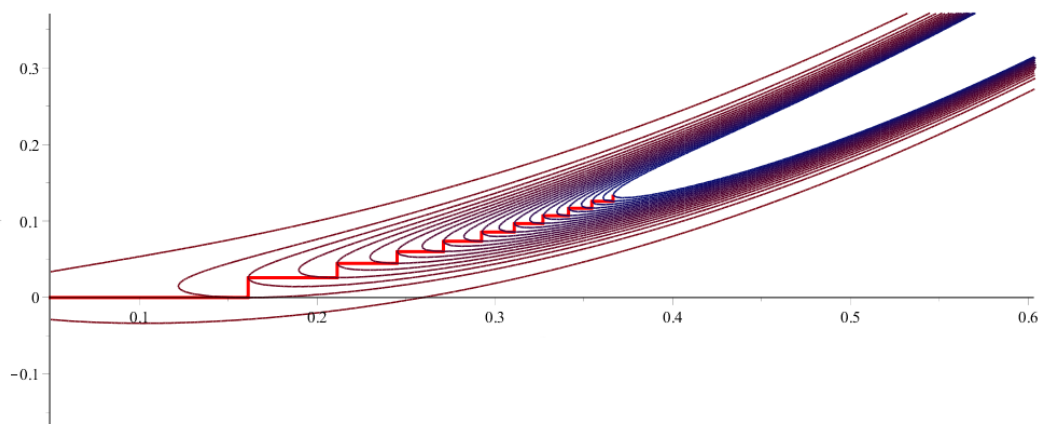
PETR ZEMÁNEK

MUNI Ústav matematiky
a statistiky
SCI

OPTIMALIZACE

ANEB KDYŽ MÉNĚ JE VÍCE

VERZE: 5. května 2020



MUNI
PŘÍRODOVĚDECKÁ
FAKULTA

MASARYKOVA UNIVERZITA

Tato publikace vznikla v rámci projektu Fondu rozvoje MU (MUNI/FR/0901/2020) realizovaného v období 01/2020-12/2020.

Copyright © 2020 **Petr Zemánek**

PUBLIKOVÁNO BY MASARYKOVA UNIVERZITA

TUFTE-LATEX.GOOGLECODE.COM

Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.apache.org/licenses/LICENSE-2.0>. Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

První vydání, květen 2020

OBSAH

1 Úvod a motivace	7
1.1 <i>Pár (3406) slov na úvod, k zamyšlení a třeba i k diskuzi</i>	8
1.2 <i>O co nám zde vlastně půjde?</i>	11
1.3 <i>První (ne)optimalizo(tel)ný historický exkurz</i>	17
2 Lineární programování	35
2.1 <i>Trocha historie (snad) nikoho nezabije</i>	36
2.2 <i>Malá ukázka na začátek</i>	49
2.3 <i>Základem všeho je slušný náčrtek</i>	53
2.4 <i>Teoretické základy aneb být, či nebýt BPB</i>	62
2.5 <i>Dualita aneb peníze až na prvním místě</i>	70
2.6 <i>Simplexový algoritmus</i>	76

2.7	<i>Simplexová tabulka</i>	82
2.8	<i>Hledání výchozího BPB</i>	91
2.9	<i>Lineární lomené programování</i>	93
3	Celočíselné programování	95
3.1	<i>ZLP v kostce (nebo obdélníku)</i>	96
3.2	<i>Závěrečné poznámky za milion dolarů</i>	96
4	Kvadratické programování	97
4.1	<i>Úvod a motivace</i>	98
4.2	<i>Když stačí tužka a pravítko...</i>	103
4.3	<i>Teoretické základy</i>	107
4.4	<i>Hildrethova–d’Esopova metoda</i>	111
4.5	<i>Wolfeho metoda v krátkém tvaru</i>	114
4.6	<i>Wolfeho metoda v dlouhém tvaru</i>	117
5	Dynamické programování	129
5.1	<i>Motivace</i>	130
5.2	<i>Konečněkrokový deterministický proces</i>	132
5.3	<i>Stochastický proces</i>	138
5.4	<i>Nekonečněkrokový deterministický proces</i>	139

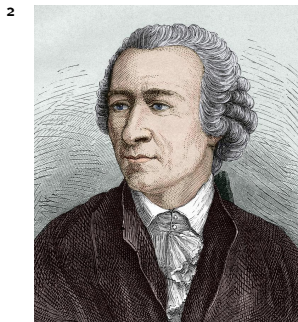
6	Variační počet	141
6.1	<i>Není to jen o nejrychlejší klouzačce</i>	142
6.2	<i>Variuji, variuješ, variujeme</i>	142
6.3	<i>Základní úloha variačního počtu</i>	147
6.4	<i>Úloha o brachistochroně</i>	153
	Litaratura	157

KAPITOLA 1

Úvod a motivace

¹ Jelikož uspořádání vesmíru je na nejvyšše dokonalé a ve skutečnosti dílem nejmoudřejšího Stvořitele, neexistuje ve vesmíru zhola nic, z čeho by nevyzařoval nějaký princip maxima nebo minima.

(For since the fabric of the universe is most perfect and the work of a most wise Creator, nothing at all takes place in the universe in which some rule of maximum or minimum does not appear.)



Leonhard Euler (15. dubna 1707 – 18. září 1783) byl švýcarský matematik, fyzik, astronom a geograf. Byl matematickým pokračovatelem slavného rodu Bernoulliů, který zcela zásadním způsobem přispěl k rozvoji matematiky na přelomu 17. a 18. století. Euler výrazně ovlivnil mnohé (ne-li všechny tehdejší) oblasti matematiky (zejména infinitesimální počet a teorii grafů) a byl také průkopníkem několika nových směrů (zejména variačního počtu a analytické teorie čísel). Je autorem mnoha matematických pojmů a symbolů (obzvláště v matematické analýze). Euler počítal lehce, tak jako člověk dýchá nebo orl létá a jeho následovníci o něm např. řekli, že *Studium Eulerova díla zůstane nejlepší školou pro nejrůznější*

1.1 Pár (3406) slov na úvod, k zamyšlení a třeba i k diskuzi	8
1.2 O co nám zde vlastně půjde?	11
1.3 První (ne)optimalizo(tel)ný historický exkurz	17

CUM ENIM MUNDI UNIVERSI FABRICA SIT PERFECTISSIMA ATQUE A CREATORE SAPIENTISSIMO ABSOLUTA, NIHIL OMNINO IN MUNDO CONTINGIT, IN QUO NON MAXIMI MINIMIVE RATIO QVAEPIAM ELUCEAT.¹

LEONHARD EULER²

Viz Additamentum 1 (str. 245) ve spisu [35], jehož nádherná (a barevná) fotokopie je k dispozici na adrese <https://odkaz.page.link/euler1>.

oblasti matematiky a nemůže je nic nahradit (Carl Friedrich Gauss) a *Čtěte Eulera, čtěte Eulera, je to učitel nás všech* (Pierre-Simon Laplace). Během svého života publikoval cca 500 knih a článků, což představuje za každý rok asi 800 stran textu. Po jeho smrti vyšlo ještě téměř 400 další prací. V roce 1907 (při příležitosti dvousetletého výročí Eulerova narození) iniciovala Švýcarská akademie přírodních věd vydání Eulerova souhrnného díla. Tato kolekce momentálně čítá 72 svazků s desítkami tisíc stran, viz <https://odkaz.page.link/euler6>. Eulerovy práce jsou také dostupné v „jeho“ archivu <https://odkaz.page.link/euler3>. Velmi zajímavý a podrobný průřez Eulerovým dílem můžete nalézt i v krátkých komentářích *How Euler Did It* na adrese <https://odkaz.page.link/euler4>. Pro trochu kratší čtení doporučujeme článek dostupný na adrese <https://odkaz.page.link/euler5>.

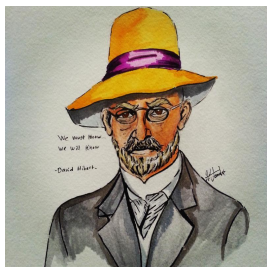
³ V tomto kontextu je pak lecky sousloví „učit se/studovat matematiku“ spíše oxymóronem než čímkoli jiným.

⁴ *Vědět, že číslo π je iracionální, může postrádat jakékoli praktické využití, ovšem pokud to můžeme vědět, bylo by jistě nepřijatelné to nevědět.*

⁵ Edward Charles Titchmarsh (1. června 1899 – 18. ledna 1963) byl významným anglickým matematikem 20. století. Je znám především díky jeho práci v analytické teorii čísel a Fourierové analýze.

⁶ *Musíme vědět. Budeme vědět.* (We must know. We will know.)

7



David Hilbert (23. ledna 1862 – 14. února 1943) byl německý matematik, který velmi výrazně ovlivnil matematiku na přelomu 19. a 20. století. Jeho jméno je mimo jiné spojeno s tzv. *Hilbertovými problémy*. Ty spatřily světlo světa 8. srpna 1900, kdy Hilbert vystoupil během Druhého mezinárodního kongresu matematiků pořádaného

1.1 Pár (3406) slov na úvod, k zamyšlení a třeba i k diskuzi

Matematika má nespočet podob, které však mají jedno společné – touhu po vědění či poznání. Vždyť také základ slova „matematika“ tvoří řecké *mathēma* (μαθημα) znamenající *vědění* či *poznání* (knowledge, study, learning) a *mathēmata* označovala to, co se lze naučit, a tudíž současně i to co může být vyučováno. „Matematikové“ neboli *mathēmatikoi* jsou ti, kteří *milují poznání* (fond of learning). U pythagorejců se takto označovali *studenti* a jejich tehdejšími „konkurenty“ byli *posluchači* – *akusmatikové* (akousmatikoi), kteří se zaměřovali spíše na náboženské a rituální aspekty pythagorejského učení³. Při takovýchto úvahách mi vždy vytanou na mysli slova dvou velkých matematiků 20. století, které vystihují podstatu matematiky. První z nich

*It can be of no practical use to know that π is irrational, but if we can know, it surely would be intolerable not to know.*⁴

je od Edwarda Titchmarsh⁵, viz [64, str. 113]. Ten druhý

*Wir müssen wissen. Wir werden wissen.*⁶

pochází od Davida Hilberta⁷ a zdobí jeho náhrobek v Göttingenu. Tato slova pronesl 8. září 1930 v závěru svého projevu ke Společnosti německých vědců a lékařů při odchodu do důchodu, slyš <https://odkaz.page.link/hilbert1>. Obširnější verzi Hilbertova epitafu nalezneme již v jeho slavné přednášce [48] z roku 1900:

*Diese Überzeugung von der Lösbarkeit eines jeden mathematischen Problems ist uns ein kräftiger Ansporn während der Arbeit; wir hören in uns den steten Zuruf: Da ist das Problem, suche die Lösung. Du kannst sie durch reines Denken finden; denn in der Mathematik gibt es kein Ignorabimus.*⁸

Toto byla reakce na tezi zformulovanou Emilem du Bois-Reymondem⁹ a zpopularizovanou především v jeho knize *Über die Grenzen des Naturerkennens* (O mezích poznání přírody) vydané v roce 1872:

*Ignoramus et ignorabimus.*¹⁰

kteřou Hilbert ve svém projevu z roku 1930 označil za pošetilou či bláhovou. Tato domněnka vyjadřuje postoj, že *člověk má ohraničené a nepřekročitelné hranice možností poznání přírody*, což bývá považováno za projev extrémního agnosticizmu, tj. názoru, že pravdivost některých tvrzení, zejména těch, která se týkají existence či neexistence jakéhokoliv boha, se nedá prokázat ani vyvrátit a že totéž platí i pro další, zejména náboženská a metafyzická tvrzení. Jak už tomu bývá v takovýchto ostrých názorových střetech, pravda bude nejspíše někde uprostřed, neboť v roce 1931 publikoval Kurt Gödel¹¹ dvojici Vět o neúplnosti. První z nich totiž ukazuje, že *v žádné rozumné teorii hovořící o přirozených číslech není dokazatelné vše*, zatímco ta druhá udává konkrétní příklad takového nedokazatelného tvrzení.

Touha po poznání je tedy hnacím motorem matematiky. Ten jsme spustili ve chvíli, kdy jsme začali počítat své prsty na rukou, a od té doby už běží jako (téměř) dokonalé perpetuum mobile,

v Paříži s přednáškou obsahující deset nevyřešených problémů. Tyto problémy spolu s dalšími třinácti pak publikoval v [48], díky čemuž vznikl seznam přitahující pozornost mnoha matematiků. Devět problémů je již zcela vyřešeno, sedm z nich má „částečné“ řešení, formulace dvou problémů je příliš vágní, aby se o jejich řešení dalo rozhodnout, a zbývají 4 problémy, které zůstávají nevyřešené dodnes (včetně Riemannovy hypotézy).

⁸ *Toto přesvědčení o řešitelnosti jakéhokoli matematického problému je pro nás silnou motivací v práci; slyšíme v nás neustálé volání: je tu problém, hledejte řešení. Můžete jej najít čistým myšlením; neboť v matematice neexistuje žádný Ignorabimus.* (This conviction of the solvability of every mathematical problem is a powerful incentive to the worker. We hear within us the perpetual call: There is the problem. Seek its solution. You can find it by pure reason, for in mathematics there is no ignorabimus.)

⁹ *Nevíme a nebudeme vědět.* (We do not know and will not know.)

¹⁰ Emil Heinrich du Bois-Reymond (7. listopadu 1818 – 26. prosince 1896) byl německý lékař a fyziolog, objevitel akčního potenciálu a zakladatel experimentální elektrofyzologie. Jedním z jeho mladších bratrů byl matematik Paul du Bois-Reymond, na kterého ještě později rozhodně narazíme.

¹¹ Kurt Gödel (28. dubna 1906 – 14. ledna 1978) byl matematik česko-rakouského původu (narozený v Brně), který se stal jedním z nejvýznamnějších logiků všech dob. Významné jsou i jeho příspěvky ve fyzice a ve filozofii matematiky.

¹² Niccolò Fontana (1499/1500 – 13. prosince 1557) byl italský matematik a konstruktér, který se jako první věnoval studiu dráhy letu dělových koulí. přezdívaný. Je znám spíše pod přezdívkou Targalia znamenající „koktavec“, což odkazuje na jeho vadu řeči způsobenou úderem šavle francouzského vojáka do čelisti při dobývání jeho rodné Brescie v roce 1512.

¹³ Gerolamo Cardano či Hieronymus Cardanus (24. září 1501 – 20. září 1576), byl italský polyhistor, jehož zájmy a dovednosti sahaly od matematiky přes lékařství, bi-

neboť každé nové poznání obvykle přináší další a další otázky. Existují také různé faktory, které mohou výrazně zvýšit výkon tohoto motoru. Např. jako ve sportu, kdy samozřejmě chcete být lepší, rychlejší a překonávat výkony svých soupeřů. V historii matematiky nalezneme celou řadu problémů, jejichž řešení se podařilo najít jen díky tomu, že se o nich někdo významný zmínil (jako v případě Hilbertova seznamu) nebo dokonce i oznámil znalost takového řešení, ale nechal si jej pro sebe (pokud jej tedy vůbec měl) – třeba když Targalia¹² našel vzorce pro řešení kubických rovnic (dnes známé spíše jako *Cardanovy*¹³ vzorce) jen kvůli matematickému souboji s Antoniem Fiorem¹⁴ v roce 1535, protože si o něm myslel, že je zná (avšak ve skutečnosti neznal, ovládal pouze řešení jednoho typu redukováných kubických rovnic). Druhý příklad stojící za zmínku je slavná *Velká Fermatova věta*¹⁵, o kterou si Fermat¹⁶ v roce 1637 napsal na okraj slavné knihy *Aritmetika* od Diofanta z Alexandrie¹⁷ a zároveň dodal

*Cubum autem in duos cubos, aut quadrato-quadratum in duos quadrato-quadratos, et generaliter nullam in infinitum ultra quadratum potestatem in duos eiusdem nominis fas est dividere cuius rei demonstrationem mirabilem sane detexi. Hanc marginis exigitas non caperet.*¹⁸

Dnes už se asi nedozvíme, zda Fermat skutečně něco tak geniálního objevil nebo se mýlil. Jediný doposud známý důkaz této věty čítá více než 100 stran velmi složité matematiky z různých oblastí a pochází od Andrewa Wilese¹⁹, kterému se jej podařilo dokončit v roce 1994.

Druhým možným „urychlovačem“ může být *krása* matematické poznání, jak se o tom zmiňuje např. Edward Titchmarsh⁵:

*Perhaps the most surprising thing about mathematics is that it is so surprising. . .*²⁰

viz [64, str. 75], nebo na půl v žertu i Hermann Weyl²¹:

*My work has always tried to unite the true with the beautiful and when I had to choose one or the other, I usually chose the beautiful.*²²

viz [33]. K tomu totiž potřebujete mít v sobě alespoň trochu matematického ducha, neboť asi neúplně každý plně docení výsledky založené např. na překvapivém propojení dvou původně zcela nesouvisících oblastí matematiky ilustrující dokonalost matematického soukolí, o kterém se zmiňuje Alfred Whitehead²³:

*The science of pure mathematics, in its modern developments, may claim to be the most original creation of the human spirit.*²⁴

viz [78, str. 20]. Nebo můžeme obdivovat různé „singularity“ typu

$$10^2 + 11^2 + 12^2 = 13^2 + 14^2 \quad 2 = \sqrt{2}^{\sqrt{2}^{\sqrt{2}^{\sqrt{2}^{\dots}}}} \quad \operatorname{tg} 10^\circ = \operatorname{tg} 20^\circ \times \operatorname{tg} 30^\circ \times \operatorname{tg} 40^\circ$$

$$\frac{\pi^2}{6} = \frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \frac{1}{5^2} + \dots \quad e = 2 + \frac{2}{3 + \frac{3}{4 + \frac{4}{5 + \frac{5}{6 + \dots}}}} \quad \int_0^1 \frac{dx}{x^x} = \sum_{k=1}^{\infty} \frac{1}{k^k}$$

ologii, fyziku, astrologii a astronomii až po gamblerství. Je jedním z nejvýznamnějších představitelů rozvoje přírodních věd období renesance. Byl také klíčovou osobností na počátku studia pravděpodobnosti a snad i prvním, kdo „na západě“ využíval binomická čísla a binomickou větu.

¹⁴ Antonio Maria del Fiore (přelom 15. a 16. století) byl italský renesanční matematik, který se od svého učitele Scipioneho del Ferra naučil řešit redukovanou kubickou rovnici $x^3 + px = q$ pro libovolná čísla $p, q > 0$.

¹⁵ Neexistují čísla $x, y, z \in \mathbb{N}$ a $n \in \mathbb{N} \setminus \{1, 2\}$ splňující $x^n + y^n = z^n$.

¹⁶ Pierre de Fermat (někdy mezi 31. říjnem a 6. prosincem 1607 – 12. ledna 1665) byl francouzský právník a matematik, který se zasloužil o rozvoj teorie čísel, pravděpodobnosti a kterému také patří uznání za rozpracování metody hledání extrému křivky jakožto předchůdce infinitesimálního počtu, viz později. Je také znám díky tzv. *Fermatovu principu* z optiky.

¹⁷ Diofantos z Alexandrie byl starověký řecký matematik působící ve 3. století n. l. v egyptské Alexandrii. Je prvním matematikem, který řešil ryze algebraické problémy a zavedl pro ně důmyslný systém značek. Ve své nejdůležitější díle *Aritmetika* popisuje ve 13 knihách mj. všechna známá řešení lineárních a kvadratických rovnic. Ačkoli se nám dodnes z původních 13 knih dochovala pouze polovina, je zcela po zásluze označován za „otce algebry“.

¹⁸ Je nemožné, aby třetí mocnina byla součtem dvou třetích mocnin, čtvrtá mocnina součtem dvou čtvrtých mocnin nebo obecně aby libovolná mocnina stupně vyššího než dva nějakého čísla byla součtem dvou mocnin téhož stupně. Objevil jsem vsutku pozoruhodný důkaz tohoto tvrzení, pro který je ale tento okraj příliš malý, aby se do něj vešel. (It is impossible for a cube to be the sum of two cubes, a fourth power to be the sum of two fourth powers, or in general for any number that is a power greater than the second to be the sum of two like powers. I have discovered a truly marvelous demonstration of this proposition that this margin is too narrow to contain.)

Obvyklá reakce bývá *No a dál?* (což bývá ještě ta přijatelnější verze). Zatímco projevem obdivu k dílu Shakespeara či Mozarta s doplněním nějakého střípku z jejich života mnohdy vzbudíte dojem vzdělaného a kulturního člověka, zmínkou o kráse matematiky podpořené třeba o některou z identit výše vyvoláte nejspíše dojem úplně jiný. Dokonce jsou situace, ve kterých bych důrazně nedoporučoval se o něčem takovém zmiňovat. Na párty to z vás asi hvězdu večera neudělá, a to ani když si k tomu si připravíte výběr těch nejlepších matematických vtipů nebo (nedej bože) přidáte detailní rozbor „situace s elektronickou tužkou“. Z vlastní zkušenosti mohu i potvrdit, že s tím příliš neuspějete ani na první (druhé či třetí – tohle bylo maximum, další pokus už jsem raději nezkoušel) schůzce s partnerem/partnerkou, pokud tedy zrovna nepatříte do téhož $\varepsilon - \delta$ klubu.

„Odhalovat krásu matematiky“ je také velmi oblíbené klišé v dnešních debatách o výuce matematiky. Jenže zkuste si do těch vět místo matematiky dosadit jiný středoškolský předmět. Český jazyk, biologii, chemii, fyziku. . . Copak se něco z toho učí kvůli tomu, abyste tím byli okouzleni? Ani náhodou. Tím třetím a zároveň nejdůležitějším palivem našeho matematického motoru je totiž *užitečnost*. Klasická „školská“²⁵ matematika by nás především měla naučit logickému a analytickému myšlení, což jsou dovednosti zcela nezbytné pro běžný život. Zde se mi vybaví všelijaké kvízy, hádanky či dokonce olympiády s mnohdy velmi nepraktickými úlohami. Mozek je jako sval a potřeba jej procvičovat, jinak „zakrní“ (a moderní technologie nám to příliš neulehčují). Bez ohledu na to, co slyšíme od různých „osobností“ ze všech stran, novodobý člověk je především *Homo Mathematicus*. Kde bychom dnes asi byli bez matematiky? Zkuste si odpovědět sami. . . Velmi výstižně to napsal americký spisovatel Robert Heinlein:

*If it can't be expressed in figures, it is not a science; it is opinion.*²⁶

viz [47, str. 240(???)]. Matematika je dnes jazykem snad už skoro všech věd a zároveň je úžasným nástrojem pro modelování reálného světa, čehož si byl velmi dobře vědom už i slavný Galileo Galilei²⁷:

*La filosofia [della natura] è scritta in questo grandissimo libro che continuamente ci sta aperto dinanzi a gli occhi (io dico l'universo), ma non si può intendere se prima non s'impara a intender la lingua, e conoscere i caratteri ne' quali è scritto. Egli è scritto in lingua matematica, e i caratteri son triangoli, cerchi, ed altre figure geometriche, senza i quali mezi è impossibile a intenderne umanamente parola; senza questi è un aggirarsi vanamente per un oscuro laberinto.*²⁸

viz [42, str. 25]. Oblast, do které se společně v následujících kapitolách vydáme, patří v tomto ohledu k těm nejvyužívanějším – optimalizace. Jen si vzpomeňte na slova Leonharda Eulera z úvodu této kapitoly, která ideálně doplňují předchozí Galileův výrok. A co vy, čtenáři? Kdy jste naposledy zvažovali různé varianty a hledali *nejlepší rozhodnutí, ideální kompromis* či *největší užitek* (ať už se týkalo čehokoli)? Samozřejmě matematika není všemocná, takže asi nedokážeme vyřešit každý rozhodovací problém, na který v životě můžete narazit – jako obvykle totiž platí, že čím realističtější je problém, tím složitější je jeho matematizace a samotné řešení. Proto vám zde asi nepomůžeme optimalizovat vaše studium, spánek, partnerský vztah či milostný život (ačkoli ten naštěstí trápí matematiky/matematicky ze všeho nejméně). Nicméně uvidíme, že s celou řadou mnohem praktičtějších úloh si už celkem snadno poradíme (a díky některým z nich lze i získat Nobelovu cenu), a sem tam odkryjeme i nějaké tajemství přírody. A pokud navíc v

¹⁹ Sir Andrew John Wiles (*11. dubna 1953) je anglický matematik, profesor na Oxfordské univerzitě a (mimo jiné) držitel Abelovy medaile z roku 2016.

²⁰ *Tím nepřekvapivějším na matematice je asi to, že je tak překvapující...*

²¹ Hermann Klaus Hugo Weyl (9. listopadu 1885 – 8. prosince 1955) byl německý matematik, teoretický fyzik a filosof, který patřil k nejvlivnějším matematikům 20. století. Ačkoli většina jeho profesního života je spojena se Spolkovou vysokou technickou školou v Curychu (ETH Zurich) a univerzitou v Princetonu (centrem Institute for Advanced Study), je brán především jako další pokračovatel matematické školy Univerzity v Göttingenu, kde v roce 1908 získal doktorát pod vedením D. Hilberta.

²² *Ve své práci jsem se vždy snažil spojit pravdu s krásou, a když jsem si musel vybrat jedno nebo druhé, obvykle jsem si vybral tu krásu.*

²³ Alfred North Whitehead (15. února 1861 – 30. prosince 1947) byl anglický matematik a filosof. Je považován za jednoho z nejvýznamnějších anglo-amerických filosofů 20. století. Společně se svým bývalým studentem Bertrendem Russellem napsal třívazkové dílo *Principia Mathematica*, které je považováno za jedno z nejdůležitějších děl matematické logiky 20. století.

²⁴ *Čistou matematiku v její moderní podobě můžeme prohlásit za nejoriginálnější výtvor lidského ducha.*

²⁵ tj. v rámci primárního a sekundárního vzdělávání na základních a středních školách

²⁶ *Pokud to nelze vyjádřit v číslech, není to věda; je to pouhý názor.*



Galileo di Vincenzo Bonaiuti de Galilei (15. února 1564 – 8. ledna 1642) byl italský astronom, fyzik a konstruktér. Bývá označován za „otce pozorovací astronomie“ a „otce moderní fyziky“ či dokonce „otce moderní vědy“.

nabídnutém výkladu občas spatříte i něco krásného, bude naše putování zcela *optimální*, neboť jak řekl Godfrey Hardy²⁹:

*The mathematician's patterns, like the painter's or the poet's, must be beautiful; the ideas, like the colours or the words, must fit together in a harmonious way. Beauty is the first test: there is no permanent place in the world for ugly mathematics.*³⁰

viz [46, str. 85]. Ovšem dopředu upozorňujeme, že s jednou (byť velmi výraznou) výjimkou se zde rozhodně nebudeme zabývat nějakou hlubokou matematickou historií. Přestože se nám to může z dnešního pohledu zdát snad i nepochopitelné, vždyť všelijaké optimalizační úlohy a rozhodovací procesy jsou jistě staré jako lidstvo samo, jejich systematické studium začalo až ve druhé polovině 20. století a výzkum v těchto oblastech je neustále velmi aktivní. Např. v nejznámější matematické databázi *MathSciNet* je toto obsahem kategorií 49, 65K a 90, ve kterých najdeme od roku 2010 celkem 145347 záznamů, zatímco od roku 1901 to je 189180 záznamů. Nahlédneme-li do druhé světoznámé matematicko-fyzikální databáze *ArXiv*, tak v ní nalezneme třeba jen za leden 2020 více než 150 nových záznamů v kategorii *Optimization and Control*.

1.2 O co nám zde vlastně půjde?

Optimalizační úlohou můžeme v tom nejobecnějším pojetí rozumět problém nalezení takového bodu v dané (tzv. *připustné*) množině \mathcal{X} , že odpovídající „*účelové zobrazení*“³¹ $F : \mathcal{X} \rightarrow \mathbb{R}$ nabývá v jistém smyslu *optima* (typicky minima či maxima), tj. úlohu

$$F(x) \rightarrow \text{opt}, \quad x \in \mathcal{X}. \quad (1.2.1)$$

Řešením této úlohy pak rozumíme bod $x^* \in \mathcal{X}$ splňující³² podmínku $F(x) \triangleright \triangleleft F(y)$ pro všechna $y \in \mathcal{X} \setminus \{x^*\}$. Takový bod ale rozhodně nemusí být jediný, takže řešením dané úlohy je spíše množina bodů

$$\mathcal{X}^* = \text{arg opt}_{x \in \mathcal{X}} F(x) := \{x \in \mathcal{X} \mid F(x) \triangleright \triangleleft F(y) \text{ pro všechna } y \in \mathcal{X}\}.$$

Avšak v některých případech nemusí být nutné (nebo dokonce možné) nějaký takový bod x^* najít a místo toho se spokojíme pouze s nalezením *optimální hodnoty účelového zobrazení*, tj. čísla

$$F^* := \text{opt} \{F(x) \mid x \in \mathcal{X}\},$$

kde symbol *opt* znamená buď hledání minima (v obecnějším případě infima) nebo maxima (případně suprema). Navíc není příliš podstatné, zda je naším úkolem minimalizovat nebo maximalizovat „zobrazení“ F , neboť úlohy

$$F(x) \rightarrow \min, \quad x \in \mathcal{X} \quad \text{a} \quad -F(x) \rightarrow \max, \quad x \in \mathcal{X}$$

jsou *ekvivalentní*, tj. odpovídající množiny řešení \mathcal{X}^* jsou totožné (obě úlohy buď mají tatáž řešení nebo žádná z nich nemá řešení) a v případě $\mathcal{X}^* \neq \emptyset$ zjevně platí

$$F^* = \min \{F(x) \mid x \in \mathcal{X}\} = -\max \{-F(x) \mid x \in \mathcal{X}\}.$$

My se v našich úvahách omezíme především na minimalizaci, což je typické pro úlohy s fyzikálním podtextem (minimalizace času, práce, „akce“), zatímco maximalizační úlohy mívají obvykle ekonomický podtext (maximalizace zisku, užitku)³³.

²⁸ *Filozofie [přírody] je psána v této úžasné knize (mluvím o vesmíru), kterou máme neustále otevřenou před očima. Ovšem nelze ji porozumět dříve, než se naučíme jazyk a znaky, ve kterých je napsána. Je psána jazykem matematiky a jeho znaky jsou trojúhelníky, kruhy a další geometrické útvary, bez nichž nelze lidsky rozumět, bez nichž je lidsky nemožné porozumět jedinému slovu; bez nich se člověk marně potuluje temným labyrintem.*

(Philosophy [of nature] is written in this grand book, the universe, which stands continually open to our gaze. But the book cannot be understood unless one first learns to comprehend the language and read the letters in which it is composed. It is written in the language of mathematics, and its characters are triangles, circles and others geometric figures without which it is humanly impossible to understand a single word of it; without these, one wanders about in a dark labyrinth.)

²⁹ Godfrey Harold Hardy (7. února 1877 – 1. prosince 1947) byl anglický matematik, který se věnoval především teorii čísla a matematické analýze. Jeho jméno je úzce spojeno s tzv. Hardyho–Weinbergovým principem v populační genetice. Mimo matematiku je Hardy znám díky své esaji [46], která bývá považována za jeden z nejlepších vhladů do matematikovy mysli určeným pro laiky. Když se slavný maďarský matematik Pál Erdős zeptal Hardyho, co bylo jeho největším příspěvkem na poli matematiky, Hardy bez rozmyšlení odpověděl, že to bylo objevení indického matematika Šrínívási Rámanudžana, jehož učitelem byl od roku 1914.

³⁰ *Matematikovy vzorce, stejně jako ty malířovy či básnickovy, musí být krásné; myšlenky, stejně jako barvy nebo slova, musí být v harmonickém souladu. Krása je první zkouškou: na světě není trvalé místo pro ošklivou matematiku.*

Takto formulovanou obecnou optimalizační úlohu (1.2.1) můžeme v závislosti na charakteru vstupních veličin z množiny \mathbb{X} a „zobrazení“ \mathbb{F} rozdělit do dvou základních skupin:

- (i) *parametrická optimalizace*, což pro nás bude zejména *matematické programování*³⁴.
- (ii) *funkční optimalizace*, což pro nás bude především *variální počet*,

V parametrické optimalizaci je přípustná množina $\mathbb{X} := X \subseteq \mathbb{R}^n$ a obvykle bývá vymezena podmínkami danými soustavou nějakých algebraických nebo transcendentních rovnic či nerovnic. Účelovým „zobrazením“ \mathbb{F} je reálná funkce $f: \mathbb{R} \rightarrow \mathbb{R}$ definovaná (alespoň) na přípustné množině. V závislosti na kontextu nalezneme pro funkci f různá označení jako *účelová*, *cílová*, *ztrátová*, *nákladová*, *nepřímá užitková*, *užitková*, *energetická* atd. Úloha (1.2.1) pak znamená najít bod $x^* \in X$, ve kterém daná reálná funkce f nabývá své nejmenší (příp. největší) hodnoty mezi všemi body $x \in X$, tj. máme řešit úlohu

$$f(x) \rightarrow \min, \quad x \in X, \quad (1.2.2)$$

příčemž je velmi výhodné množinu X vyjádřit jako (je-li to možné)

$$X = \{x \in \mathbb{R}^n \mid g_1(x) \leq 0, \dots, g_L(x) \leq 0 \text{ \& } h_1(x) = 0, \dots, h_K(x) = 0\}$$

pro $K, L \in \mathbb{N} \cup \{0\}$ a dané funkce $g_1, \dots, g_L, h_1, \dots, h_K$, tj. hledáme

$$x^* \in \operatorname{argmin}_{x \in X} f(x) = \{x \in X \mid f(x) \leq f(y) \text{ pro všechna } y \in X\}.$$

Úlohu (1.2.2) pak čteme jako:

Najdi bod x^ , v němž reálná funkce f dosahuje nejmenší hodnoty (tj. minima) mezi všemi body $x \in D(f) \subseteq \mathbb{R}^n$, které vyhovují soustavě L nerovností*

$$g_1(x) \leq 0 \text{ \& } g_2(x) \leq 0 \text{ \& } \dots \text{ \& } g_L(x) \leq 0$$

a soustavě K rovností

$$h_1(x) = 0 \text{ \& } h_2(x) = 0 \text{ \& } \dots \text{ \& } h_K(x) = 0.$$

Bod x^* pak této úloze představuje *optimální řešení* a zejména v případě $X \subseteq \mathbb{R}^n$ také také *optimální strategii* nebo *návrhový vektor* matematického programu. Místo úlohy (1.2.2) můžeme případně hledat řešení „pouze“ úlohy

$$f^* = \inf_{x \in X} f(x) \quad (1.2.3)$$

Pro úlohu (1.2.2) v jejím obecném pojetí samozřejmě nemáme zaručenou existenci jejího řešení, zatímco v případě úlohy (1.2.3) jej najdeme vždy – buď to bude nějaké reálné číslo $x^* \in X$ nebo $x = \pm\infty$, přičemž hodnota $x^* = +\infty$ odpovídá³⁵ situaci $X = \emptyset$, kdežto $x^* = -\infty$ znamená, že funkce f není zdola ohraničená na X . Pokud ale existuje konečné řešení úlohy (1.2.2), pak to není nic jiného než hledání vrstevnice funkce f na úrovni f^* . Úlohu (1.2.2) můžeme navíc rozdělit ještě do dvou subkategorií:

- úlohy bez omezujících podmínek (tzv. *volné extrém*), tj. $X = \mathbb{R}^n$, např. $x^2 + 1 \rightarrow \min$
- úlohy s omezujícími podmínkami (tzv. *vázané extrém*), tj. $X \subseteq \mathbb{R}^n$, např. $x \cos y \rightarrow \min$ pro $[x, y] \in [-5, 5] \times \mathbb{R}$.

S oběma typy úloh jsme se již setkali v základním kurzu matematické analýzy a my si nyní některé pojmy připomeneme.

DEFINICE 1.2.1

³¹ Zde jsou úvozovky zcela na místě, neboť skutečné „zobrazení“ by pro naše úvahy bylo příliš obecné.

³² Symbol $\triangleright\triangleleft$ nahrazuje některou z nerovností \leq , \geq , $<$ nebo $>$, jejichž konkrétní volba je odvislá od dané úlohy.

³³ Toto rozdělení mezi fyziku a ekonomii samozřejmě není nikterak striktní, můžeme mít např. maximalizaci účinnosti

VĚTA 1.2.2

v elektrickém obvodu nebo minimalizaci nákladů. Toto ale pochopitelně nejsou jediné dvě oblasti, kde se můžeme s optimalizačními problémy setkat.

³⁴ Z dnešního pohledu se může zdát označení „programování“ trochu nešťastné a zavádějící. Jedná se o pozůstatek vojensky motivované terminologie, která odkazuje na *plánování* či *rozvrhování* tréninků, logistiky či rozmístění mužstva. Více se o tom dozvíte v další sekci a na začátku následující kapitoly.

VĚTA 1.2.3

³⁵ Jelikož infimum rozumíme největší dolní závorku dané množiny, takže zcela „přirozeně“ může brát $\inf \emptyset = +\infty$.

³⁶ XYZ tohle si zaslouží podrobnější rozbor!!!

Bod $x^* \in X$ nazveme bodem *globálního minima* funkce f na množině $X \subseteq D(f)$, tj. řešením úlohy (1.2.2), pokud $f(x^*) \leq f(x)$ pro všechny $x \in X$. Bod $x^* \in X$ nazveme bodem *lokálního minima* funkce f na X , neboli *lokálním řešením* úlohy (1.2.2), jestliže existuje takové ε -okolí bodu x^* , tj.

$$\mathcal{O}_\varepsilon(x^*) := \{x \in \mathbb{R}^n \mid \|x - x^*\| < \varepsilon\},$$

že $f(x^*) \leq f(x)$ pro každé $x \in X \cap \mathcal{O}_\varepsilon(x^*)$. V případě ostrých nerovností hovoříme o *ostrých* globálních/lokálních minimech. Analogicky definujeme globální/lokální (ostrá) *maxima*.

Jak je ale najít? V případě diferencovatelné funkce f nám pomůže tzv. *Fermatova věta*^{16,36}

Nechť pro funkci $f : \mathbb{R}^n \rightarrow \mathbb{R}$ existují všechny parciální derivace 1. řádu v bodě x^* , ve kterém má funkce f lokální extrém na \mathbb{R}^n . Potom x^* je *stacionárním bodem* funkce f , tj.

$$\text{grad } f(x^*) := \begin{pmatrix} f_{x_1}(x^*) \\ f_{x_2}(x^*) \\ \vdots \\ f_{x_n}(x^*) \end{pmatrix} = 0.$$

Velmi dobře ale víme, že toto je pouze nutná podmínka pro existenci lokálního extrému.

Má-li funkce $f : \mathbb{R}^n \rightarrow \mathbb{R}$ v bodě $x^* \in \mathbb{R}^n$ a nějakém jeho okolí spojitě parciální derivace 2. řádu, $\text{grad } f(x^*) = 0$ a symetrická $n \times n$ *Hessova matice*

$$\nabla^2 f(x^*) = \left(\frac{\partial^2 f}{\partial x_i \partial x_j}(x^*) \right)_{i,j=1,\dots,n}$$

je pozitivně definitní, pak bod x^* je bodem ostrého lokálního minima funkce f na \mathbb{R}^n .

Jak se pozná *pozitivně definitní* matice?! Platí také opačná implikace? Jak se pozná *pozitivně semidefinitní* matice?!

A co existence globální extrémů? Viva la Weierstrass!

VĚTA 1.2.4

Nechť $X \subseteq \mathbb{R}^n$ je kompaktní³⁷ množina a $f : X \rightarrow \mathbb{R}$ je spojitá funkce na X . Pak funkce f nabývá své největší a nejmenší hodnoty na X (tj. globálních extrémů).

³⁷ Podmnožina \mathbb{R}^n je kompaktní právě tehdy, když je uzavřená a ohraničená

Dokážeme extrémy „lokalizovat“? Tohle je Fermatova metoda.

VĚTA 1.2.5

Nechť $X \subseteq \mathbb{R}^n$ je kompaktní množina a funkce $f : X \rightarrow \mathbb{R}$ je diferencovatelná na X . Pak f nabývá svého maxima a minima na X buď ve stacionárním bodě ležícím uvnitř X nebo v některém hraničním bodě množiny X .

Je to ale praktické? Neexistuje nějaká efektivnější metoda?

VĚTA 1.2.6

Nechť f, g_1, \dots, g_m jsou funkce třídy C^1 na otevřené množině $\mathcal{U} \subseteq \mathbb{R}^n$, kde $m \in \{1, \dots, n\}$. Uvažme množinu \mathcal{M} danou jako

$$\mathcal{M} := \bigcap_{i=1}^m \{x \in \mathbb{R}^n \mid g_i(x) = 0\} \subseteq \mathcal{U},$$

přičemž vektory $\text{grad } g_1(x), \dots, \text{grad } g_m(x)$ jsou lineárně nezávislé (a tedy nutně $m \leq n$) pro všechna $x \in \mathcal{M}$, tj. Jacobiho matice

$$DG(x) = D(g_1(x), \dots, g_m(x))^T = \begin{pmatrix} \frac{\partial g_1}{\partial x_1}(x) & \dots & \frac{\partial g_1}{\partial x_n}(x) \\ \vdots & \ddots & \vdots \\ \frac{\partial g_m}{\partial x_1}(x) & \dots & \frac{\partial g_m}{\partial x_n}(x) \end{pmatrix} \in \mathbb{R}^{m \times n}$$

má plnou hodnotu, tj. $\text{rank } DG(x) = m$. Je-li bod $x^* \in \mathcal{M}$ lokálním extrémem funkce f na množině \mathcal{M} , pak existují taková čísla $\lambda_1^*, \dots, \lambda_m^* \in \mathbb{R}$ (tzv. *Lagrangeovy multiplikátory*), že x^* je stacionárním bodem *Lagrangeovy funkce*

$$L(x, \lambda^*) = f(x) + \sum_{i=1}^m \lambda_i^* g_i(x),$$

tj. $\text{grad}_x L(x^*, \lambda^*) = 0$ neboli

$$\frac{\partial f}{\partial x_j}(x^*) + \sum_{i=1}^m \lambda_i^* \frac{\partial g_i}{\partial x_j}(x^*) = 0 \quad \text{pro všechna } j = 1, \dots, n.$$

Co je množina \mathcal{M} ? Množina

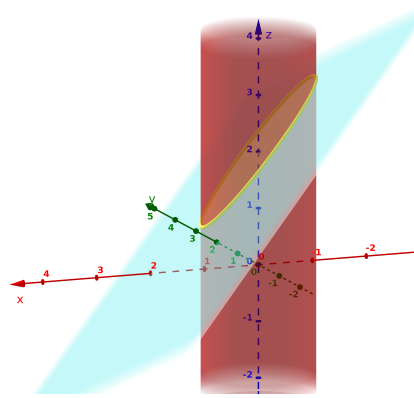
$$\{x \in \mathbb{R}^n \mid g_1(x) = 0\}$$

obvykle představuje nějakou nadplochu, tj. „útvár“ dimenze $n - 1$. Přidáním další nadplochy dostaneme

$$\{x \in \mathbb{R}^n \mid g_1(x) = 0\} \cap \{x \in \mathbb{R}^n \mid g_2(x) = 0\},$$

což bude mít dimenzi $n - 2$ atd. Tedy \mathcal{M} bude $(n - m)$ -dimenzionální plocha v \mathbb{R}^n . Např.

$$g_1(x, y, z) = x + y + z - 2, \quad g_2(x, y, z) = x^2 + y^2 - 1$$



Co znamená, že x^* je stacionárním bodem funkce $L(x, \lambda^*)$?

A postačující podmínka? K tomu, aby bod x^* byl lokálním vázaným extrémem, rozhodně není nutné (ale postačující), aby byl současně lokálním extrémem funkce $L(x, \lambda^*)$. Avšak je-li x^* vázaným lokálním minimem, nutně $\nabla^2 L(x^*, \lambda) \geq 0$ na $\text{Ker}(DG(x^*))$.

VĚTA 1.2.7

Nechť f, g_1, \dots, g_m jsou funkce třídy C^2 na otevřené množině $\mathcal{U} \subseteq \mathbb{R}^n$, kde $m \in \{1, \dots, n\}$. Uvažme množinu \mathcal{M} danou jako

$$\mathcal{M} := \bigcap_{i=1}^m \{x \in \mathbb{R}^n \mid g_i(x) = 0\} \subseteq \mathcal{U},$$

přičemž vektory $\text{grad } g_1(x), \dots, \text{grad } g_m(x)$ jsou lineárně nezávislé pro všechna $x \in \mathcal{M}$. Jestliže pro bod $x^* \in \mathcal{M}$ existují multiplikátory $\lambda_1^*, \dots, \lambda_m^* \in \mathbb{R}$ takové, že platí následující podmínky

- (i) Lagrangeova funkce $L(x, \lambda^*)$ má v bodě x^* stacionární bod, tj. $\text{grad}_x L(x^*, \lambda^*) = 0$,
- (ii) Hessova matice $\nabla^2 L(x^*, \lambda^*)$ je pozitivně definitní na

$$\text{Ker}(DG(x^*)) = \text{span}\{\text{grad } g_1(x^*), \dots, \text{grad } g_m(x^*)\}^\perp = \bigcap_{i=1}^m \{x \in \mathbb{R}^n \mid x \perp \text{grad } g_i(x^*)\},$$

pak funkce f má v bodě x^* ostré vázané lokální minimum.

Toto je velmi speciální případ úlohy matematického programování, ve které jsou všechna omezení pouze ve tvaru rovností a žádná proměnná není omezena na znaménko. V kapitole ?? si ukážeme, jak řešit zcela obecnou úlohu matematického programování. Ovšem existují i další speciální typy úloh matematického programování, kterým se budeme věnovat nejdříve.

- (i) *Lineární programování*: funkce f je lineární a množina X je vymezena soustavou lineárních rovnic a nerovnic.
- (ii) *Celočíselné programování*: jako úloha lineárního programování doplněná o požadavek celočíselných hodnot jednotlivých proměnných.
- (iii) *Kvadratické programování*: funkce f je kvadratická forma (plus lineární funkce) a množina X je opět vymezena soustavou lineárních rovnic a nerovnic.

Ve funkční optimalizaci je přípustná množina \mathcal{X} prostorem funkcí (skalárních nebo vektorových) n -rozměrné proměnné definovaných na nějaké množině $\Omega \subseteq \mathbb{R}^n$, které v závislosti na dané úloze mohou mít jisté kvalitativní vlastnosti (spojitost, diferencovatelnost apod.) a splňovat další (tzv. vedlejší) podmínky v podobě soustav algebraických, transcendentních, diferenciálních nebo integrálních rovnic či nerovnic. Účelové zobrazení \mathbb{F} pak je *funkcionál*, tj. zobrazení přiřazující každé funkci z přípustné množiny \mathcal{X} nějaké reálné číslo. Takový funkcionál může mít mnoho podob, např. tím nejjednodušším by mohlo být vyčíslení každé funkce z \mathcal{X} v jednom konkrétním bodě z množiny Ω , ale mnohem častější je spíše funkcionál ve tvaru nějakého určitého integrálu. Úloha (1.2.1) pak znamená najít funkci $f \in \mathcal{X}$, pro kterou daný funkcionál nabývá největší/nejmenší hodnotu mezi všemi funkcemi z přípustné množiny.

Ukážeme si jeden ilustrativní příklad s původním Eulerovým elegantním řešením. Pro historické podrobnosti viz následující sekci.

Příklad 1.2.8

Mezi všemi funkcemi třídy $C^1[a, b]$ najít tu, jejíž graf je nejkratší spojnicí bodů $A = [a, \alpha]$ a $B = [b, \beta]$, tj.

$$\mathcal{J}[f] = \int_a^b \sqrt{1 + [f'(x)]^2} dx \rightarrow \min.$$

Řešení. Nejdříve místo speciálního funkcionálu určujícího délku grafu funkce uvažme zcela obecný funkcionál, tj.

$$\mathcal{J}[f] = \int_a^b F(x, f(x), f'(x)) dx$$

pro vhodnou funkci $F: \mathbb{R}^3 \rightarrow \mathbb{R}$. Rozdělme si nyní uzavřený interval $[a, b]$ na $n + 1$ stejně dlouhých podintervalů, tj.

$$a = x_0 < x_1 < \dots < x_n < x_{n+1} = b,$$

takže délka každého takového podintervalu je

$$\Delta x = x_{i+1} - x_i = \frac{b - a}{n + 1}.$$

Nahradíme-li graf funkce $f(x)$ lomenou čarou s vrcholy v bodech

$$[x_0, f(x_0)] \ \& \ [x_1, f(x_1)] \ \& \ \dots \ \& \ [x_{n+1}, f(x_{n+1})].$$

pak můžeme původní funkcionál

$$\mathcal{J}[f] = \sum_{j=0}^n \int_{x_0+j\Delta x}^{x_0+(j+1)\Delta x} F(x, f(x), f'(x)) dx$$

aproximovat konečným součtem

$$\mathcal{J}[f] = \mathcal{J}(f_1, \dots, f_n) := \sum_{j=0}^n F\left(x_j, f_j, \frac{f_{j+1} - f_j}{\Delta x}\right) \Delta x,$$

kteřý je funkcí n -tice proměnných $f_1 := f(x_1), \dots, f_n := f(x_n)$ ³⁸. Jaký bude mít efekt zvýšení či snížení hodnoty některého f_i ? Zvolme se jednu z nich a označme ji f_k . Jelikož

³⁸ Nezapomínejte, že hodnoty $f_0 := f(x_0) = f(a)$ a $f_{n+1} := f(x_{n+1}) = f(b)$ jsou pevně dány.

člen f_k se v sumě $\mathcal{J}(f_1, \dots, f_n)$ objevuje pouze ve sčítacích s indexy $j = k$ a $j = k - 1$, dostaneme parciálním derivováním vzhledem k tomuto f_k výraz

$$\frac{\partial \mathcal{J}}{\partial f_k} = F_f \left(x_k, f_k, \frac{f_{k+1} - f_k}{\Delta x} \right) \Delta x + F_{f'} \left(x_{k-1}, f_{k-1}, \frac{f_k - f_{k-1}}{\Delta x} \right) - F_{f'} \left(x_k, f_k, \frac{f_{k+1} - f_k}{\Delta x} \right). \quad (1.2.4)$$

Pro nalezení extrému bychom pak obvykle položili tuto derivaci rovnu nule pro každé k . Jenže bychom chtěli následně udělat i limitní přechod pro $n \rightarrow \infty$, v kterémžto případě je $\Delta x \rightarrow 0$ a hodnota pravé strany rovnice (1.2.4) půjde k nule. Takže bychom dostali rovnost $0 = 0$, která je sice pravdivá, ale bohužel i nepříliš užitečná. Chceme-li získat trochu méně triviální výsledek, zkusme nejdříve vydělit obě strany rovnosti (1.2.4) výrazem Δx , čímž dostaneme

$$\frac{1}{\Delta x} \frac{\partial \mathcal{J}}{\partial f_k} = F_f \left(x_k, f_k, \frac{f_{k+1} - f_k}{\Delta x} \right) - \frac{1}{\Delta x} \left[F_{f'} \left(x_k, f_k, \frac{f_{k+1} - f_k}{\Delta x} \right) - F_{f'} \left(x_{k-1}, f_{k-1}, \frac{f_k - f_{k-1}}{\Delta x} \right) \right].$$

Odtud nyní pro $n \rightarrow \infty$ a $\Delta x \rightarrow 0$ získáme tzv. *variální derivaci*

$$\frac{\delta \mathcal{J}}{\delta f} = F_f(x, f, f') - \frac{d}{dx} F_{f'}(x, f, f'),$$

kteřá hraje pro funkcionály tutéž roli jako parciální derivace pro funkce více proměnných. Proto má-li být funkce f lokální extrém, očekáváme, že její hodnota bude nulová pro každé x , což nás přivádí k *Eulerově-Lagrangeově* rovnici

$$\frac{\partial F}{\partial f} - \frac{d}{dx} \left(\frac{\partial F}{\partial f'} \right) = 0 \quad (1.2.5)$$

Nyní se vraťme k původnímu problému, kde

$$F(x, f, f') = \sqrt{1 + [f'(x)]^2}.$$

Jelikož tento výraz nezávisí na f , dostaneme z (1.2.5) požadavek

$$\frac{d}{dx} \left(\frac{f'}{\sqrt{1 + [f'(x)]^2}} \right) = 0 \quad \text{neboli} \quad \frac{f'}{\sqrt{1 + [f'(x)]^2}} = \text{konst.}$$

To je možné ale pouze v případě, že

$$f' \equiv C$$

pro libovolné $C \in \mathbb{R}$, a tedy hledaná funkce je nutně tvaru

$$f(x) = Cx + D.$$

Dosažením souřadnice krajních bodů pak dostaneme řešení

$$f(x) = \frac{\alpha - \beta}{a - b} (x - b) + \beta.$$

1.3 První (ne)optimalizo(tel)ný historický exkurz⁴²

⁴² *Kdo ovládá minulost, ovládá budoucnost.* (Who controls the past controls the future.)

George Orwell: 1984 (str. 25)

Jak už jsme se zmínili, skutečně systematickému studiu různých optimalizačních problémů a rozhodovacích procesů se matematici začali věnovat až ve druhé polovině 20. století. Toto ovšem

⁴³ Toto můžeme chápat i jako projev toho, co český geograf a popularizátor vědy, dr. Václav Cílek, označuje za *inteligenci přírody*.

44



Joseph-Louis Lagrange (25. ledna 1736 – 10. dubna 1813) byl dalším důležitým matematikem a astronomem 18. století, který výrazně ovlivnil především rozvoj matematické analýzy a teorie čísel a také klasickou i nebeskou mechaniku. Je jedním ze 16 matematiků, jejichž jméno je napsáno na Eiffelově věži. V roce 1766 na doporučení Eulera a francouzského matematika Jeana-Baptista le Ronda d'Alemberta stanul v čele Pruské akademie věd v Berlíně, kde nahradil práce Eulera. Zde působil více než 20 let na, během nichž vykonal mnoho práce a získal řadu ocenění Francouzské akademie věd.

⁴⁵ Sir William Rowan Hamilton (4. srpna 1805 – 2. září 1865) byl irský matematik, profesor astronomie na Trinity College v Dublinu, a královský astronom. Věnoval se především optice, klasické mechanice a algebře. Ačkoliv nebyl fyzik (sám se považoval za čistého matematika), svoji práci ovlivnil především právě fyziku, a to zejména přeformulováním newtonovské mechaniky, která je nyní spojována především s jeho jménem. Je autorem algebraického přístupu ke komplexním číslům, čímž snad definitivně vyřešil hádanku ohledně významu symbolu i . Tato práce jej přivedla k jeho druhému klíčovému výsledku – zavedení kvaternionů.

⁴⁶ Carl Gustav Jacob Jacobi (10. prosince 1804 – 18. února 1851) byl německý matematik, který zásadně přispěl k rozvoji teorie eliptických funkcí, diferenciálních rovnic, determinantů a teorie čísel. Jacobi byl prvním židovským matematikem, který byl jmenován profesorem na německé univerzitě.

neplatí pro úlohy variačního počtu, jejichž teorie byla vybudována mnohem dříve. Abychom se seznámili s největšími osobnostmi, které zásadně ovlivnili rozvoj různých oblastí optimalizace, a pochopili některé souvislosti či jejich motivaci, uděláme si v závěrečné části této kapitoly malý historický exkurz, který však bude sledovat spíše časovou souslednost nežli váhu jednotlivých témat v dalších kapitolách.

Před chvílí jsme viděli původní Eulerovo řešení pro nalezení nejkratší křivky v rovině. S tímto problémem také úzce souvisí tzv. *Fermatův princip*, který vznikl při řešení problému geometrické optiky (1662). Podle tohoto principu si světelný paprsek ze všech možných trajektorií mezi dvěma body vždy vybírá právě tu, podél níž se dostane z výchozího bodu do cílového bodu za nejkratší čas (ve skutečnosti spíše za „extrémní čas“, tj. po jiné dráze by to vždy trvalo delší/kratší/stejnou dobu). Jako obvykle i zde si příroda najde to „nejekonomičtější“ řešení⁴³ – vždyť čas jsou peníze. Máme-li opticky nehomogenní rovinu, pak rychlost světla $c(x, y)$ se mění v každém bodě v závislosti na optických vlastnostech. Protože rychlost je derivace dráhy podle času, je délka křivky $y = u(x)$ určené světelným paprskem rovna

$$c(x, u(x)) = \frac{ds}{dt} = \sqrt{1 + [u'(x)]^2} \frac{dx}{dt}.$$

Integrováním od počátku do cíle pak získáme celkový čas pohybu podél této křivky, tj.

$$T[u] = \int_0^T dt = \int_a^b \frac{dt}{dx} dx = \int_a^b \frac{\sqrt{1 + [u'(x)]^2}}{c(x, u(x))} dx.$$

Fermatův princip (nebo též *princip nejkratšího času*) pak říká, že z bodu $A = (a, \alpha)$ do bodu $B = (b, \beta)$ se světelný paprsek dostane podél křivky $y = u(x)$, která minimalizuje funkcionál $T[u]$ vzhledem k okrajovým podmínkám

$$u(a) = \alpha \quad \& \quad u(b) = \beta.$$

Jestliže je prostředí homogenní (např. vakuum při zanedbání vlivu gravitace vzhledem k teorii relativity), pak $c(x, y) \equiv c$ a $T[u] = \frac{1}{c} \ell[u]$, tedy hledaným minimem bude opět vhodná přímka. V případě nehomogenního prostředí již řešení tak zřejmé není a bylo by vhodné mít systematickou metodu pro řešení takového minimalizačního problému. Navíc všechny známé zákony optiky, designu optických čoček, ostření, lámání, odchylky atd. jsou vlastně důsledkem geometrických a analytických vlastností řešení Fermatova principu. Tato úvaha inspirovala vědce k hypotéze, že každý děj v přírodě probíhá tak, aby se během tohoto procesu minimalizovala jistá veličina. Ovšem dlouho se nevědělo, o kterou veličinu se jedná. Až v roce 1760 Joseph Lagrange⁴⁴ poprvé přesně zformuloval *princip nejmenší akce* pro konzervativní mechanické systémy a vymezil platnost tohoto principu. Tento princip zobecnil Rowan Hamilton⁴⁵ i pro systémy nekonzervativní, kdy transformoval Lagrangeovy rovnice (které jsou diferenciálními rovnicemi 2. řádu) na soustavu diferenciálních rovnic prvního řádu s dvojnásobným počtem rovnic, tzv. *Hamiltonovy kanonické rovnice*, jejichž význam daleko přesahuje rámec klasické mechaniky. V Hamiltonově práci pokračoval Carl Jacobi⁴⁶, který rozvinul transformační teorii kanonických rovnic a odstranil řadu obtíží spojených s jejich integrací.

Avšak asi tu nejdůležitější roli v rozvoji variačního počtu sehrála *úloha o brachistochroně*⁴⁷. V roce 1696 se Johann Bernoulli⁴⁸ v červnovém vydání časopisu Acta Eruditorum snažil ukázat, že právě vznikající diferenciální počet je tím nutným a postačujícím nástrojem k vyplnění mezer v klasické geometrii, což na závěr doplnil neodmítnutelnou výzvou, viz [1, str. 269],

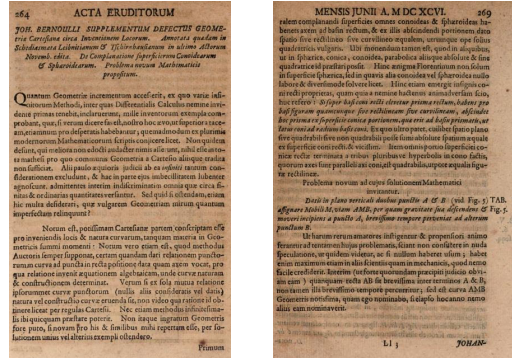
⁴⁷ Tento název pochází z řečtiny, kde *brachistos* je superlativ od *brachys*=krátký a *chronos*=čas. Leibniz navrhol označení *Tachyoptotam* z řeckého *tachy-* *stos*=nejrychlejší a *piptein*=padat. Tento návrh ale nebyl vyslyšen (naštěstí).



Johann Bernoulli (6. srpna 1667 – 1. ledna 1748) byl švýcarský matematik, fyzik a lékař. Je jedním z prvních členů matematické větve rodu Bernoulliů, která výrazně ovlivnila rozvoj matematiky na přelomu 17. a 18. století. Byl učitelem Leonharda Eulera, jehož matematický rozvoj (zejména v mládí) byl výrazně ovlivněn několika členů rodiny Bernoulliů. Při studiu medicíny na univerzitě se jako jeden z prvních naučil ovládat Leibnizův kalkulus. V roce 1691 odešel do Paříže, kde se seznámil l'Hospitalem, se kterým následně uzavřel smlouvu o tom, že jej za určitý roční honorář bude učit novým Leibnizovým metodám. Současně mu ale touto dohodou vzniklo právo užívat Bernoulliho objevy. „Své“ poznatky pak l'Hospital publikoval v roce 1696 v první učebnici infinitesimálního počtu *Analyse des Infinites Petits pour l'Intelligence des Lignes Courbes*, ve které se objevilo i známé pravidlo pro výpočet limity podílu, které dnes nese l'Hospitalovo jméno. Proti tomu se samozřejmě Johann Bernoulli ohradil, neboť jeho jméno nebylo nikdy uvedeno. Veřejně své námítky prezentoval zejména až po l'Hospitalově smrti v roce 1704. Nicméně důkazů pro své tvrzení příliš neměl, a tak pravda vyšla najevo až v roce 1922.

Problema Novum ad cujus Solutionem Mathematici invitantur.

*Datis in plano verticali duobus punctis A & B assignare Mobili M, viam AMB, per quam gravitate sua descendens & moveri incipiens a puncto A, brevissimo tempore perveniat ad alterum punctum B.*⁴⁹



Obrázek 1.1: Úvodní strana Bernoulliho článku (vlevo) a úloha o brachistochroně (vpravo).

Zároveň aby Johann Bernoulli přitáhl větší pozornost k této úloze, ihned vzápětí zdůrazňuje, že řešení úlohy není úplně zřejmé a že rozhodně hledání *křivky nejkratšího času* není totéž jako hledání nejkratší spojnice bodů A a B:

*Ut harum rerum amatores instigentur & propensiori animo ferantur ad tentamen hujus problematis, sciendum non consistere in nuda speculatione, ut quidem videtur, ac si nullum haberet usum; habet enim maximum etiam in aliis scientiis quam in mechanicis, quod nemo facile crederit. Interim (ut forte quorundam praecipiti iudicio obviam eam) quanquam recta AB sit brevissima inter terminos A & B, non tamen illa brevissimo tempore percurritur; sed est curva AMB Geometris notissima, quam ego nominabo, si elapso hoc anno nemo alius eam nominaverit.*⁵⁰

Johann Bernoulli také stanovil lhůtu na řešení do 31. prosince 1696, což se ale ukázalo být nedostatečné (avšak z důvodů spíše nematematických). Proto byla tato úloha krátce zopakována v lednovém vydání časopisu *Acta Eruditorum*, viz [2, str. 95–96], a další podrobnosti se můžeme dozvědět z Bernoulliho veřejného prohlášení v Groningenu z ledna 1697, jehož pojetí jakoby nás vracelo zpět do časů slavných matematických soubojů, který svedli např. del Fiore a Tartaglia:

⁴⁹ *Nová úloha, k jejímuž řešení se vyzývají matematikové.*

Nechť jsou dány dva body A a B ve svislé rovině. Určete křivku vyznačenou hmotným bodem, který startuje v bodě A a pouhým působením gravitace dosáhne bodu B v nejkratším možném čase.

(New Problem Which Mathematicians Are Invited.)

If two points A and B are given in a vertical plane, to assign to a mobile particle M the path AMB along which, descending under its own weight, it passes from the point A to the point B in the briefest time.)

⁵⁰ *Pro povzbuzení touhy k ujmnutí se řešení tohoto problému u těch, kteří takové úlohy milují, je možné zdůraznit, že předložený problém se nesešává, jak by se mohlo zdát, z pouhých spekulací, které tudíž nemají žádné využití. Naopak, jak bychom mohli snadno uvěřit, má velký užitek v různých vědních disciplínách jako v mechanika. Mezitím (abychom předešli ukvapenému soudu) [můžeme poznamenat] že ačkoli úsečka AB je vskutku nejkratší spojnicí bodů A a B , rozhodně není křivkou s nejkratším časem pohybu. Avšak křivka AMB , jejíž jméno uvedu já, pokud ji nikdo neobjeví před koncem tohoto roku, je velmi dobře známa geometrům.*

(To arouse in lovers of such things the desire to undertake the solution of this problem, it may be pointed out that the question proposed does not, as might appear, consist of mere speculation having therefore no use. On the contrary, as one would readily believe, it has great usefulness in other branches of science such as mechanics. Meanwhile (to forestall hasty judgment) [it may be remarked that] although the straight line AB is indeed the shortest between the points A and B it nevertheless is not the path traversed in the shortest time. However the curve AMB whose name I shall give if no one else has discovered it before the end of this year, is one well known to geometers.)

Jean Bernoulli public professor of mathematics pays his respects to the most acute mathematicians of the entire world.

Since it is known with certainty that there is scarcely anything which more greatly excites noble and ingenious spirits to labors which lead to the increase of knowledge than to propose difficult and at the same time useful problems through the solution of which, as by no other means, they may attain to fame and build for themselves eternal monuments among posterity; so I should expect to deserve the thanks of the mathematical world if, imitating the example of such men as Mersenne, Pascal, Fermat, . . . and others who have done the same before me, I should bring before the leading analysts of this age some problem upon which as upon a touchstone they could test their methods, exert their powers, and, in case they brought anything to light, could communicate with us in order that everyone might publicly receive his deserved praise from us. The fact is that a half a year ago in the June number of the Leipzig Acts [i.e., Acta Eruditorum] proposed such a problem whose usefulness linked with beauty will be seen by all who successfully apply themselves to it. . . Only the celebrated Leibniz who is so justly famed in the higher geometry has written me that he has by good fortune solved this, as he himself expresses it, very beautiful and hitherto unheard of problem; and he has courteously asked me to extend the time limit to next Easter in order that in the interim the problem might be made public in France and Italy and that no one might have cause to complain of the shortness of time allotted.⁵¹

viz [67, str. 646–647]. Poté Bernoulli poslal kopii tohoto problému Newtonovi, aby měl jistotu, že o něm ví. Zde je potřeba samozřejmě připomenout vleklý spor mezi Leibnizem a Newtonem ohledně prvenství v základech infinitesimálního počtu, do kterého vstoupil také Bernoulli, neboť v průběhu roku 1696 vyjádřil své přesvědčení, že Newtonova metoda publikovaná ve Wallisově *Opera mathematica* byla „štípnuta“ z Leibnizova článku. Bernoulli i Leibniz tak očividně interpretovali ticho od června do prosince jako demonstraci, že tento problém byl pro Newtona záhadou. Prodloužením lhůty měli asi v úmyslu veřejně demonstrovat svou nadřazenost. Ať už si Bernoulli a Leibniz mysleli cokoli, Newton tuto výzvu považoval za namířenou osobně proti němu. No, kdyby to nebylo úplně zřejmé z přímé korespondence, zakomponoval Bernoulli do svého vyjádření i tato slova:

Qpod si vero festum paschatis praeterierit nemine deprehensio qui quaesitum nostrum solverit, nos quae ipsi invenimus publico non invidemus. Incomparabili enim Leibnizii solutiones tura suam tura nostram ipsi jam pridem commissam protinus, ut spero, in lucem emittet, quas si Geometrae ex penitiori quodam fonte petitas perspexerint, nulli dubitamus quin angustos vulgaris Geometriae limites agnoscant, nostraque proin inventa tanto pluris faciant, quanto pauciores eximiam nostram quaestionem solutori extiterint etiam inter illos ipsos qui per singulares quas tantopere commendant methodos, interioris Geometriae latibula non solum intime penetrasse, sed etiam ejus pomperia Theorematis suis aureis, nemini ut putabant cognitis, ab aliis tamen jam longe prius editis, mirum in modum extendisse gloriantur.⁵²

viz [65, str. 220–229]. Zjevně se to Newtonovi vrylo velmi hluboko do paměti, neboť o dva roky později se při jedné z debat s prvním královským astronomem Johnem Flamsteedem na něj ohradil se slovy

⁵¹ Já, Johann Bernoulli, oslovuji nejbystřejší matematiky na světě.

Jelikož je jistě známo, že není nic přitažlivějšího pro inteligentní lidi než čestný a náročný problém, jehož jakékoli řešení přinese slávu a zůstane trvalým památkem. Následující řady příkladů Mersenna, Pascala, Fermata, ... doufám, že získám vděčnost celé vědecké komunity tím, že před nejlepší matematiky naší doby postavím problém, který otestuje jejich metody a sílu jejich intelektu. Jestliže mi kdokoli sdělí řešení navrhovaného problému, veřejně prohlásím, že je hoděn uznání. Ve skutečnosti jsem takový problém, jehož užitečnost spojenou s krásou uvidí každý, kdo se zdárně odhodlá k jeho řešení, zveřejnil před půl rokem v červnovém vydání Acta Eruditorum... Pouze slavný Leibniz, který je po právu věhlasný vyšší geometrii, mi napsal, že se mu poštěstilo nalézt řešení tohoto, jak se sám vyjádřil, překrásného a doposud nevidaného problému. Zdvorně mě požádal o prodloužení lhůty pro řešení do následujících Velikonoc, aby se mezitím problém mohl rozšířit do Francie a Itálie, takže by nikdo neměl důvod si stěžovat na nedostatečný čas určený k řešení.

*I do not love... to be dunned & teezed by foreigners about Mathematical things...*⁵³

Newton výzvu přijal v roce 1697, když si na dopis poznamenal datum doručení: 29. ledna 1697. Už následující den odeslal řešení Charlesi Montagueovi, tehdejšímu prezidentu Královské společnosti, a bylo anonymně zveřejněno ještě v lednovém vydání *Philosophical Transactions of the Royal Society*. Bernoullimu samozřejmě nedalo příliš práce určit „tajného“ autora se slovy *tanquam ex ungue leonem*⁵⁴. Newtonův pocit vítězství byl tak velký, že se tento příběh dostal prostřednictvím jeho neteře Catherine dostal až do Conduittovi sbírky anekdot:

*When the problem in 1697 was sent by Bernoulli – Sir I. N. was in the midst of the hurry of the great recoinage did not come home till four from the Tower very much tired, but did not sleep till he had solved it which was by 4 in the morning.*⁵⁵

viz [77, str. 582]. Avšak nezanedbatelnou kaňkou na Newtonově úspěchu je fakt, že v článku chybí podrobnější zdůvodnění získaného řešení.

Měli bychom zmínit, že Johann Bernoulli rozhodně není prvním matematikem, který se zabýval takovýmto problémem. Již v roce 1638 se mu věnoval také Galileo Galilei ve své slavné práci *Matematické rozpravy a pokusy* (*Discourses and Mathematical Demonstrations Relating To Two New Sciences*). Nejdříve požadoval bod B pouze na přímkce a hledal vhodnou úsečku. Správným řešením je úsečka svírající s přímkou b úhel 45° . Galileo poté také určil (správně), že kratší čas bude dosažen při pohybu po dvou úsečkách AC , BC , kde C je bod ležící na oblouku kružnici procházející body A , B . Ačkoli Galileovy úvahy byly doposud správné, vyvodil odtud, že nejrychlejší bude pohyb po kružnici. Jenže ona existuje ještě rychlejší dráha. Leibniz údajně předpověděl (téměř správně), že existuje pouze pět geometrií, kteří dokáží danou úlohu vyřešit. Měl na mysli Jacoba Bernoulliho, l'Hospitala, Huygense (kdyby žil), Huddeho (kdyby toho nezanedbal) a Newtona (kdyby se takové výzvy chopil). Jednotlivá řešení pak byla zveřejněna v květnovém vydání časopisu Acta Eruditorum z roku 1697 a jejich autory byli

- (0) Johann Bernoulli (on sám řešení znal již v době publikování problému, pouze vyzýval ostatní k jeho řešení), viz [2, str. 206–211];
- (1) Jacob Bernoulli, viz [2, str. 211–214];
- (2) Isaac Newton, viz [2, str. 223], což je přetisk výše zmíněného „anonymního“ článku;
- (3) Guillaume l'Hospital, viz [2, str. 217–218];
- (4) Gottfried Leibniz, viz [2, str. 201–205];
- (5) Ehrenfried Walther von Tschirnhaus (na něj se velmi často zapomíná!), viz [2, str. 220–223];

Hledanou křivkou byla jedna z těch, které již tehdy byly *velmi dobře známy všem geometriím* – otočená *cykloida* (jméno jí dal v roce 1599 Galileo a objevuje⁵⁵ se dokonce v prvním díle Gulliverových cest od anglického spisovatele Jonathana Swifta, který vyšel v roce 1726). Tato křivka byla studována již Pascalem a Huygensem, avšak nikdo nic netušil, že je to křivka nejrychlejšího spádu. Johann Bernoulli toto okomentoval s nadhledem sobě vlastním

*... you will be petrified with astonishment when I say that precisely this cycloid... of Huygens is our required brachistochrone.*⁵⁶

viz [32, str. 201]. Tato úloha vzbudila velký zájem o podobné problémy, kdy se nehledaly extrémní funkce, ale samotná extrémální funkce (pomocí analytických metod – nikoli geometricky). Do-

⁵² *Pokud ovšem ani do velikonočních oslav se nenajde nikdo, kdo by byl schopen náš vznešený problém vyřešit, nebudeme se zdráhat zveřejnit to, co jsme sami objevili. Doufám, že jedinečný Leibniz neprodleně zveřejní své i mé řešení, které jsem mu před časem svěřil. Pokud geometři pečlivě prozkoumají tato řešení získaná tak, jak jsou, z toho, co můžeme nazývat hlubokou studnicí (myšlenek), nepochybujeme o tom, že rozpoznají úzké hranice běžné geometrie a ocení naše objevy o to více, že je jen několik těch, kteří jsou schopni vyřešit naše skvělé problémy, ba dokonce, malý počet mezi těmi nejlepšími matematicky, kteří se chlubí, že svými pozoruhodnými metodami, za než jsou chvályhodní, nejenže pronikli hluboko do tajemství esoterické geometrie, ale úžasně rozšířili její hranice díky zlatým větám, které (jak se domnívají) nebyly známy nikomu jinému, avšak které byly ve skutečnosti publikovány již dávno předtím jinými.*

(If however the festival of Easter shall have passed without our having found any who are capable of solving our noble problem, we shall not grudgingly refuse to the public what we have ourselves discovered. For the peerless Leibniz will forthwith, I hope, publish the solutions, his own and mine, which I entrusted to him some time ago. If geometers carefully examine these solutions, drawn as they are from what may be called a deeper well (of thought) we are in no doubt but that they will recognise the narrow limits of the common geometry, and will value our discoveries so much the more as there are fewer who are likely to solve our excellent problems, aye, the small number even among the very mathematicians who boast that by the remarkable methods they so greatly commend, they have not only penetrated deeply the secret places of esoteric geometry but have also wonderfully extended its bounds by means of the golden theorems which (they thought) were known to no one, but which in fact had long previously been published by others.)

⁵³ *Nemám rád... když jsem uháněn a popichován cizinci ohledně matematických záležitostí...*

⁵⁴ *Podle drápů se pozná lev.*

(We may judge the lion from its claw).

dejme ještě, že cykloida je také současně *tautochronou*, tj. ať kuličku vypustíme z jakéhokoli bodu na brachistochroně mezi *A* a *B*, pak dorazí do nejnižšího bodu na dráze vždy za stejný čas.

Navzdory nezměrnému historickému významu úlohy o brachistochroně je nutné zdůraznit, že to rozhodně nebyla jedna z prvních úloh tohoto typu. Již ve starověku se řešili tzv. *isoperimetrické problémy*, jejichž cílem bylo mezi všemi uzavřenými rovinnými křivkami nalézt takovou, která ohraničuje plochu s maximálním obsahem (např. máme-li 10 metrů drátu a chceme s jeho pomocí vymežit největší výběh pro králíky). Nejznámější takovou úlohou je *Problém královny Didó*, díky kterému je královna Didó nedílnou součástí úvodních kapitol drtivé většiny knih věnovaných variačnímu počtu. O této úloze víme díky eposu Aeneis od římského básníka Publius Vergilius Maro. Tento epos vznikl v letech 29–19 př.n.l. a je psán hexametrem (tj. nerýmovaným veršem):

Punské říše vidíš, Tyrského Agénora město! toť meze Libyčanů, lidu nevládného ve válce. Z města Tyru přišla království Didó to řídí, bratrem zahnána. Dlouhá je to křívda a dlouhé okliky; já tedy jen co je hlavnějšího, podotknu. Choť jí byl Sycheys, Foeničan nejbohatější statkem a od nebohé milován ze celého to srdce panny, jemuž ji otec vydal a v manželstvo prvotní zasnoubil; bratr než se ujal Tyrského vladařství Pygmalion, zločinec horší nade všechny zločince. Rozdvojilo vzteklé oba záští, tak že ve chrámě bezbožný Sychéa tyran, zaslepen zlata vášní, tajně a náhle zabil dýkou, ani lásky neváživ sestřiny; jí pak dlouho tajil to zločinstvo a marnou přetvářen zlostník nadějí manželku klamával. Pohřebu však zbavený ve snách jí sám se ukázal manžel, a vybledlá otvíraje ústa podivně, ukrutný oltář, ocelí proťatá pak i nádra odhalil a projevil vše tajemství zlosti domác. K neprodlenému posléz rady z otčiny odchodu dává a v prospěch té pouti pravetché vymkne ze země poklady, neznámé nikomu zlata břímě a stříbra. Pohnula tím chystá i hledá druhy k outěku Didó. Scházeli se, kdo nenáviděli zlostného tyranna neb kdo se báli jeho; zmocnivše se právě korábů ustrojených, naloží je zlatém a statky lakomce Pygmalióna mořem unesou; činu ženština vůdcem. Do kraje přišli polom, kdež náramně ted' uhlédáš ohrady a pnoucí se nové Kartháginu bašty. Koupili místo také, a k paměti nazvali Byrsou, tak veliké, co koží volovou zahrnouti se mohlo.

„Modernější“ (a lépe srozumitelná) verze této legendy by mohla znít takto:

Po tom, co manžela a současně strýce fénické princezny Ellisy, později nazývané královnou Didó, zavraždil její bratr Pygmalion, musela sama uniknout a hledat bezpečí. S lidmi, kteří ji byli ochotni následovat, přistála se svou lodí na severoafrickém pobřeží — na území, které se nacházelo v dnešní blízkosti Tunisu – a požádala Iarbase, tehdejšího krále Numidie, o azyl. Iarbas rozhodl,

že Ellise daruje území – jeho nabídka ale nebyla příliš velkorysá, neboť své území nové území musela ohraničit jedinou býčí kůží. Doslova jí řekl „Užívej pozemek tak velký, jaký vymeží volská kůže“.

Ellisa dala rozřezat kůži na velmi tenké pásky,

⁵⁵ Když byl ten problém Bernoullim odeslán v roce 1697, Sir I. N. byl zrovna uprostřed shonu kvůli velké měnové reformě a vrátil se domů z londýnského Toweru [kde byla Královská mincovna] velmi unavený až po 16. hodině. Ovšem nešel spát, dokud problém nevyřešil, což se mu povedlo kolem čtvrté hodiny ranní.

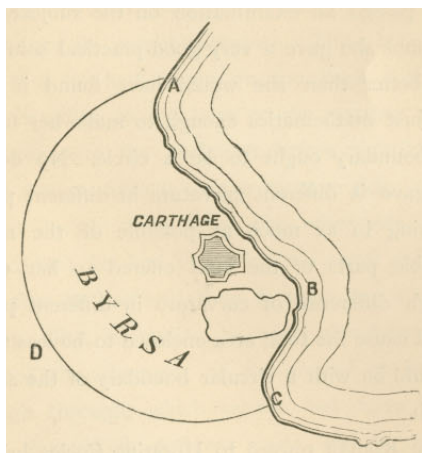
⁵⁵ Pokud jsem dovedl posouditi, dával mi král různé otázky a já jsem jej zase oslovoval všemi jazyky, jež jsem znal. Když se ukázalo, že nerozumím ani já jim, ani oni mně, zavedli mě na králův rozkaz do jedné komnaty v paláci (tento panovník totiž vynikal pohostinstvím nad všechny své předchůdce), kde mi byli dáni k obsluze dva sluhové. Přinesli mi oběd, a čtyři šlechtici, jež jsem předtím viděl v těsné blízkosti krále, mi prokázali tu čest, že obědvali se mnou. Měli jsme dva chody, každý o třech jídlech. První chod záležel ve skopovém ramínku vykrojeném do rovnoramenného trojúhelníku, z kusu hovězího, připraveného v podobě kosodélníku, a z pudíngu v podobě cykloidy.

⁵⁶ ... budete zkameněli úžasem, když řeknu, že právě tato Huygensova cykloida... je hledanou brachistochronou.



Obrázek 1.2: *Didó a býčí kůže* od italského malíře Gregoria Lazzarinioho (1655–1730).

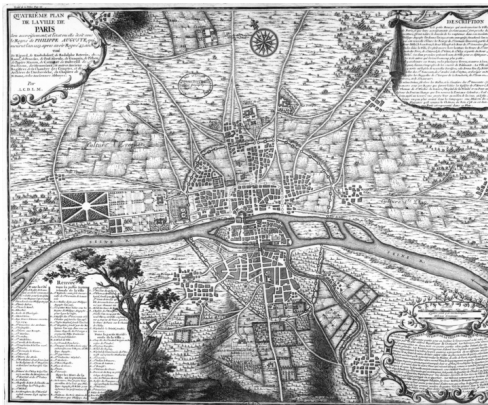
kteří dohromady utvořili pruh dlouhý 4 km. Tímto pruhem pak nechala vymezit území o tvaru půlkruhu uzavřeného z jedné strany pobřežím. Toto území ohraničené hradbami se nazývalo *Byrsa* (což v překladu znamená „býčí kůže“) a tvořilo centrum Kartága, které vzniklo 72 let před Římem, tj. 825 př..l. (někdy se uvádí rok 814 př.n.l.). Didó tak vyřešila isoperimetrický problém, neboť se jí podařilo pro své lidi získat největší možné území.



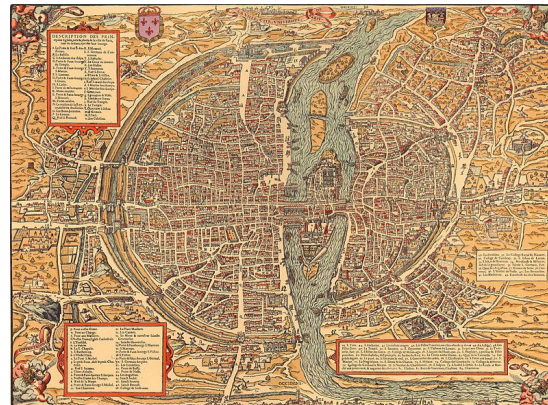
Obrázek 1.3: Situační obrázek městy Byrsa a Tuniského zálivu ve Středozezemním moři. Obrázek pochází z přednášky William Thomsona⁵⁷ s názvem *Isoperimetrické problémy*, kterou přednesl 12. května 1893 na konferenci Královské společnosti v Londýně, viz [71] a také [4]. Byl to asi právě tento příspěvek lorda Kelvina, který umožnil královně Didó vstup do historie matematiky a učinil z ní nedílnou součást mnoha knih.

⁵⁷ William Thomson (26. června 1824 – 17. prosince 1907) je známý spíše pod svým šlechtickým jménem lord Kelvin (of Largs). Byl irsko-skotský matematický fyzik a vynálezce. Je považován za dovršitele mechaniky ve fyzice.

Po zániku antické civilizace nastává dlouhé období vědecké stagnace a úpadku, nicméně i středověcí stavitelé museli vyřešit analogický problém vzhledem k náročnosti výstavby hradebního opevnění. Což se jim mnohdy povedlo, jak můžeme vidět na historických mapách Paříže aj. níže.



Obrázek 1.4.a: Paříž (rok 1223).



Obrázek 1.4.b: Paříž (rok 1575).



Obrázek 1.5.a: Milán (rok 1572).



Obrázek 1.5.b: Frankfurt nad Mohanem (rok 1572).

Řešení isoperimetrické úlohy bylo známo již Aristotelovy (cca 384–322 př.n.l.), který věděl, že mezi rovinnými obrazci o stejném obvodu má největší obsah kruh a mezi tělesy stejného povrchu má největší objem koule. Ve starověku lze nalézt i další podobné úlohy s geometrickým základem, např.

- (i) *Apollonius* (3. století př.n.l.): jak vést z daného bodu k dané kuželosečce nejkratší a nejdelší úsečku tak, aby jeden její krajní bod byl tím zadaným a druhý krajní bod ležel na dané kuželosečce?
- (ii) *Euklides z Alexandrie* (cca 340–270 př.n.l.): jak vepsat do daného trojúhelníku rovnoběžník maximálního obsahu?
- (iii) *Archimédes ze Syrakus* (cca 278–212 př.n.l.): ze všech kulových úsečí stejného povrchu určit tu s maximálním objemem.

Později byly isoperimetrické úlohy různě modifikovány a dnes se užívají pro daleko širší třídu extrémálních úloh, ve kterých se hledají extrémy integrálních funkcionalů s omezeními v integrálním tvaru, tj. nalézt (alespoň diferencovatelnou) funkci (křivku) $y = y(x)$ splňující

$$y(a) = A \quad \& \quad y(b) = B \quad \& \quad \mathcal{K}[y] := \int_a^b G(x, y, y') dx = \ell,$$

pro kterou nabývá funkcional

$$\mathcal{J}[y] = \int_a^b f(x, y, y') dx$$

extremální hodnoty pro pevně dané $\ell \in \mathbb{R}$ a funkce $F, G: \mathbb{R}^3 \rightarrow \mathbb{R}$. V případě královny Didó máme při pevných koncích umístěných do bodů $[-1, 0]$ a $[1, 0]$ by šlo tedy o úlohu

$$\int_{-1}^1 y(x) dx \rightarrow \min$$

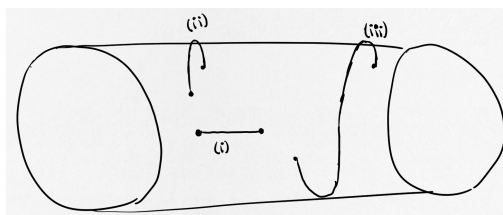
za podmínek

$$y(-1) = 0 \quad \& \quad y(1) = 0 \quad \& \quad \int_{-1}^1 \sqrt{1 + [y'(x)]^2} dx = \ell.$$

Řešením je oblouk jisté kružnice, přičemž pro $\ell = \pi$ máme půlkružnici se středem v počátku.

Krátce pro vyřešení úlohy o brachistochroně byla řešena řada podobných úloh. Např. *úloha o nejkratších čarách* (neboli *geodetikách*), které se věnoval Euler ve své první publikaci [34]. Podnět k tomuto tématu dostal v roce 1728 od svého školitele Johanna Bernoulliho (ten ji znal již od roku 1698, jak ukazují jeho lekce o integrálním počtu vydané tiskem roku 1742). Mějme dva body A, B na ploše $S \subseteq \mathbb{R}^3$. Chceme najít nejkratší křivku, která spojuje dané body a současně leží na ploše S . Např. uvažme rotační válec. Pak máme tři možné geodetiky (viz Obrázek 1.6 níže):

- (i) přímku rovnoběžnou s osou x (pokud body leží na jedné přímce vzniklé řezem rovinou procházející osou válce);
- (ii) kružnice kolmou na osu válce (pokud body leží na kružnici vzniklé řezem rovinou kolmou na osu válce);
- (iii) šroubovice.



Obrázek 1.6: Geodetiky na válci.

V letectví, aby se minimalizovala dráha letu, letadla „využívají“ právě geodetiky. V ideálním případě spíše koridory, neboť v každém bodě je jiný kurz – na mapě se zdá, že dráha není nejkratší (není to přímka), ale nežijeme na zeměploše. Pro jednoduchost uvažme, že plocha S je dána jako graf funkce

$$S: z = F(x, y)$$

(to válce samozřejmě nespĺňuje, ale můžeme uvažovat válcové souřadnice; podobně pro kouli a sférické souřadnice; v diferenciální geometrii to lze rozšířit na libovolnou parametrickou plochu). Chtěli bychom najít geodetiku $C \subset S$, která spojuje body

$$A = [a, \alpha, F(a, \alpha)] \quad \& \quad B = [b, \beta, F(b, \beta)].$$

Připusťme, že C může být parametrizována pomocí x jako

$$y = u(x) \quad \& \quad z = v(x) = F(x, u(x)),$$

příčemž poslední rovnice zaručuje to, že C leží na ploše S . Toto zjednodušení ovšem vyžaduje, aby $a \neq b$. Délka křivky je pak dána standardně, takže chceme minimalizovat funkcionál

$$\ell[u] = \int_a^b \sqrt{1 + [u'(x)]^2 + [v'(x)]^2} dx = \int_a^b \sqrt{1 + [u'(x)]^2 + \left[\frac{\partial F}{\partial x}(x, u(x)) + \frac{\partial F}{\partial u}(x, u(x)) \frac{du}{dx} \right]^2} dx$$

za požadavku $u(a) = \alpha$ a $u(b) = \beta$. Např. pro geodetiky na paraboloidu

$$z = \frac{x^2}{2} + \frac{y^2}{2}$$

chceme

$$\int_a^b \sqrt{1 + [u'(x)]^2 + [x + u(x)u'(x)]^2} \rightarrow \min.$$

Euler své výsledky v této oblasti shrnul v práci [35]. V ní je odvozena obecná metoda (tzv. *Eulerova přímá metoda konečných diferencí*) pro řešení takovýchto problémů a následně ilustrována na stovce z nich. My jsme si tuto metodu ukázali bez větších detailů již v Příkladě 1.2. Euler tuto svoji metodu opustil poté, co 12. srpna 1755 obdržel od Lagrange dopis s mnohem elegantnější (*přímou*) *metodou variací*, kterou objevil při řešení *problému tautochrony*⁵⁸. Tato metoda uchvátla Eulera natolik, že už v textu [36] z roku 1756 (publikován byl ale až roce 1766) označil na Lagrangeovu počest tuto oblast názvem *variální počet*.

Zobecněním problému geodetik jsou tzv. *minimální plochy*. Nejjednodušší interpretace je, že máme křivku C v \mathbb{R}^3 a cílem je najít plochu s minimálním povrchem mezi všemi plochami, které mají za hranici křivku C . Např. máme-li C jako uzavřenou rovinnou křivku (např. kružnici), pak minimální plocha je pouze oblast ohraničená touto křivkou. Ovšem pokud je křivka zakřivená i ve třetím rozměru, pak podoba minimální plochy již není tak zřejmá. Fyzikální interpretace: pokud ohneme dva dráty do tvaru křivky C a ponoříme do mýdlové vody, pak povrchové napětí výsledné plochy, kterou obdržíme odtazením drátů jako mýdlovou plochu, způsobí to, že máme minimální plochu. Mají-li oba dráty tvar kružnice, pak řešením je tzv. *katenuoid*, což je rotační plocha vzniklá rotací *řetězovky*⁵⁹. Řetězovka je další zajímavou křivkou – je totiž řešením dalšího isoperimetrického problému hledajícího křivku (v ideálním případě homogenní dokonale ohebné nepružné vlákno – řetěz) dané délky s minimální potenciální energií ve svislé rovině tíhového pole Země (má tedy nejnižší položené těžiště). Její tvar zaujímají závěsné mosty (např. lásky přes Svatku v Brně–Komíně) či dráty elektrického vedení. Objevuje se také v architektuře – např. zastřešení Pavilonu A na BVV, Gateway Arch v St. Louis či Casa Milà od Antonia Gaudího v Barceloně. Mýdlové bubliny a plochy byly po staletí zdrojem velkého zájmu fyzikálního, estetického i matematického, viz tzv. *Plateauův problém*.

Ovšem parametrická optimalizace rozhodně takovou pozornost neupoutala, ačkoli takových úloh byla v historii řešeno velké množství⁶⁰. Vlastně jeden takový problém dal vzniknout celé naší civilizaci, když Adam a Eva našli optimální řešení *jablíčkového dilematu*, které mělo pro hada za následek věčné plazení po břiše a pro lidstvo znamenalo vyhnání z ráje, viz [6]. V Bibli najdeme i možné řešení plánovacího problému týkajícího se *egyptské sedmiletky* či *tenkého programování*:

⁵⁸ Jde o nalezení křivky, podél níž dorazí hmotný bod do cíle vždy za stejný čas bez ohledu na výchozí bod. Už jsme se zmínili, že řešením i tohoto problému je obrácená cykloida. Toto ukázal již Huygens v roce 1659.

⁵⁹ Anglicko-matematický vtíp: *How do you make cat-enoid? By pulling its tail.* Tohle je bohužel do češtiny asi nepřenositelné...

⁶⁰ Velkou inspirací pro některé z následujících historických informací byla kniha [43]

25 Josef faraónovi odpověděl: „Faraónův sen je jeden a týž. Bůh faraónovi oznámil, co učiní.

26 Sedm pěkných krav, to je sedm let. Také sedm pěkných klasů je sedm let. Je to jeden sen.

27 Sedm vychrtlých a šeredných krav, vystupujících za nimi, je sedm let, stejně jako sedm prázdných a východním větrem sežehlych klasů; to bude sedm let hladu.

28 Když jsem faraónovi řekl: Bůh faraónovi ukázal, co učiní, mínil jsem toto: 29 Přichází sedm let veliké hojnosti v celé egyptské zemi.

30 Po nich však nastane sedm let hladu a všechna hojnost v egyptské zemi bude zapomenuta. Hlad zemi úplně zničí.

31 V zemi nebude po hojnosti ani potuchy pro hlad, který potom nastane, neboť bude velmi krutý.

32 Dvakrát byl sen faraónovi opakován proto, že slovo od Boha je nezvratné a Bůh to brzy vykoná.

33 Ať se tedy farao nyní poohlédne po zkušeném a moudrém muži a dosadí ho za správce egyptské země.

34 Nechť farao ustanoví v zemi dohlážitěle a po sedm let hojnosti nechť vybírá pětinu výnosu egyptské země.

35 Ať po dobu příštích sedmi úrodných let shromažďují všechnu potravu a ve městech ať uskladňují pod faraónovu moc obilí a hlídají je.

36 Tato potrava zabezpečí zemi na sedm let hladu, která přijdou na egyptskou zemi. A země nezajde hladem.”

37 Tato řeč se faraónovi i všem jeho služebníkům zalíbila.

viz [7]. V méně vzdálené historii zase najdeme úlohy s geometrickým základem a s geometrickým řešením. Znamé jsou (k již dříve uvedeným třem úlohám) např.

- (i) *Heronova úloha* z 1. stol. př.n.l.: jsou-li dány dva body A a B na téže straně přímky ℓ , nalezněte bod C na této přímce tak, aby součet vzdáleností z A do C a z C do B byla minimální (populární je její formulace pomocí *kosa a žízal*);
- (ii) *Tartagliova úloha*: číslo 8 rozdělte na dvě části tak, aby jejich součin vynásobený jejich rozdílem byl maximální;
- (iii) *Toricelliho–Fermatův problém*: v trojúhelníku ABC nalezněte bod X , pro který bude součet vzdáleností $|AX| + |BX| + |CX|$ minimální;
- (iv) *Keplerova úloha*: do dané koule vložte válec maximálního objemu.

Ve středověku je možné odhalit úlohy parametrické optimalizace zejména v souvislosti se studiem vítězných strategií v různých hrách, tj. s jistým prvkem nejistoty, což vedlo až ke vzniku rigorózní teorie pravděpodobnosti. Za zmínku zde stojí např. Cardanova kniha *Liber de Ludo Aleae* [Kniha o hazardních hrách (The Book on Games of Chances)] z roku 1564, která ale byla publikována až v roce 1663, nebo práce Blaise Pascala⁶¹ a Fermata. Všichni se věnovali i tzv. *Úloze o rozdělení sázky* (či *Problém s body*), která pochází nejspíše ze 14.–15. století a v tištěné podobě se poprvé objevuje v knize *Summa de arithmetica, geometrica, proportioni et proportionalita* [Přehled aritmetiky, geometrie, proporcí a proporcionality (Summary of arithmetic, geometry, proportions and proportionality)] od Lucy Pacioliho⁶² a z roku 1494. V této úloze jde o dva hráče A a B hrající několik kol nějaké spravedlivé hry, kteří se dohodli, že celou sázku získá ten, jenž

⁶¹ Blaise Pascal (19. června 1623 – 19. srpna 1662) byl francouzský matematik, fyzik, vynálezce, spisovatel a katolický teolog. V 19 letech začal pracovat na výrobě mechanického kalkulátoru. Po třech letech, ve kterých zkonstruoval 50 prototypů, úspěšně zkonstruoval (tzv. Pascalův kalkulátor či *Pascalinu* a v následujících 10 letech jich nechal vyrobit 20 kusů. Díky tomu se stal jedním z prvních dvou konstruktérů mechanického kalkulátoru. V 16 letech sepsal krátké pojednání o *mystickém hexagramu*, které nazval *Essai pour les coniques* [Pojednání o kuželosečkách (Essay on Conics)], které jakožto svoji první skutečnou matematickou práci odeslal Marinu Mersenneovi do Paříže. Jejím obsahem je dodnes známá Pascalova věta týkající se šestiúhelníku vepsaného do kružnice. Tuto práci ocenila i Pařížská královská akademie a Descartes je pokládal za práci jeho otce, když to komentoval slovy „Nezdá se mi divné, že předvedl důkazy o kuželosečkách mnohem vhodnější než ty antické, jenže v této souvislosti lze zmínit jiné otázky, které by sotva přišly na mysl šestnáctiletému dítěti“. Pascal byl také v určité době velmi vášnivým hráčem, což jej přivedlo k tomu, že společně s Fermatem položily základy teorie pravděpodobnosti (jím označované jako *matematická naděje*) a vytvořil pojem střední hodnota.

⁶² Luca Bartolomeo de Pacioli (někdy 1447 – 19. června 1517) byl italský matematik, františkánský mnich, spolupracovník Leonarda da Vinci a jeden ze zakladatelů účetnictví.

⁶³ Jinými slovy jde o tzv. *princip úměrnosti: při řešení nejisté úlohy v budoucnosti máme uvažovat pouze to, co se může stát, a nikoliv to, co se již stalo*. Na toto si vzpomeňte, až se dostaneme ke Kapitole ??

⁶⁴ Nicolaus Bernoulli (21. října 1687 – 29. listopadu 1759) byl švýcarský matematik. Doktorát získal ve 22 letech za práci o teorii pravděpodobnosti v právu. Je autorem věhlasného Petrohradského paradoxu.

⁶⁵ Pierre Rémond de Montmort (27. října 1678 – 7. října 1719), byl francouzský matematik. V roce 1715 byl zvolen členem Královské společnosti v Londýně a v roce 1716 také Francouzské akademie věd. Je znám především díky knize *Essay d'analyse sur les jeux de hazard* [Analýza hazardních her (Analysis of games of chance)] o pravděpodobnosti a hazardních hrách a je také autorem souloví *Pascalův trojúhelník*, pro který používal označení *Table de M. Pascal pour les combinaisons* [Tabulka pana Pascala pro kombinace (Table of Mr. Pascal for combinations)].

⁶⁶ Tak o tomhle pánovi nevíme asi vůbec nic. Vlastně bylo docela těžké jej vůbec identifikovat, viz [8, 9]

⁶⁷ Charles Babbage (26. prosince 1791 – 18. října 1871) byl anglický matematik, filosof, vynálezce a strojní inženýr. Bývá považován za „otce počítačů“, neboť jako první přišel s myšlenou programovatelných počítačů. Některé části z jeho nedokončených mechanických počítačů jsou vystaveny v Londýnském vědeckém muzeu.

⁶⁸ Johann Carl Friedrich Gauss (30. dubna 1777 – 23. února 1855) byl jeden z největších (nejen) německých matematiků všech dob. Výrazně ovlivnil zejména rozvoj geometrie, matematické analýzy, teorie čísel, astronomie, elektrostatiky, geodézie a optiky. Nicméně v oblasti našeho současného zájmu tak významnou roli nesehrál. Koluje také řada různě důvěryhodných historek o jeho genialitě již v brzkém věku.

⁶⁹ Adrien-Marie Legendre (18. září 1752 – 10. ledna 1833) byl francouzský matematik, který výrazně přispěl k rozvoji statistiky, teorie čísel, abstraktní algebry a matematické analýzy.

jako první bude mít N výher. Jenže vlivem vnějších okolností je nutné hru přerušit ve chvíli, kdy hráč A má pouze r výher a hráč B pouze s výher. Jak si mají hráči rozdělit sázku? Pacioli navrhol, aby si sázku rozdělili v poměru výher, tj. $r : s$. Takové řešení se nelíbilo např. Tartagliovi, který trefně namítal, že kdyby se to stalo hned po první hře, tak by si celou sázku vzal vítěz této hry. Tartaglia přišel s jistým návrhem řešení založeným na poměru čísla N a rozdílu $|r - s|$. Částečné řešení přinesl v roce 1539 i Cardano, který přišel s tím, že jde pouze o to, „co by se mohlo stát v budoucnosti, nikoli v minulosti“⁶³. Ale ani Cardanovo řešení nebylo správné. S tím přišli až Fermat s Pascalem ve vzájemné korespondenci z roku 1654.

Další zajímavou karetní hrou je *Le Her* („Jí“), ve které dva hráči A a B losují po jedné kartě z klasického balíčku 52 karet s hodnotami od 1 do 13 ve 4 barvách. Hráč A může přinutit hráče B , aby si karty vyměnili s výjimkou situace, kdy B má číslo 13. Pokud B není spokojen se svoji původní kartou nebo s tou, kterou dostal po výměně s A , může si z balíčku zbývajících 50 karet vylosovat jinou, ovšem pokud si vylosuje číslo 13 není výměna povolena. Poté si oba hráči porovnají karty a vyhrává ten, který má větší číslo, přičemž v případě shody vyhrává B . Tato hra se poprvé objevuje v dopise od Nicholase Bernoulliho⁶⁴ z 10. listopadu 1711 určeném Pierru de Montmortovi⁶⁵. Řešení tohoto problému se podařilo najít Francisí Waldegraveovi⁶⁶, který hledal strategii maximalizující pravděpodobnost výhry bez ohledu na protihráčovu taktiku. Tím se mu podařilo najít tzv. *minimax* řešení.

Pro některé z těchto úloh lze využít tvrzení, které jsme si uvedli v předchozí části, avšak v minulosti byly obvykle řešeny jinak a zcela individuálně – neexistovala žádná univerzální metoda. Jenže zmíněné nástroje se zdají být nedostatečné např. pro řešení jednoduchého *dopravního problému*. Podnikatel XY má v EU 3 výrobní závody s kapacitami 250, 500 a 700 ks denně. Současně je po Evropě 5 prodejních skladů, z nichž v každém lze prodat až 300 ks denně. Kolik kusů rozeslat do jednotlivých skladů tak, aby ve skladech bylo na začátku každého dne tolik kusů zboží, kolik tam lze prodat, a současně s tím byly spojeny minimální přepravní náklady? Toto vlastně úzce souvisí s výzkumem Charlese Babbageho⁶⁷ ohledně nákladů na přepravu a třídění poštovních zásilek, který vedl v roce 1840 ke zřízení *Penny Post* coby jedné ze součástí kompletní reformy Královské pošty. Nebo můžeme připomenout např. *metodu nejmenších čtverců*, ve které minimalizujeme součet kvadrátů jednotlivých odchylek. Tato metoda pochází z přelomu 18. a 19. století a její vznik je spojována především s Friedrichem Gaussem⁶⁸ a Adrienem-Marie Legendrem⁶⁹.

Překvapivě až do první poloviny 20. století nenalezneme příliš mnoho metod k řešení takovýchto problémů (tj. rozhodovacích či řídicích procesů). Zájem o tyto problémy stoupal zejména v ekonomii v souvislosti s rozvojem průmyslu a zefektivněním výroby, což pokračovalo i ve 30. letech, i když tentokrát ve spojení s příchodem *Nového údělu*⁷⁰ v USA a podobných plánů v ostatních částech světa, které měly za cíl překonání *Velké hospodářské krize*. Zlom v dosavadním *ad hoc* přístupu přinesla až druhá světová válka, která si žádala řešení i jiných úkolů, a to především těch vojenských. Během této války armádní vedení Velké Británie vyzývalo skupiny vědců z různých oblastí k využití jejich vědeckých znalostí v různých strategických a taktických válečných problémech. Díky tomu vznikla nová oblast matematiky nazývaná *operační výzkum*⁷¹, což je víceméně totéž jako „parametrická optimalizace“, ale toto označení odkazuje na skutečnost, že se jednalo především o výzkum ve vojenských operacích. Za skutečný počátek lze proto považovat rok 1936, kdy britský ministr letectví ustanovil v Suffolku tzv. *Bawdsey Research Station* v panství Bawdsey Manor, které dříve patřilo jednomu ze zakladatelů *National Telephone Company*. Úkolem této skupiny bylo hledání vojenského využití nově objevené radarové technologie, např. pro cílené zachytávání nepřátelských letadel. Prvním šéfem této skupiny byl Robert Watson-Watt,

⁷⁰ *New Deal*, v letech 1933–1937 díky prezidentu Franklinu Rooseveltovi

⁷¹ *operations research* nebo v původním britském označení *operational research*

⁷² Právě Roweovi je připisováno autorství termínu „operační výzkum“.

⁷³ mj. držitel Nobelovy ceny za fyziku z roku 1948

jehož v polovině roku 1938 vystřídal Albert Percival Rowe⁷².

V roce 1940 byla pod vedením fyzika Patrick M. S. Blackett⁷³ ustanovena multioborová skupina (zahrnující tři fyziology, jednoho obecného fyzika, dva matematické fyziky, dva matematiky, jednoho astrofyzika, jednoho armádního úředníka a jednoho geodeta) s organizačním názvem *The Anti-Craft Command Research Group, Royal Air Force*, jejímž úkolem bylo zjištění možného využití radarů při protiletadlové střelbě. Tato skupina byla sice označována jako *Blackettův cirkus*, ale podařilo se jim ukázat funkčnost multioborových týmů v operačním výzkumu a zároveň i demonstrovat jejich užitečnost při řešení komplexních (tj. velmi spleťových) reálných problémů. Díky tomu se začaly formovat i další podobné týmy. Tomuto „cirkusu“ se podařilo snížit průměrný počet výstřelů protivzdušné obrany potřebných ke zničení nepřátelského letounu z více než 20000 kusů až na pouhé 4000 při Bitvě o Británii v roce 1941. Blackett o sobě říkal, že jeho cílem je nalezení čísel, na nichž by místo nárazových emocí měla být založena vojenská strategie. Proto Blackett např. kritizoval množství zdrojů investovaných do plošných náletů, které podle něj měly jít do jiných armádních složek, jelikož jeho studie ukázaly neefektivnost těchto bombardovacích plánů ve srovnání s důležitostí bojů proti německým ponorkám, které potápěním obchodních lodí výrazně ovlivňovaly válečné úsilí. Tímto názorem samozřejmě tehdejší vojenské velení příliš nenadchl, takže byl po zásluze odměněn tím, že jej odstříhli z různých komunikačních okruhů. Nicméně po válce [sic!] mu bylo ve spojenecké zprávě *United States Strategic Bombing Survey* dáno za pravdu.

Když byl Blackett v prosinci 1941 dotazován britskou admirálitou ohledně vytvoření sekce operačního výzkumu, napsal memorandum s názvem *Scientists at the operational level*, které silně rezonovalo na obou stranách Atlantiku. Díky tomuto podnětu vznikla skupina *U.S. Navy Anti-submarine Warfare Operations Research Group* (ASWORG) a v lednu 1942 se Blackett do vedení britského námořnictva, aby vytvořil také skupinu věnující se operačnímu výzkumu. Ta dosáhla velkého úspěchu, když se jim podařilo určit optimální velikost obchodního konvoje, při kterém budou minimalizovány ztráty při útoku ponorek v kombinaci s požadavky na doprovod. Nebylo totiž zřejmé, zda jsou výhodnější malé konvoje (které mohou plout rychleji a jsou hůře detekovatelné německými ponorkami) nebo velké konvoje (které mají více válečných lodí schopných bránit náklad). Blackettova skupina ukázala, že mnohem důležitější je počet doprovodných plavidel než velikost samotného konvoje. Závěr proto byl, že několik velkých konvojů je mnohem lépe bránitelných než mnoho malých konvojů, což po implementování do námořní strategie vedlo k výraznému snížení ztrát. Americký ASWORG byl první skupinou s civilními zaměstnanci zapojenou do vojenského operačního výzkumu. Na začátku se jednalo o 15 zaměstnanců, jejichž počet se v průběhu války rozrostl až ke stovce. V říjnu 1942, když vrcholila válka, dorazila první delegace amerických analytiků operačního výzkumu do Anglie, aby spolupracovala s osmým americkým bombardovacím komandem (*VIII Bomber Command*). Jejich prvním úkolem bylo zlepšení přesnosti bombardování. Na základě svých kvantitativních studií předchozích útoků navrhli, že nejlepší bombardér by měl být v hlavním letounu tak, aby mohl zacílit celou dávku bomb vyhozenou v salvě tím, že letadla přesně dodržují úzkou formaci, čímž se výrazně sníží rozptýlení dávky bomb. Na základě této analýzy došlo k nejméně 1000 procentnímu nárůstu počtu bomb dopadajících na stanovené cíle.

Metody operačního výzkumu během války sice byly stále vyvíjeny přesně na míru konkrétním vojenským problémům, ale jejich výsledky zaznamenaly velký úspěch a všichni si byli dobře vědomi, že je to velmi strategické odvětví, a tudíž je potřeba mít k dispozici vědce, kteří ovládají vojenské plánování a řešení související vládních problémů. Proto vláda USA v prosinci 1945 založila *Projekt RAND*⁷⁴, ze kterého se v únoru 1948 stala nezávislá nezisková organizace a která

⁷⁴ Research and Development

⁷⁵ Operations Research Group (ORG)

⁷⁶ Operations Evaluation Group (OEG)

⁷⁷ Project SCOOP (U.S. Air Force Scientific Computation of Optimum Programs)

sehrál v následném rozvoji operačního výzkumu klíčovou roli. Vznikaly také další neméně významné organizace. Vzhledem ke klesajícímu počtu nepřátelských ponorek a vzniku nových aplikací operačního výzkumu v námořnictví, došlo k přejmenování skupiny na ASWORG na *Skupinu operačního výzkumu*⁷⁵ a jejímu podstoupení pod Atlantické loďstvo Spojených států amerických. Jenže po válce se velikost této skupiny smrškla cca na čtvrtinu, což se samozřejmě nelíbilo jejímu vedení, takže v listopadu došlo k dalšímu přejmenování na *Skupinu pro vyhodnocování operací*⁷⁶ a především k jejímu propojení s Massachusettským technologickým institutem (MIT). V červnu 1947 také Pentagon zřídil v rámci letectva výzkumnou skupinu nazývanou *Projekt SCOOP*⁷⁷. Jejím cílem bylo nalezení vhodných (a rychlých) odpovědí na problémy plánování požadavků letectva, např. určení časového harmonogramu materiálových potřeb na podporu válečného plánu. V době vzniku těchto organizací tomu ale ještě stále chyběl jakýsi obecný pohled. Ten však přišel záhy (podzim 1947) díky rozvoji algoritmického řešení úloh *lineárního programování*, tj.

$$f(x) := \langle c, x \rangle = \sum_{i=1}^n c_i x_i \rightarrow \min, \quad A_1 x \leq b_1, \quad A_2 x = b_2 \quad (1.3.1)$$

a případně $x_i \geq 0$ pro $i \in I \subseteq \{1, \dots, n\}$, kde $c \in \mathbb{R}^n$, $A_1 \in \mathbb{R}^{m \times n}$, $A_2 \in \mathbb{R}^{k \times n}$, $b_1 \in \mathbb{R}^m$ a $b_2 \in \mathbb{R}^k$ jsou dané vektory a matice.

Příklad 1.3.1

Jako jednoduchý příklad může posloužit situace z nedávné doby karanténních opatření kvůli pandemii COVID-19. Ta způsobila, že nám po v lednici pár dnech zbyla pouze syrová mrkev, syrové bílé zelí a zavařená okurky. Internet našťestí fungoval, takže jsme si na webových stránkách WHO (nebo spíše Blesku pro ženy) našli *nutriční standardy*, z nichž jsme si vybrali tři faktory: vitamín A, vitamín C a vlákninu. Veškeré potřebné informace jsou shrnuté v následující tabulce (cena okurek je ovlivněna nákupem v „předkaranténní“ době ze zbytků armádních zásob, které by jinak byly nejspíše neprodejně).

	mrkev	zelí	okurky	minimální požadované množství v 1 porci
vitamín A [mg/kg]	35	0,5	0,5	0,5 mg
vitamín C [mg/kg]	60	300	10	15 mg
vláknina [g/kg]	30	20	10	4 g
cena [Kč/kg]	15	12,50	3,75	

Samozřejmě v této těžké době nemá smysl hrát si na hrdiny ani na milionáře, takže jsme chtěli utrácet za jídlo co nejméně, což nás přivedlo k tomu, že náš jídelníček musí být



Theodor Samuel Motzkin (26. března 1908 – 15. prosinec 1970) byl izraelsko-americký matematik. Jeho otec ve svých třinácti letech přestěhoval do Berlína kvůli studiu matematiky. Po univerzitních studiích byl přijat Leopoldem Kroneckerem k doktorskému studiu, avšak toho zanechal před dokončením disertační práce kvůli zapojení do sionistického hnutí. Theodor Motzkin prokázal své matematické nadání již v dětství. Už v patnácti letech nastoupil na univerzitu a v souladu s tehdejšími zvyky prošel univerzitami v Göttingenu, Paříži a Berlíně (světová centra matematiky na začátku 20. století). Disertační práci obhájil v roce 1936 na téma lineárních nerovností a lineárního programování.

⁷⁹ Robert Dorfman (27. října 1916 – 24. června 2002) byl profesorem politické ekonomie na Harvardově univerzitě. Významně přispěl k rozvoji ekonomie, statistiky, skupinových testů a teorie kódování.



Moritz Werner Fenchel (3. května 1905 – 24. ledna 1988) byl německý matematik, který kvůli nacistickým protizidovským zákonům emigroval v roce 1933 do Dánska. Do roku 1933 působil na Univerzitě v Göttingenu, odkud se přesunul na Kodaňskou univerzitu. V roce 1946 byl zvolen čle-

založen na řešení úlohy (1.3.1) ve tvaru

$$15x_M + 12,5x_Z + 3,75x_O \rightarrow \min$$

$$35x_M + 0,5x_Z + 0,5x_O \geq 0,5$$

$$60x_M + 300x_Z + 10x_O \geq 15$$

$$30x_M + 20x_Z + 10x_O \geq 4$$

$$x_1 \geq 0 \quad \& \quad x_2 \geq 0 \quad \& \quad x_3 \geq 0,$$

kde x_M , x_Z a x_O postupně představují konzumované množství mrkve, zelí a okurek. Nezápornost jednotlivých proměnných je celkem přirozený požadavek, jehož případné porušení by znamenalo, že budeme některé suroviny dále přeprodávat za nákupní ceny. To ale rozhodně nepřicházelo v úvahu – dělit se se sousedy, ani náhodou... Následující dny jsme tak přežili jenom díky tomu, že náš jídelníček byl založen na konzumaci 9,52634 g mrkve, 38,265 g zelí a 294,891 g okurek s cenou 1,72705 Kč za jednu porci.

Jedná se o nejjednodušší (a zároveň nejstudovanější) typ úlohy *matematického programování*, což je vlastně rozhodovací proces, ve kterém se má určit efektivní využití omezeného množství zdrojů k dosažení požadovaných cílů. Díky tomuto zmechanizování řešení došlo k obrovskému rozmachu úloh lineárního programování a následně i jiných typů v ekonomii, průmyslu a moha dalších odvětvích současnosti. V této souvislosti je ale zapotřebí zmínit jeden velmi důležitý fakt, který asi není na první pohled úplně zřejmý a který velmi úzce s požadavkem uprostřed (1.3.1). Zatímco teorie ohledně řešení soustav lineárních rovnic je velmi dobře prozkoumána nejméně od poloviny 19. století díky intenzivnímu studiu maticové algebry, podobná otázka řešení soustav lineárních nerovnic byla nejméně do roku 1936 na okraji jakéhokoli matematického zájmu. Avšak díky Theodoru Motzkinovi⁷⁸ začala být i této oblasti věnována intenzivnější pozornost, díky čemuž mohlo být lineární programování vybudováno na velmi pevných základech.

Samozřejmě existuje velké množství problémů, které nelze linearizovat. Proto v závislosti na typu účelové funkce f a přípustné množiny X se vyvinuly další typy úloh matematického programování – souhrnně pojmenovatelné jako *nelineární programování*. Samotné pojmenování „matematické programování“ pochází z roku 1953 od Roberta Dorfmana⁷⁹, kterému se označení lineární programování zdálo příliš restriktivní či zavádějící (jelikož ne vše bylo vždy pouze lineární, viz [29]). Nelineární programování vzniká v roce 1951 a jeho základy tvoří tzv. *Lagrangeův princip* a Johnovy podmínky z roku 1948, čemuž se budeme věnovat později i my. Nejjednodušší úlohou nelineárního programování je např. již dříve zmíněné *kvadratické programování*

$$f(x) := \frac{1}{2} \langle Cx, x \rangle + \langle d, x \rangle \rightarrow \min, \quad A_1 x \leq b_1, \quad A_2 x = b_2$$

a případně $x_i \geq 0$ pro $i \in I \subseteq \{1, \dots, n\}$, kde $C \in \mathbb{R}^{n \times n}$, $d \in \mathbb{R}^n$, $A_1 \in \mathbb{R}^{m \times n}$, $A_2 \in \mathbb{R}^{k \times n}$, $b_1 \in \mathbb{R}^m$ a $b_2 \in \mathbb{R}^k$ jsou dané vektory a matice, přičemž matice C je navíc symetrická a pozitivně semi-definitní, tj. $C = C^T \geq 0$. Tyto úlohy hrají důležitou roli např. při tvorbě optimálního portfolia. Avšak kromě kategorie nelineárního programování patří všechny tyto úlohy mezi problémy tzv. *konvexního programování*, tj. úlohy, kde hledáme minimum „konvexní funkce na konvexní množině“. Vznik této oblasti je úzce spojen s Wernerem Fenchelem⁸⁰, který v roce 1949 položil základy skutečné (tj. systematické) základy teorie konvexních funkcí. Takové funkce a množiny byly studovány již v první polovině 20. století. Za zmínku stojí především Constantin Carathéodory, Hermann Minkowski (ten byl asi první), Ernst Steinitz, Farkas Gyula (nebo Julius Farkas). S většinou z těchto jmen se v našem výkladu ještě setkáme. Fenchelova monografie/referát [37],

nem Královské dánské akademie věd a vzdělanosti. V letech 1949–1951 působil na Stanfordově univerzitě a na Princetonské univerzitě. Po návratu byl krátce mezi lety 1952 a 1956 profesorem mechaniky na polytechnice v Kodani, aby se v roce 1956 vrátil zpět na univerzitu v Kodani, kde pracoval do 1974 jako profesor matematiky.

81



Ralph Tyrrell Rockafellar (★ 10. února 1935) je americkým matematikem, který je v současnosti emeritním profesorem Oddělení matematiky a aplikované matematiky na Washingtonské univerzitě. Je jedním z předních odborníků na teorii optimalizace a související oblasti analýzy a kombinatoriky. V roce 1999 získal *John von Neumann Theory Prize* od *Institute for Operations Research and the Management Sciences* (INFORMS; Institut pro operační výzkum a vědy o řízení) za „zásadní a trvalé příspěvky k teorii operačního výzkumu a vědám o řízení“. V roce 1992 získal nejvyšší ocenění *Society for Industrial and Applied Mathematics* (SIAM; Společnost pro průmyslovou a aplikovanou matematiku), když získal *John von Neumann Prize*. Tato cena je udělována za „vynikající a mimořádné příspěvky do oblasti aplikovaných matematických věd a za efektivní komunikaci těchto myšlenek v rámci komunity“.

kteřá vznikla na základě jeho přednášek na Princetonské univerzitě na jaře 1951, výrazně inspirovala mnohé zájemce o tuto problematiku. V roce 1970 vydal Ralph Rockafellar⁸¹ knihu s názvem *Konvexní analýza*, viz [62], což je nejspíše první výskyt takového slovního spojení, které bylo následně obecně přijato. Podle poděkování v předmluvě je jeho autorem zřejmě Albert Tucker⁸². V čem ale tkví důležitost celého konceptu konvexnosti? To velmi trefně vystihl samotný Rockafellar ve své přednášce při udělování *John von Neumann Prize* v roce 1992:

One distinguishing idea which dominates many issues in optimization theory is convexity. . . An important reason is the fact that when a convex function is minimized over a convex set every locally optimal solution is global. Also, first-order necessary conditions for optimality turn out to be sufficient. A variety of other properties conducive to computation and interpretation of soluti-

*ons ride on convexity as well. In fact the great watershed in optimization isn't between linearity and nonlinearity, but convexity and nonconvexity.*⁸³

viz [63, str. 184–185]. Jinými slovy, díky konvexnosti není potřeba rozlišovat mezi lokálním a globálním minimy a v případě nalezení stacionárního bodu účelové funkce v přípustné množině není nutné ověřovat ještě nějaké dodatečné podmínky druhého řádu.

Přidáme-li k úloze lineárního programování (1.3.1) navíc požadavek celočíselnosti jednotlivých proměnných, získáme úlohu *celočíslného programování*, případně *smíšeného programování*, pokud vyžadujeme celočíselnost pouze od některých proměnných. Toto je velmi obsáhlá oblast matematického programování, která byla iniciována Ralphem Gomorym⁸⁴ v roce 1958. Zde je samozřejmě nutné zdůraznit, že takové úlohy nelze zanedbáním podmínek celočíselnosti řešit jako klasickou úlohu lineárního programování. Kdyby totiž řešení takové úlohy bylo neceločíselné, pak bychom se totiž následným zaokrouhlením mohli dostat mimo přípustnou množinu nebo do jiného bodu, než je skutečné minimum. Pro „malé“ úlohy celočíselného programování dokážeme s pomocí naší intuice a trochy selského rozumu přesně nalézt jejich řešení, což se rozhodně nedá říci o úlohách lineárního programování. V dalších kapitolách uvidíme, že z matematického pohledu je ta situace diametrálně odlišná. Velmi. . .

Další důležitou oblastí operačního výzkumu, které se budeme věnovat, je *dynamické programování*. Jedná se o víceřadový rozhodovací proces, ve kterém rozhodovací proměnná může být mít buď celočíselnou hodnotu nebo libovolnou hodnotu z nějakého předem daného intervalu. I takové úlohy se dají formulovat jako problémy lineárního (či matematického) programování, jenže jejich řešení tímto způsobem nemusí být úplně snadné a efektivní. Naštěstí časová (či jiná) souslednost jednotlivých kroků (tj. dynamika celého procesu) umožňuje i jiný pohled na tuto oblast. Pokud jednotlivé fáze celého procesu vykazují jakousi *separovatelnost*, lze ně nahlížet optikou *dynamického programování*. Navíc se v praxi může celkem snadno stát, že ani neznáme přesné důsledky našich rozhodnutí, což nás přivádí do oblasti *stochastického dynamického programování* nebo v jednodušším případě pouze ke *stochastickému programování*, které se poprvé objevilo v Dantzigově článku z roku 1955, viz [19].



Albert William Tucker (28. listopadu 1905 – 25. ledna 1995) byl kanadský matematik, který výrazně ovlivnil topologii, teorie her a především teorii nelineárního programování. V roce 1950 poprvé formalizoval nezákladnější problém teorie her, který následně nazval *věznovým dilematem*. Jedním z jeho mnoha doktorských studentů byl i John Nash, který v roce 1994 získal *Cenu Švédské národní banky za rozvoj ekonomické vědy na památku Alfreda Nobela*. V letech 1961–1962 byl prezidentem Americké matematické asociace.

⁸³ *To, co odlišuje mnoho optimalizačních problémů, je konvexnost... A to zejména proto, že když konvexní funkce je minimalizována na konvexní množině, pak každé lokální minimum je současně globálním minimumem. Také nutné podmínky prvního řádu pro lokální extrém se stanou postačujícími. Řada dalších vlastností usnadňující výpočty a interpretaci jejich výsledků také velmi úzce souvisí s konvexností. V této oblasti tedy není klíčové rozlišovat mezi lineární a nelineární úlohou, ale mezi konvexní a nekonvexní úlohou.*



Ralph Edward Gomory (* 7. května 1929) je americký matematik a vrcholový manažer. Pracoval v IBM, kde jeho výzkum vedl ke vzniku nových oblastí aplikované matematiky.

KAPITOLA 2

Lineární programování

2.1 Trocha historie (snad) nikoho nezabije	36
2.2 Malá ukázka na začátek	49
2.3 Základem všeho je slušný náčrtek	53
2.4 Teoretické základy aneb být, či nebýt BPB	62
2.5 Dualita aneb peníze až na prvním místě	70
2.6 Simplexový algoritmus	76
2.7 Simplexová tabulka	82
2.8 Hledání výchozího BPB	91
2.9 Lineární lomené programování	93

MP: YOU ARE OFTEN CALLED THE *FATHER OF LINEAR PROGRAMMING*. IS THAT A TITLE YOU'RE COMFORTABLE WITH?

DANTZIG: I HAVE TO TELL YOU A STORY. TWENTY-FIVE YEARS AGO I VISITED JAPAN FOR THE FIRST TIME. WHEN I GOT OFF THE PLANE, THE JAPANESE WHO MET ME WERE VERY SURPRISED AT HOW YOUNG I WAS. SINCE I HAD BEEN BILLED AS THE FATHER OF LINEAR PROGRAMMING, THEY APPARENTLY EXPECTED TO SEE AN OLD MAN WITH WHITE HAIR AND A CANE BEING HELPED DOWN THE RAMP. I WAS 45 AT THE TIME.

MP: THE PREFACE TO YOUR BOOK *LINEAR PROGRAMMING AND EXTENSIONS* OPENS WITH A PROVOCATIVE STATEMENT: "THE FINAL TEST OF A THEORY IS ITS CAPACITY TO SOLVE THE PROBLEMS WHICH ORIGINATED IT."

DANTZIG: DID I SAY THAT? IT'S A GREAT QUOTE. SHOW ME WHERE...¹

Viz poutavý rozhovor s Georgem Dantzigem v [3].

¹ **MP:** Často jste nazýván „otcem lineárního programování“. Líbí se vám toto označení?

Dantzig: Povím vám historku. Před 25 lety jsem poprvé navštívil Japonsko. Když jsem vystoupil z letadla, Japonci, se kterými jsem se setkal, byli překvapeni, jak jsem byl mladý. Jelikož jsem byl označován za otce lineárního programování, očividně očekávali, že uvidí starého muže s bílými vlasy a holí pomáhající mu dolů na pojezdových schodech. V té době mi bylo 45 let.

MP: Předmluvu vaší knihy „Lineární programování a jeho rozvoj“ začínáte provokativními slovy: „Konečným testem teorie je její schopnost vyřešit problémy, kvůli kterým vznikla.“

Dantzig: To jsem řekl? To je úžasný citát. Ukažte mi kde...

2



François Quesnay (4. června 1694 – 16. prosince 1774) byl francouzský ekonom a lékař. Byl také jedním z hlavních představitelů první ekonomické školy (tzv. *fyziokratů*), kterou později nahradila *klasická ekonomie* (první moderní škola ekonomie vycházející z díla Adama Smithe *Pojednání o podstatě a původu bohatství národů* z roku 1776). Quesnay je znám především díky jeho knize *Ekonomická tabulka* (*Tableau économique*) z roku 1758, která byla prvním modelem ekonomického koloběhu.

3



John von Neumann (28. prosince 1903 – 8. února 1957; rodným jménem Margittai Neumann János Lajos) byl maďarsko-americký matematik, fyzik, počítačový vědec (informatik) a polyhistor. Významně přispěl k rozvoji funkcionální analýzy, teorie množin, numerické analýzy, statistiky a mnoha dalších matematických disciplín stejně jako kvantové fyziky, ekonomie, informatiky a hydrodynamiky. Byl průkopníkem využití matematické teorie operátorů v kvantové mechanice, což považoval za svůj největší přínos. Byl také klíčovou postavou v rozvoji teorie her i celulárních automatů a digitálních počítačů. Bývá označován za *posledního představitele velkých matematiků, kteří se věnovali čisté i aplikované matematice a během své kariéry oba tyto směry rozvíjeli*, viz [44, str. 89]. Už během svého života se stal sku-

2.1 Trocha historie (snad) nikoho nezabije

Problém „lineárního programování“ lze (v jistém smyslu) nalézt již v knize z roku 1758 od Françoise Quesnaye², ve které se hledá vzájemný vztah statkáře, rolníka a řemeslníka. Ovšem během dalších 150 let nalezneme jen velmi málo využití lineárního modelu, ačkoli se objevuje jako část Walrasova modelu ekonomické rovnováhy z roku 1874. Až do roku 1930 se lineární model příliš nestudoval. Většina matematiků – ekonomů se věnovala analýze teoretických problémů spojených s možností hospodářských rovnováh a efektivností rozdělování výrobních prostředků v konkurenčním nebo monopolním prostředí. V těchto studiích uvažovali klasické konvexní funkce se spojitou derivací, které se hodí více na zobrazení podmínek stability než funkce založené na lineárních nerovnostech. Za zmínku stojí především úsilí skupiny rakouských a německých ekonomů, kteří v roce 1930 pracovali na zobecnění Walrasovy lineární „technologie“. Tato práce vyvolala pozornost a nejspíše podnítila Johna von Neumanna³ v roce 1932 formulovat model dynamického lineárního programování (příslušný článek vyšel v roce 1937), ve kterém uvedl alternativní metody výroby daných výrobků samostatně (odděleně, tj. *diskrétně*) nebo současně (tj. *spojitě*). Von Neumann předpokládal a) stálý koeficient růstu národního hospodářství, b) úplně soběstačné národní hospodářství. Ačkoli jeho model neměl žádný explicitní účel, von Neumann ukázal, že síly trhu maximalizují míru hospodářské expanze, a dokázal, že při maximu se tato míra rovná úrokové míře z kapitálu investovaného do výroby. Nicméně praktický vliv von Neumannova článku si nebyl příliš velký, neboť obsahoval (podobně jako jiné teoretické články) „jen“ zajímavé teoretické věty. Matematici – ekonomové se pravděpodobně zajímali více o získání podobných výsledků pro obecnější model, neboť „pro mnohé ekonomy je výraz *linearity* spojený se zúžením, omezením a nepružností hypotéz“. Jinými slovy, toto úsilí patřilo jako mnoho jiných do kvalitativní oblasti národního hospodářství tehdejší doby – světa, ve kterém účelem matematického modelu byl spíše kvalitativní než kvantitativní popis předpokládaných vzájemných vztahů, viz [17, 61].

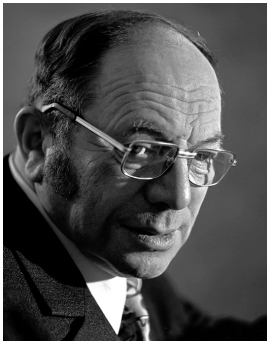
Průkopníkem lineárního programování je sovětský matematik Leonid Kantorovich⁴, který byl jako mladý profesor Leningradské univerzity v roce 1938 požádán o pomoc místním tristem výrobců překlíček se sestavením harmonogramu výroby pro 8 soustruhů tak, aby se maximalizoval zisk z produkce 5 různých druhů překlíček v dané skladbě. Ovšem po roce 1939 již dále nemohl pokračovat v této oblasti z politických (válečných) důvodů. V roce 1942 sepsal jednu ze svých ekonomických prací *The Best Use of Economic Resources*, ve které Kantorovich využívá optimalizační metody v širokém spektru ekonomických problémů, viz [53] a také přepis stenografického záznamu z roku 1939 v [54]. Kniha ale vyšla až v roce 1959 a chtělo by se říci *Post bellum auxilium* (s křížkem po funuse), neboť v té době toho již bylo o lineárním programování známo skutečně mnoho. Jenže v době, kdy tato kniha vznikala, určovala hlavní směr vědeckého zkoumání především druhá světová válka. Kantorovich proto nastoupil do armády, kde učil.

Při obklíčení Leningradu (září 1941 – leden 1944) tvořilo Ladožské jezero a řeka Něva hlavní dopravní tepny proti obklíčení. Po osvobození v lednu 1944 měly tanky a cisterny přejet přes toto jezero, jenže se báli protržení ledu. Kantorovich určil sílu ledu a vypočetl, že vzdálenost tanků musí být alespoň 27 metrů, aby se led neprotrhl. Sám si pak nastoupil do prvního tanku.

Za své výsledky v oblasti optimalizace získal v roce 1949 Stalinovu cenu (a to byl nějaký matematik) a v roce 1975 společně s Tjallingem Koopmansem⁵ obdrželi dokonce *Cenu Švédské národní*

tečnou matematickou legendou. V roce 1924 pobýval na univerzitě v Göttingenu a od 30. let je spojen především s univerzitou v Princetonu. Po vypuknutí 2. světové války se stal jednou z hlavních postav projektu *Manhattan*, jehož cílem bylo vyvinutí atomové bomby. Měl také vášeň pro automobily, ovšem ne pro to, aby se v nich „štoural“, ale aby je řídil, jako by to byly těžkotonážní tanky. Každý rok se v Princetonu objevil s novým kouskem, neboť vzhledem ke způsobu, jakým jezdil, mu žádné auto nemohlo vydržet déle. Pravidelně býval zatčen za překročení povolené rychlosti a některé z jeho nehod se staly legendárními. Jedna z princetonových křižovatek byla po nějakou dobu známá jako *Von Neumannův roh* kvůli množství nehod, které tam způsobil. Jednou vyšel z úplně zdemolovaného auta s jednoduchým vysvětlením: „Jel jsem ulicí. Stromy na pravé straně mě míjely pravidelně rychlostí 60 mil za hodinu. Najednou se mi jeden z nich postavil do cesty. Bum!“, viz [15, str. 96].

4



Leonid Vitaliyevich Kantorovich (19. ledna 1912 – 7. dubna 1986) byl sovětský matematik a ekonom. Kantorovichovy první práce se dotýkaly problematiky operací na projektivních množinách a měly zcela teoretický charakter. Ve svém vědeckém vývoji se Kantorovich postupně dostával k „aplikovanějším“ otázkám, nikdy však nepřerušil aktivní kontakt s nejabstraktnější matematickou teorií. Skutečnost, že matematik Kantorovich se stal také ekonomem, by bylo nesprávné vidět v tom, že dokázal úspěšně „algoritmizovat“ úlohy, vznikající v hospodářské praxi. Podstatné jsou obecnější implikace jeho práce, umožněné jeho mimořádnými schopnostmi výstižně formalizovat ekonomické problémy a adekvátně interpretovat relevantní

banky za rozvoj ekonomické vědy na památku Alfreda Nobela za jejich „příspěvek k teorii optimální alokace zdrojů“.

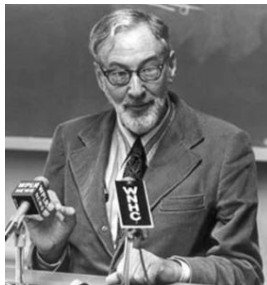
Navzdory nezměrnému a nezpochybnitelnému Kantorovichovu přínosu k celé oblasti parametrické optimalizace ten nejzásadnější zlom, který skutečně rozpohyboval celou mašinérii, je spojen s jiným jménem. Během války se úlohy lineárního programování využívaly ke snižování nákladů a zvyšování ztrát nepřítele. Typickým příkladem může být např. *dopravní problém* pro rozesílání zboží ze zásobovacích skladů až na místo určení, který můžete využít buď zcela mírově pro minimalizaci vlastních nákladů (plánováním rozvozu a nikoli rozvážením *ad hoc*) nebo pro maximalizaci ztrát nepřítele (demolice klíčových kolejových tratí). Tento problém asi poprvé objevuje v práci z roku 1930 u sovětského matematika A. Tolstoje⁶, na které později navázal i Kantorovich, viz [52, 72, 73]. Klasický dopravní problém jakožto úloha lineárního programování byl zformulován v článku [50] z roku 1941 od Franka Hitchcocka⁷ včetně návrhu možného řešení. Podobnému problému se během druhé světové války věnoval i Koopmans, když pracoval pro *Britsko-americký úřad pro kombinovanou přepravu*⁸. Vlastně před rokem 1947 byly publikovány pouze 4 články pojednávající o speciálních případech lineárního programování:

- (i) Jean-Baptiste Joseph Fourier (asi první, kdo si uvědomil důležitost řešení soustavy lineárních nerovnic v aplikované matematice; v článku [39] z 1826 je metoda pro jejich řešení, která je vlastně jakýmsi předchůdcem simplexové metody, viz také [51, 79])
- (ii) Charles-Jean de la Vallée Poussin (metoda z článku [74] z roku 1911 pro Čebyševův problém, která je dalším předchůdcem základní metody pro řešení úloh lineárního programování, viz také [38])
- (iii) L. Kantorovich (už jsme se zmínili o jeho práci z roku 1939)
- (iv) Frank Lauren Hitchcock (a jeho dopravní problém).

V roce 1947 přichází se svým objevem G. Dantzig⁹. V červnu 1941 před obhajobou své disertační práce Dantzig přijal práci ve Washingtonu jako vedoucího oddělení založeného Pentagonem pro analýzy bitev v útvaru statistické kontroly v rámci armádního letectva Spojených států¹⁰. Jeho motivací bylo „podílení se na druhé světové válce“, jelikož předpokládal, že USA se brzy do války aktivně zapojí. Vyvinul systém hlášení, díky němuž byly bojové jednotky schopné zaznamenávat počty vzletů, ztracených a zničených letadel, shozených bomb a napadených cílů. Právě zde se Dantzig důvěrně seznámil s přístupem letectva k plánování programů souvisejících činností, což jsou myšlenky, které mu pomohly se sestavením základní struktury lineárních modelů. Mezi jeho kolegy byli např. Robert McNamara (ze kterého se později stal ministr obrany USA a také šéf Světové banky) nebo Warren Hirsch (který se věnoval pravděpodobnosti a v roce 1957 zformuloval domněnku ohledně maximální počtu kroků potřebných pro řešení úloh lineárního programování, viz [81]) Za svoji zásluhnou civilní práci v časech války získal Dantzig nejvyšší vojenské ocenění *Exceptional Civilian Service Award* z ministerstva války.

Dantzig se tak stal expertem na *metody programování plánování pomocí stolních kalkulačků*. V roce 1946 se vrátil na Kalifornskou univerzitu v Berkeley, aby formálně splnil všechny studijní požadavky a získal konečně doktorát. V Berkeley mu dokonce na ústavu matematiky nabídli učitelské místo, ale on ji z prostého důvodu nepřijal: „bylo málo placené“. Vrátil se proto do Pentagonu, kde začal pracovat jako matematický poradce v rámci nově ustanoveného oddělení finanční kontroly letectva USA¹¹. Svě setrvání v Pentagonu ale považoval spíše za dočasné, neboť se přirozeně stále poohlížel i po nějaké akademické pozici. Jenže kolegové ho chtěli udržet, takže přicházeli s různými výzvami ohledně možností mechanizace plánovacích procesů a jednou z nich byla žádost o nalezení způsobu, jak co nejrychleji vypočítat program rozmístění

výsledky. Nezůstal však jen u optimalizace dílčích úloh, snažil se též uplatnit vyvinuté optimalizační procedury a principy na úrovni celého národního hospodářství. Kantorovichovi se dostalo nejvyšších ocenění za jeho výsledky v ekonomickém výzkumu jak v SSSR, tak na mezinárodní úrovni, jeho cesta k těmto počtům však nebyla lehká a obhajoba a prosazování vědeckých prvků v procesu řízení si vyžádalo hodně úsilí. Je jediným (ryze) sovětským držitelem Nobelovy ceny za ekonomii, viz [75].



Tjalling Charles Koopmans (28. srpna 1910 – 26. února 1985) byl nizozemsko-americký matematik a ekonom. První tři léta na universitě v Utrechtu se věnoval především matematice, poté pak studiu teoretické fyziky. První dvě odborné statě publikoval z oblasti kvantové mechaniky a doktorát získal v roce 1936 v oboru matematické statistiky na universitě v Leidenu za práci *Lineární regresní analýza ekonomických časových řad*. Velká deprese počátku 30. let probudila v Koopmansovi zájem o ekonomii, zejména o matematickou ekonomii. V současnosti patrně nejvíce ceněnými jsou výsledky, které Koopmans dosáhl v oblasti modelování výrobního procesu a v makroekonomické teorii ekonomického růstu. Tradiční nástroj ekonomů pro popis výrobního procesu – produkční funkce, založenou na předpokladu jeho efektivního uspořádání a určitého specifického chování ekonomických subjektů, nahradil modelem, který má popisovat výrobní proces nezávisle na institucionální struktuře ekonomiky, na aspiracích ekonomických subjektů. Ukázal souvislost mezi jím zavedeným pojmem výrobní efektivnosti, cenovým systémem a alokací zdrojů v ideální konkurenční ekonomice a tím vytvořil most mezi normativní teorií alokace zdrojů a teorií všeobecné rovnováhy. Významných výsledků dosáhl Koopmans rovněž v oblasti ekonomiky do-

mužstva a vybavení, tréninků a logistické podpory. Díky své matematické průpravě se pustil do sestavování modelu. V tomto ohledu byl inspirován (a zároveň i fascinován) prací Wassilye Leontiefa¹², který v roce 1932 přišel s „Mezipřůmyslovým input–output modelem“ americké ekonomiky mající jednoduchou maticovou strukturu. Dantzig ale s tímto *statickým* modelem příliš spokojen nebyl – postrádal v něm totiž dynamiku umožňující změny v průběhu času. V Leontiefově modelu bylo jednoznačné spojení mezi výrobním procesem a výrobky generovanými těmito procesy. Chyběly tomu *alternativní činnosti!* Reálné využití by mělo být ve velkém měřítku se stovkami položek a činností, ale současně to musí být i *spočítatelné*. Jakmile byl model hotov¹³, bylo zapotřebí najít praktický způsob výpočtu toho, jaká množství těchto činností je potřeba využít, aby to bylo v souladu s jejich „input–output“ charakterem a dostupným množstvím. Jenže na začátku Dantzig *neuvažoval* ve svém modelu žádnou účelovou funkci: explicitní cíle neexistovaly, jelikož plánovači z praxe jednoduše neměli způsob, jak něco takového implementovat. Bez toho se ale dostáváme do skutečně obřích potíží, protože např. při přiřazování každého ze 70 mužů k právě jednomu ze 70 úkolů rozumíme „činností“ přidělení i -tého muže k j -tému úkolu s binární hodnotou 0 (úkol není přidělen) nebo 1 (úkol je přidělen), v kterémžto případě dostáváme 2×70 omezení a 70×70 činností. Jenže takový problém má $70! \approx 1,12 \times 10^{100}$ možností. Ale na základě jakých kritérií z nich vybrat to nejlepší?!? A to nemluvíme o tom, jak dlouho by nám vůbec trvalo projít všechny možnosti¹⁴. Tento jednoduchý příklad jednoduše ilustruje, proč až do roku 1947 a vlastně i do dnešních dní existuje obrovská propast našimi sny a činy. Můžeme si stanovit naše plány tak, abychom optimalizovali jednotlivé cíle, jenže k jejich dosažení máme nepřeberné množství způsobů (každý s různými výhodami a nevýhodami), takže jejich vzájemné porovnávání se zdá být nemožné, natož některé z nich označit jako „to nejlepší“. Nakonec vždy skončíme u intuice a zkušeností (našich či ostatních). Do tohoto stavu se Dantzig dostal v závěru roku 1946, kdy měl sestavený striktně technicistní model zdárně zahrnující všechny běžné souvislosti. Tohle všechno se samozřejmě dělo v době „počítačového pravěku“. Stolní počítače sice k dispozici byly, ale k jejich použití bylo zapotřebí mít přesně formulovaný model, což si vyžádalo *axiomatizaci* celé oblasti. A přesně to udělal i Dantzig v polovině roku 1947. Rozhodl se, že účelová funkce musí být zcela explicitní, a sestavil plánovací problém jako sadu axiomů, které byly dvou druhů: první zahrnovaly vyráběné či spotřebovávané položky a druhá sada obsahovala činnosti či výrobní procesy, do nichž mohly být vkládány nebo z nichž mohly vystupovat položky v pevně daných poměrech, dokud tyto poměry byly vzájemnými nezápornými násobky. Výsledkem pak bylo hledání řešení matematického systému, ve kterém jde o *minimalizaci lineární formy vzhledem k lineárním rovnostem a nerovnostem*. Neotřelost takového přístupu tkvěla právě ve využití lineární formy jakožto účelové funkce. Dantzig tak zavedením *programování v lineární struktuře*¹⁵ zásadně zmodernizoval plánovací proces.

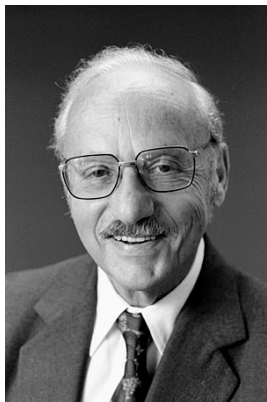
Jenže teď byla na řadě odpověď na zcela přirozenou otázku odkazující na Dantzigův rozhovor zmíněný v úvodu této kapitoly: dokážeme takový systém vyřešit? Dantzig kvůli tomu dokonce v červenci 1947 vyrazil za Koopmansem, neboť předpokládal, že ekonomové se takovým problémem již věnovali. Koopmans z toho byl samozřejmě nadšený, protože v tom spatřoval velký potenciál směrem k různým ekonomickým aplikacím. Začal rozšiřovat poznatky týkající se modelů založených na lineárním programování mezi mladé ekonomy, což nakonec vedlo až k několika Nobelovým cenám za ekonomii. Jenže Dantzigovi s jeho problémem nepomohl. Rozhodl se proto zkusit štěstí „sám u sebe“ a vymyslet nějaký vlastní algoritmus. To jej vrátilo zpět k jeho disertační práci na Berkeley sepsané pod vedením Jerzyho Neymana¹⁶, v níž se věnoval dvěma slavným nevyřešeným problémům z matematické statistiky:

pravy a ekonometrických metod. Uznávané jsou i jeho zásluhy o rozvoj lineárního programování. Neopomenutelný je také způsob, kterým ve svých pracích uplatňoval matematické modelování. Bývá považován za vzor využívání této metodologie ekonomického výzkumu, za vzor důsledného dodržování pravidel vědeckého postupu, rozhodného odmítnutí argumentů a analogií, kterými je možné přesvědčit a získat, avšak pouze za cenu nepřesnosti nebo skrytých hodnotových postojů, viz [76].

⁶ Další z těch, o kterém nevíme vůbec nic.

⁷ Frank Lauren Hitchcock (6. března 1875 – 31. května 1957) byl americký matematik a fyzik.

⁸ British–American Combined Shipping Board



George Bernard Dantzig (8. listopadu 1914– 13. května 2005) byl americký matematik, který se věnoval průmyslovému inženýrství, operačnímu výzkumu, matematické informatice, ekonomii a statistice. Je považován za *otce lineárního programování*, jemuž vděčí život díky objevu a následnému zveřejnění *simplexového algoritmu*.

¹⁰ Army Air Force's Combat Analysis Branch of Statistical Control

¹¹ U.S. Air Force Comptroller

¹² Wassily Wassilyevich Leontief (5. srpna 1905 – 5. února 1999), byl rusko-americký ekonom známý především díky jeho „input-output“ analýze a studiu efektu změn v jednom ekonomickém sektoru na jiné. V roce 1973 získal *Cenu Švédské národní banky za rozvoj ekonomické vědy na památku Alfreda Nobela* za „rozvoj input-output analýzy a její apli-

Jednoho dne [Dantzig] dorazil na přednášky pozdě, a tak si v rychlosti opsal dva problémy ze statistiky napsané na tabuli, neboť se domníval, že jde o domácí úkol. Úlohy se mu však zdály neobvykle obtížné, a tak je vyřešil až po delší době. Když je potom profesorovi [Neymanovi] donesl s omluvou, že domácí úlohu donesl tak pozdě, byl vyzván, aby je hodil na stůl (kde byly kupy papírů, jak už to tak na některých pracovních stolech bývá). Domníval se, že o svých příkladech už neuslyší. Po dlouhé době jej v neděli ráno vzbudilo bouchání na dveře, za nimiž stál jeho profesor se slovy: „Napsal jsem předmluvu k Vašemu článku, přečtěte si to, ať to můžeme zaslat redakci!“ Až tehdy Dantzig zjistil, že vyřešil dva dosud otevřené problémy ze statistiky. A tím byla na světě podstata jeho disertační práce¹⁷.

Disertační práce byla (z dnešního pohledu) o hledání Lagrangeových multiplikátorů pro obecný lineární optimalizační problém s lineárními omezeními v integrálním tvaru. Jeho řešení bylo založeno na jisté geometrizaci související s přístupem skrze sloupce jakési matice místo obvyklých řádků. Toto jej v létě 1947 přivedlo k tzv. *simplexovému algoritmu* a přesvědčení, že to musí fungovat. A vskutku fungovalo! Viz také [22].

Prvním opravdovým testem simplexového algoritmu bylo určení optimálního jídelníčku s minimálními náklady, díky čemuž snad každá učebnice věnovaná lineárnímu programování začíná obdobným příkladem jako jsme to učinili my v předchozí kapitole. Jenže tady šlo o zcela reálný problém, ve kterém bylo sledováno 9 nutričních hodnot, k jejichž naplnění bylo k dispozici 77 surovin. Jednalo se o klasický *Stiglerův*¹⁸ *problém optimálního jídelníčku* z roku 1945 určený pro průměrně činného muže vážícího 68 kg, jehož cílem bylo nalezení jídelníčku splňujícího doporučení *Národní rady pro výzkum*¹⁹ z roku 1943 uvedená v následující tabulce, viz také [68].

	doporučená denní dávka
kalorie	3000 jednotek
bílkoviny	70 g
vápník	0.8 g
železo	12 mg
vitamín A	5000 jednotek
tiamin (vitamín B1)	1.8 mg
riboflavin (vitamín B2 nebo G)	2.7 mg
niacin	18 mg
kyselina askorbová	75 mg

Tabulka 2.1: Minimální požadavky pro Stiglerův problém optimálního jídelníčku.

Toho mělo být docíleno pomocí 77 surovin/potravin s cenami z roku 1939 a nutričními hodnotami, které získal z amerického ministerstva zemědělství, viz (dlouhou) tabulku níže. Jenže tyto hodnoty samozřejmě nejsou (a snad ani nemohou) být zcela přesné, což Stigler ve svém článku ilustroval na šesti nejmarkantnějších případech a následně okomentoval slovy

kaci v důležitých ekonomických problémech". Tutéž cenu (za jiné aktivity) získali i čtyři jeho doktorští studenti: Paul Samuelson (1970), Robert Solow (1987), Vernon Smith (2002) a Thomas Schelling (2005).

¹³ Z dnešního pohledu se jednalo o vícekrokový dynamický lineární model s trojúhelníkovou (či dokonce blokově diagonální) maticí.

¹⁴ Kdybychom každou sekundou stihli projít 1 milion možností a začali bychom s tím už v době Velkého třesku, tak bychom k dnešnímu dni ... neměli hotovo skoro nic.

¹⁵ Tento název vydržel cca 1 rok. V létě 1948 navštívil Dantzig a Kopmans společnost RAND v kalifornské Santa Monice a při jedné z procházek podél tamní pláže Kopmans řekl „Proč nezkrátit název *programování v lineární struktuře* pouze na *lineární programování*?“. Na to Dantzig zareagoval slovy „To je ono! Od teď se to bude nazývat takto.“ a ještě ten-
týž den přednesl v RANDu přednášku s názvem *Lineární programování*.

¹⁶ Jerzy Neyman (16. dubna 1894 – 5. srpna 1981), byl polský matematik a statistik, který strávil první část profesní dráhy na různých institucích ve Varšavě a poté na University College London, zatímco druhá část je spojena s Kalifornskou univerzitou v Berkeley. Položil základy moderní teoretické statistiky. Představil nové přístupy k testování statistických hypotéz a je spoluautorem konceptu tzv. nulové hypotézy, který našel uplatnění v genetice, lékařské diagnostice, astronomii i meteorologii.

¹⁷ Tato historka údajně inspirovala i kázání jistého reverenda v Americe jako ukázka pozitivního myšlení – kdyby Dantzig věděl, že jde o otevřené problémy, pravděpodobně by je tak snadno nevyřešil, neboť by si tolik nevěřil jako v případě domácího úkolu.

*Enough difficulties have been indicated to suggest the almost infinite complexity of a refined and accurate assessment of nutritive value of a diet.*²⁰

*In light of the foregoing remarks it should not be necessary to belabor the tentativeness of the figures.*²¹

¹⁸ George Joseph Stigler (17. ledna 1911 – 1. prosince 1991) byl americký ekonom a společně s Miltonem Friedmanem byli předními představiteli Chicagské ekonomické školy. V roce 1982 získal *Cenu Švédské národní banky za rozvoj ekonomické vědy na památku Alfreda Nobela* za „významné studie o průmyslové struktuře, fungování trhu a o příčinách a následcích veřejné regulace“.

¹⁹ National Research Council

²⁰ *Byl zmíněn dostatek komplikací, abychom ukázali téměř nekonečnou složitost softistikovaného a přesného stanovení nutričních hodnot jídelničky.*

²¹ *Ve světle předchozích poznámek snad není nezbytné rozvádět nepřesnost těchto čísel.*

Tabulka 2.2: Dostupné suroviny/potraviny pro Stiglerův problém optimálního jídelničky.

		množství živin obsažené v porci za 1 USD									
		množství za 1 USD [g]	kalorie [1000 j]	bílkoviny [g]	vápník [g]	železo [mg]	vitamín A [1000 j]	tiamin [mg]	riboflavin [mg]	niacin [mg]	kyselina askorbová [mg]
1.	pšeničná mouka	12600	44,7	1411	2	365		55,4	33,3	441	
2.	makarony	3217	11,6	418	0,7	54		3,2	1,9	68	
3.	pšeničné cereálie	3280	11,8	377	14,4	175		14,4	8,8	114	
4.	kukuřičné lupínky	3194	11,4	252	0,1	56		13,5	2,3	68	
5.	kukuřičná mouka	9861	36	897	1,7	99	30,9	17,4	7,9	106	
6.	kukuřičná kaše	8005	28,6	680	0,8	80		10,6	1,6	110	
7.	rýže	6048	21,2	460	0,6	41		2	4,8	60	
8.	ovesné vločky	6389	25,3	907	5,1	341		37,1	8,9	64	
9.	bílý chleba	5742	15	488	2,5	115		13,8	8,5	126	
10.	celozrnný chleba	4985	12,2	484	2,7	125		13,9	6,4	160	
11.	žitný chleba	4930	12,4	439	1,1	82		9,9	3	66	
12.	pound cake („piškotový dort“)	1829	8	130	0,4	31	18,9	2,8	3	17	
13.	slané krekry	3004	12,5	288	0,5	50					
14.	mléko	8867	6,1	310	10,5	18	16,8	4	16	7	177
15.	sušené mléko	6035	8,4	422	15,1	9	26	3	23,5	11	60
16.	máslo	1473	10,8	9	0,2	3	44,2		0,2	2	
17.	margarín	2817	20,6	17	0,6	6	55,8	0,2			

18.	vejce	1857	2,9	238	1	52	18,6	2,8	6,5	1	
19.	čedar	1874	7,4	448	16,4	19	28,1	0,8	10,3	4	
20.	smetana	1689	3,5	49	1,7	3	16,9	0,6	2,5		17
21.	burákové máslo	2534	15,7	661	1	48		9,6	8,1	471	
22.	majonéza	1198	8,6	18	0,2	8	2,7	0,4	0,5		
23.	Crisco (rostlinný tuk)	2234	20,1								
24.	sádlo	4628	41,7				0,2		0,5	5	
25.	roštěnka	1145	2,9	166	0,1	34	0,2	2,1	2,9	69	
26.	kýta	1246	2,2	214	0,1	32	0,4	2,5	2,4	87	
27.	pečená žebra	1553	3,4	213	0,1	33			2		
28.	hovězí krk	2007	3,6	309	0,2	46	0,4	1	4	120	
29.	hovězí bok	3107	8,5	404	0,2	62		0,9			
30.	hovězí játra	1692	2,2	333	0,2	139	169,2	6,4	50,8	316	525
31.	skopové stehno	1643	3,1	245	0,1	20		2,8	3,9	86	
32.	skopová žebra	1239	3,3	140	0,1	15		1,7	2,7	54	
33.	vepřová žebra	1477	3,5	196	0,2	30		17,4	2,7	60	
34.	vepřová pečeně	1874	4,4	249	0,3	37		18,2	3,6	79	
35.	slanina	1772	10,4	152	0,2	23		1,8	1,8	71	
36.	uzená šunka	1655	6,7	212	0,2	31		9,9	3,3	50	
37.	„solené vepřové”	2835	18,8	164	0,1	26		1,4	1,8		
38.	pečené kuře	1497	1,8	184	0,1	30	0,1	0,9	1,8	68	46

39.	telecí kotlet	1072	1,7	156	0,1	24		1,4	2,4	57	
40.	losos	3489	5,8	705	6,8	45	3,5	1	4,9	209	
41.	jablka	9072	5,8	27	0,5	36	7,3	3,6	2,7	5	544
42.	banány	4932	4,9	60	0,4	30	17,4	2,5	3,5	28	498
43.	citróny	2380	1	21	0,5	14		0,5		4	952
44.	pomeranče	4439	2,2	40	1,1	18	11,1	3,6	1,3	10	1998
45.	zelené fazole	5750	2,4	138	3,7	80	69	4,3	5,8	37	862
46.	kapusta	8949	2,6	125	4	36	7,2	9	4,5	26	5369
47.	mrkev	6080	2,7	73	2,8	43	188,5	6,1	4,3	89	608
48.	celer	3915	0,9	51	3	23	0,9	1,4	1,4	9	313
49.	hlávkový salát	2247	0,4	27	1,1	22	112,4	1,8	3,4	11	449
50.	cibule	11844	5,8	166	3,8	59	16,6	4,7	5,9	21	1184
51.	brambory	1681	14,3	336	1,8	118	6,7	29,4	7,1	198	2522
52.	špenát	4592	1,1	106		138	918,4	5,7	13,8	33	2755
53.	batáty	7649	9,6	138	2,7	54	290,7	8,4	5,4	83	1912
54.	broskve (plechovka)	4894	3,7	20	0,4	10	21,5	0,5	1	31	196
55.	hrušky (plechovka)	4030	3	8	0,3	8	0,8	0,8	0,8	5	81
56.	ananas (plechovka)	3993	2,4	16	0,4	8	2	2,8	0,8	7	399
57.	chřest	1945	0,4	33	0,3	12	16,3	1,4	2,1	17	272
58.	zelené fazole (plechovka)	5386	1	54	2	65	53,9	1,6	4,3	32	431

59.	vepřové s fazolemi (plechovka)	63889	7,5	364	4	134	3,5	8,3	7,7	56	
60.	kukuřice (plechovka)	5452	5,2	136	0,2	16	12	1,6	2,7	42	218
61.	hrášek (plechovka)	4109	2,3	136	0,6	45	84,9	4,9	2,5	37	370
62.	rajčata (plechovka)	6263	1,3	63	0,7	38	53,2	3,4	2,5	36	1253
63.	rajčatová polévka (plechovka)	3917	1,6	71	0,6	43	57,9	3,5	2,4	67	862
64.	broskve (sušené)	2889	8,5	87	1,7	173	86,8	1,2	4,3	55	57
65.	švestky (sušené)	4284	12,8	99	2,5	154	85,7	3,9	4,3	65	257
66.	rozinky	4524	13,5	104	2,5	136	4,5	6,3	1,4	2,4	136
67.	hrášek (sušený)	5742	20	1367	4,2	345	2,9	28,7	18,4	162	
68.	bílé fazole	5097	17,4	1055	3,7	459	5,1	26,9	38,2	93	
69.	fazole navy	7688	26,9	1691	11,4	792		38,4	24,6	217	
70.	káva	2025						4	5,1	50	
71.	čaj	652							2,3	42	
72.	kakao	2637	8,7	237	3	72		2	11,9	40	
73.	čokoláda	1400	8	77	1,3	39		0,9	3,4	14	
74.	cukr	8773	34,9								
75.	kukuřičný sirup	4966	14,7		0,5	74				5	
76.	melasa	3752	9		10,3	244		1,9	7,5	146	
77.	jahodový džem	2213	6,4	11	0,4	7	0,2	0,2	0,4	3	

Stiglerovým prvním krokem bylo nahrazení některých surovin jinými, „výživnějšími“, čímž snížil jejich počet ze 77 na 15. Pak se ale dostal do potíží, neboť jak sám píše

*Thereafter the procedure is experimental because there does not appear to be any direct method of finding the minimum of a linear function subject to linear conditions.*²²

²² Další postup už je zkoušením, protože neexistuje žádná přímá metoda pro nalezení minima lineární funkce při lineárních omezeních.

takže metodou pokus–omyl vyzkoušel 510 vyzkoušel kombinací a dospěl k celkovým denním nákladům 0,1093 USD (tj. 39,93 USD za rok) pro jídelníček složený z

- pšeničné mouky (1) za 0,0365 USD (tj. 459,9 gramů)
- sušeného mléka (15) za 0,0105134 USD (tj. 63,448 gramů)
- kapusty (46) za 0,011252 USD (tj. 100,694 gramů)
- špenátu (52) za 0,005065 USD (tj. 23,258 gramů)
- fazolí navy (69) za 0,0459959 USD (tj. 353,61648 gramů)

²³ Neexistuje žádný důvod k tomu, abychom věřili, že jsme našli nejlevnější kombinaci... Na druhou stranu ale roční náklady nemohou být sníženy o více než pár dolarů lepším výběrem z těchto komodit.

Stigler netvrdil, že našel skutečně optimální řešení

*There is no reason to believe that the cheapest combination was found... On the other hand the annual cost could not have been reduced by more than a few dollars by a better selection from these commodities.*²³

²⁴ Další z těch, o kterých se mi nepodařilo nic najít.

ale nabídl několik důvěryhodných argumentů k uvěření, že roční náklady nemohou být sníženy o více než několik málo dolarů. A on očividně věděl, co dělá a říká! Na podzim 1947 provedl Jack Laderman²⁴ z *Projektů matematického tabelování v Národním úřadu standardů*^{25,26} po 120 človeko-dnech výpočtů na ručně ovládaných stolních kalkulátorech^{27,28} našel skutečně optimální řešení s denními náklady 0,108662 USD, tj. 39,6888 USD za rok, což se od původního Stiglerova výsledku liší o pouhých 24 centů. Samozřejmě pro domácí vaření nemá taková úloha asi příliš velký smysl (vázně?), ale vraťte se zpět k armádním základům celého operačního výzkumu a uvažte armádu USA s cca 12 000 000 vojáků v roce 1945... Nalezené řešení je založeno na

²⁵ *Mathematical Tables Project of the National Bureau of Standards* byla jedna z největších a nejsložitějších výpočetních organizací vytvořená před vynálezem digitálních elektronických počítačů.

²⁶ V červnu 1947 se Dantzig stal součástí Projektu SCOPCO coby tamní šéf matematiků. Tento projekt sponzoroval i dva externí programy: první byl součástí *Národního úřadu standardů*, aby její zaměstnanci mohli experimentovat se simplexovou metodou, a druhý byl spojen s *Graduate School of Industrial Administration na Univerzitě Carnegieho-Mellonových* v Pittsburghu s cílem implementovat nástroje lineárního programování do průmyslu.

²⁷ Abychom ještě více přiblížili tuto dobu: v roce 1945 trval na tehdejších kalkulátorech výpočet inverze matice 10 × 10 přibližně 15 človeko-dnů!

²⁸ Ne všechny úlohy byly takto výpočetně náročné. Např. v roce 1949 se pomocí simplexové metody podařilo vyřešit *dopravní problém* s 25 výchozími místy a 60 cílovými místy během pouhých 9 človeko-dnů ručních výpočtů, viz [18]. To bylo způsobeno tím, že tento typ problémů má velmi speciální strukturu.

- pšeničné mouky (1) za 0,0295191 USD (tj. 371,94066 gramů)
- hovězích jater (30) za 0,00189256 USD (tj. 3,20221152 gramů)
- kapusty (46) za 0,0112144 USD (tj. 100,3576656 gramů)
- špenátu (52) za 0,00500766 USD (tj. 22,99517472 gramů)
- fazolí navy (69) za 0,0610286 USD (tj. 469,1878768 gramů)

s příjmem 3000 kalorií, 147,414 g bílkovin, 0,8 g vápníku, 60,4669 mg železa, 5000 jednotek vitamínu A, 4,12044 mg tiaminu, 2,7 mg riboflavinu, 27,316 mg niacinu a 75 mg kyseliny askorbové. V tomto řešení jsou ale některé doporučené denní dávky překročeny na úkor minimální ceny. Kdybychom chtěli dosáhnout zcela přesné doporučené denní dávky, pak by minimální náklady vzrostly na 0,138171 USD (tj. na roční náklady 50,46696 USD) a jídelníček by byl složen z

- pšeničné mouky (1) za 0,0180918 USD (tj. 227,95668 gramů)
- kukuřičné mouky (5) za 0,0170207 USD (tj. 167,8411227 gramů)
- sušeného mléka (15) za 0,0452014 USD (tj. 272,790449 gramů)
- burákového másla (21) za 0,00317017 USD (tj. 8,03321078 gramů)
- sádla (24) za 0,0243964 USD (tj. 112,9065392 gramů)
- hovězích jater (30) za 0,015273 USD (tj. 25,841916 gramů)
- kapusty (46) za 0,00921709 USD (tj. 82,48373841 gramů)
- brambor (51) za 0,0051373 USD (tj. 8,6358013 gramů)
- špenátu (52) za 0,000663078 USD (tj. 3,044854176 gramů)

Samozřejmě ne vždy se hned na poprvé podaří získat smysluplný jídelníček (či výsledek jiného modelu) a správně matematizovat daný problém je skutečné umění – samotný Dantzig o tom ví své. Na začátku 50. let totiž dostal od svého lékaře doporučení, aby upravil svoji stravu. Rozhodl se proto, že si k tomu po vzoru Stiglera sestaví vlastní lineární program a svůj nový jídelníček založí na jeho řešení. Ovšem nešlo mu o minimalizaci ceny. Místo toho se chtěl vyhnout nejčastějšímu úskalí každé diety: neustálý *pocit hladu*, takže cílem bylo maximalizovat celkovou váhu jídla poníženou o množství obsažené vody. Jelikož v té době už Dantzig pracoval ve společnosti RAND (od června 1952), využil k jeho řešení tamní počítač IBM 701, který naštěstí dával výsledky mnohem rychleji než v roce 1947. Jednoho dne své ženě oznámil, že už nastal „ten den“, a cokoli 701 oznámí, to by mu měla nachystat. Vstupní data zahrnovala informace o více než 500 surovinách a jídel. Když mu v 5 hodin odpoledne manželka zavolala, aby zjistila, jaká bude večeře, oznámil ji Dantzig výsledek. Manželka to okomentovala slovy „Je to trochu divné, ale realizovatelné“. Ovšem poté Dantzig dodal poslední položku v seznamu: 1892 litrů octu. To samozřejmě nebyl příliš dobrý nápad, takže další den Dantzig trochu poupravil některé požadavky, což jej přivedlo k novému jídelníčku. Ten byl tentokrát založen na konzumaci 200 kostek bujONU – na to Dantzigova manželka zareagovala řečnickou otázkou, zda „se chystá ovládnout trh s bujónovými kostkami“. Dantzig (jako správný manžel) se ale nedal její poznámkou odradit a druhý den ráno otestoval, kolik dokáže takových kostek sníst. Dal si čtyři do hrnku s horkou vodou a... nedalo se to ani polknout. Bylo to neskutečně slané. Do modelu proto přidal omezení množství přijímané soli, které má lidský organismus přirozeně zabudované, ale matematický model je trochu něco jiného. Třetí pokus jej přivedl k požívání cca 900 gramů otrub denně a čtvrtý k obdobnému množství třtinové melasy. To už bylo na jeho manželku trochu moc, takže celý tento problém vzala do svých rukou a nasadila mu vlastní jídelníček, díky němuž Dantzig zhubl 10 kg, viz [23].

Nicméně trvalo téměř rok než si Dantzig a ostatní uvědomili skutečnou sílu simplexového algoritmu. Během té doby, se Dantzig rozhodl navštívit von Neumanna, aby zjistil, zda by nemohl pomoci s metodami řešení. Poprvé ho navštívil 3. října 1947 v Princetonu. Ačkoli už v té době už byl mnohými považován za největšího žijícího matematika, snažil se mu Dantzig popsat studované problémy jako normálnímu smrtelníkovi. Začal se sestavováním modelu lineárního programování pomocí činností atd. V tom ho von Neumann zastavil se slovy „Jděte k věci.“ A jak si přál, tak se stalo. Během jedné minuty mu Dantzig „plácnul“ na tabuli algebraickou i geometrickou verzi celého problému. Na to von Neumann zareagoval slovy „Aha, toto.“ a následovala jeho cca 90 minutová přednáška o matematické teorii lineárního programování. Dantzig zůstal sedět s vyvalenýma očima a otevřenou pusou, vždyť nic z toho v literatuře nebylo! Von Neumann si toho v jednu chvíli všiml a řekl

²⁹ *Nechci, abyste si myslel, že to všechno teď tady tahám bezprostředně z rukávu jako nějaký kouzelník. Nedávno jsem dokončil s Oscarem Morgensternem knihu o teorii her. A já se teď snažím ukázat, že oba problémy jsou ekvivalentní. Teorie, kterou tu nastiňuji, je analogií k té, kterou jsme vybudovali pro hry.*

³⁰ Alan Jerome Hoffman (* 30. května 1924) je americký matematik, který významně ovlivnil rozvoj společnosti IBM. V roce 1992 získal společně s Philipem Wolferem *John von Neumann Theory Prize*.



Harold William Kuhn (29. července 1925 – 2. července 2014) byl americký matematik a bývalý emeritní profesor v Princetonu, který se věnoval především teorii her. V roce 1980 získal společně s Galem a Tuckerem *John von Neumann Theory Prize*. Po mnoho let byl považován za autora tzv. *maďarského algoritmu* pro řešení přiřazovací úlohy z teorie grafů. Ovšem v roce 2006 se ukázalo, že tentýž algoritmus navrhoval již Jacobi v pracech, které byly posmrtně publikovány v roce 1890.

³² David Gale (13. prosince 1921 – 7. března 2008) byl americký matematik a ekonom, který působil především na Kalifornské univerzitě v Berkeley. Svůj výzkum zaměřil především na matematickou ekonomii, teorii her a konvexní analýzu. V roce 1960 publikoval knihu *The Theory of Linear Economic Models*, kterou přispěl k dalšímu rozšíření lineárního programování. V roce 1962 s Lloydem Shapleyem napsali článek věnovaný *problému stabilního manželství* s algoritmem, který se v dnešní době používá pro rozdělování žáků do veřejných škol v New Yorku a Bostonu, viz také <https://odkaz.page.link/smp1>.

*I don't want you to think I am pulling all this out of my sleeve on the spur of the moment like a magician. I have recently completed a book with Oscar Morgenstern on the theory of games. What I am doing is conjecturing that the two problems are equivalent. The theory that I am outlining is an analogue to the one we have developed for games.*²⁹

Díky tomu se Dantzig poprvé dozvěděl o *Farkasova(-Minkowského) lemmatu* a *dualitě*. Von Neumann slíbil Dantzigovi, že se nad jeho výpočetním problémem zamyslí a během pár týdnů se ozve. To splnil a navrhl jistý nelineární iterativní algoritmus. Někdy kolem roku 1952 lidé ze skupiny Alana Hoffmana³⁰ von Neumannův návrh otestovali na několika problémech a porovnali jej se simplexovým algoritmem a s metodou navrhovanou Motzkinem. Simplexový algoritmus byl jasný vítěz, viz [21]. Mimochodem v roce 2000 byl v časopise *Computing in Science & Engineering* vyhlášen seznam *The Top 10 Algorithms*, které byly vybrány jako ty s *největším vlivem na vývoj a využití vědy a techniky ve 20. století*. Dantzigův algoritmus v něm samozřejmě nemohl chybět, viz [28].

Při další návštěvě v Princetonu v červnu 1948 se Dantzig setkal s Tuckerem, který krátce poté začal se svým studentem Haraldem Kuhnem³¹ a Davidem Galem³² budovat teorii her, nelineárního programování a duality. Výsledky této skupiny z ní učinily ústřední bod matematického výzkumu v této oblasti. O 12 let později se Tucker, který v té době pročítal rukopis Dantzigovy knihy [20], se setkali znovu a společně zabředli i do následujícího rozhovoru

AT: *Proč přisuzujete dualitu von Neumannovi a ne mojí skupině?*

GD: *Protože on byl první, kdo mi ji ukázal.*

AT: *To je divné, protože jsme nenašli žádný záznam o tom, co von Neumann udělal. Našli jsme akorát jeho článek „On a Maximizing Problem“.*

GD: *To je pravda. Pošlu vám můj článek, který jsem napsal na základě mého prvního setkání s von Neumannem.*

Dantzig poslal Tuckerovi zprávu *A Theorem on Linear Inequalities* datovanou 5. ledna 1948, která obsahovala (zdá se) první rigorózní důkaz duality. „Proč jste to nepublikoval“ zeptal se později Tucker, na což Dantzig reagoval slovy „Protože to nebyly moje výsledky – byly von Neumannovi. Jediné, co jsem udělal, bylo sepsání důkazu naznačeného von Neumannem, aby mohl dále kolovat. Byl to můj způsob vzdělávání lidí z mého úřadu v Pentagonu.“ Dnes je von Neumann obvykle považován za zakladatele teorie duality, zatímco Kuhnovi, Tuckerovi a Galeovi je připisován její první rigorózní důkaz.

Začátky lineárního programování skutečně nebyly lehké, což dokládá ještě jedna Dantzigova vzpomínka spojená s von Neumannem. Nedlouho po prvním setkání s Tuckerem se ve Wisconsinu konalo setkání ekonomické společnosti, kterého se zúčastnili světoznámí matematici jako von Neumann či Harold Hotelling³³ a ekonomové jako Koopmans. Dantzig byl v té době době spíše „mladý neznámý“, a tak není divu, že byl před takovým publikem celkem vystrašený při své přednášce věnované lineárnímu programování. Po přednášce následovala diskuze. Chvilí bylo ticho, než se přihlásil Hotelling. Vstal, takže to vypadalo, že zabírá asi polovinu místnosti (rád plaval v oceánu a říkalo se o něm, že vždy způsobí znatelný vzestup hladiny), a prohlásil: *But we all know the world is nonlinear.*³⁴ Tím vynesl nad Dantzigovou přednáškou devastující verdikt. Než se Dantzig zmohl na nějakou odpověď, zvedla se další ruka se slovy: *Mr. Chairman,*

³³ Harold Hotelling (29. září 1895 – 26. prosince 1973) byl americký matematik, statistik a vlivný ekonomický teoretik. Jeho jméno je spojováno s Hotellingovým zákonem, Hotellingovým lemmatem a Hotellingovým pravidlem známými z ekonomie.

³⁴ *Ale my všichni víme, že svět je nelineární.*

³⁵ *Pane předsedo, pane předsedo, pokud to řečníkovi nevdá, rád bych odpovídal za něj.*

³⁶ *Řečník nazval svoji přednášku „Lineární programování“ a pečlivě představil své axiomy. Máte-li příklad, který tyto axiomy splňuje, rozhodně to využijte. V opačném případě to nedělejte.*

³⁷ *International Institute of Applied Systems Analysis*

*Mr. Chairman, if the speaker doesn't mind, I would like to reply for him.*³⁵ Byl to von Neumann a Dantzig samozřejmě souhlasil, načež vstal a pokračoval: *The speaker titled his talk „linear programming“ and carefully stated his axioms. If you have an application that satisfies the axioms, well use it. If it does not, then don't.*³⁶ a sedl si. Když se nad tím zamyslíte, tak Hotelling měl samozřejmě pravdu – svět je značně nelineární. Naštěstí ale systémy lineárních nerovnic (na rozdíl od rovností) nám umožňují aproximovat většinu z těchto nelinearit, které se objevují v praktickém plánování.

V roce 1960 započal Dantzig svoji zářnou akademickou kariéru, když nastoupil na místo profesora na Ústavu průmyslového inženýrství na Kalifornské univerzitě v Berkeley. Poté (1966) se přesunul na Stanfordovu univerzitu, kde se z něj v roce 1985 stal emeritní profesor a aktivnímu výzkumu se věnoval až do roku 1997. Po celou dobu také často navštěvoval *Mezinárodní institut pro aplikovanou systémovou analýzu*³⁷ (IIASA) v rakouském Laxenburgu, což je mezinárodní výzkumná organizace *provádějící mezioborové vědecké studie v oblastech ochrany životního prostředí, ekonomie, technologií a sociální problematiky v kontextu lidského rozsahu globálních změn*. V letech 1973–1974 zde společně se svoji ženou strávil „sabatikal“ a jeho pobyt bývá spojován se dvěma historkami, viz [5, str. 229],

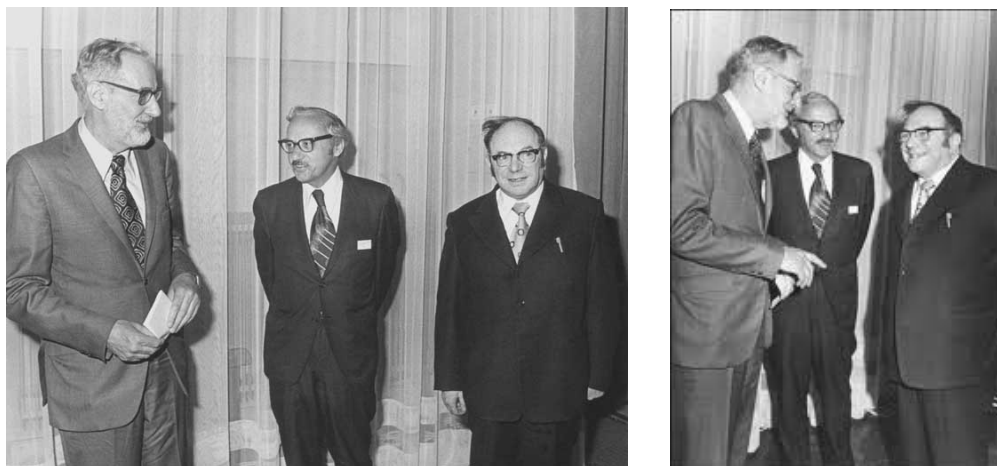
Pár dní po svém příchodu do IIASA volal George [Dantzig] Ruth Steinerové, která měla na starosti návštěvníky, s následující žádostí: „Před mou kanceláří stojí velmi dlouhý nákladní vůz. Nedovedu si představit, že je výhodné mít takovou délku. Můžete mi, prosím, zjistit, co je v něm naloženo, odkud přijel, kudy jel a jak projížděl kolem rohů?“ Přišla odpověď: „Nábytek ze Salzburgu, přes dálnici, častokrát musel couvat a vracet se.“ Za nějaký čas informoval Georgeův výzkumný asistent Ruth, že by firma mohla ušetřit 40 % nákladů, pokud by použila čtyři menší nákladní vozy, a navrhoval, aby to firmě oznámila. To také udělala a následně jej informovala o výsledku: „Mysleli si o mně, že jsem se zbláznila.“

V IIASA byl George také proslulý tím, že pil polévku z pivního püllitru. Když se jej jeho asistentka ptala proč, napsal „dlouhý matematický vzorec dokazující, že tepelné ztráty při použití püllitru jsou menší ve srovnání s polévkovým talířem“.

V roce 1975 byla poprvé udělena *John von Neumann Theory Prize* a jejím držitelem se stal Dantzig za „jeho práci v oblasti lineárního programování“. Při udělení ceny se pak Dantzig podělil s publikem o několik osobních historek souvisejících s von Neumannem. Avšak když v roce 1975 obdrželi Kantorovich s Koopmansem Nobelovu cenu za jejich práce související s lineárním programováním, Dantzig mezi oceněnými nebyl. A ani nikdy později tuto cenu za svůj nezměrný přínos k počátkům a následnému rozvoji lineárního programování, jeho využití v praxi a ekonomické interpretaci nikdy nezískal. To je samozřejmě ve světle všeho, co bylo napsáno výše, zcela udivující a nepochopitelné. Jaké důvody k tomu ve skutečnosti vedly, zatím zůstávají zahaleny – snad byl jeho přístup až příliš matematický. Podklady pro rozhodnutí komise mohou být po 50 letech zveřejněny, tj. nejdříve v roce 2025. Koopmans byl „hluboce zklamán“ tím, že nezískali Nobelovu cenu společně, takže alespoň věnoval 40 000 USD institutu IIASA na Dantzigovu počest³⁸. Koopmans projevoval svoji nespokojenost s neudělením ceny i Dantzigovy také tím, že navrhoval Kantorovichovi, aby toto ocenění odmítli. Byla to asi jedna z nejtěžších diskuzí mezi nimi – zejména pro Kantoroviche. Jenže vývoj šel trochu jiným směrem, než by se na základě rozhodnutí Nobelova shromáždění mohlo zdát. Práce Kantoroviche i Dantziga věnované lineárnímu programování byly zcela původní a nezávislé, jenže Dantzig rozvinul teorii i mnoha dalšími směry (dualita a zejména simplexový algoritmus), takže většina knih a jiných textů je založena

³⁸ Výše Koopmansova daru měla velmi racionální základ. Celková odměna tehdy totiž činila 240 000 USD, takže Koopmansovi zbylo 80 000 USD, které by obdržel, kdyby byl mezi oceněnými i Dantzig.

především na Dantzigových výsledcích. Kantorovichův výzkum neměl žádný významný dopad na rozvoj lineárního programování. Ovšem chcete-li ocenit někoho za prvenství jeho objevu, tak to asi nebude ten, který dokázal uspokojit potřeby všech. Jinými slovy, Kantorovich je jakýsi Kryštof Kolumbus lineárního programování, který je sice považován za „objevitele Ameriky“, ale jejího skutečného území se vlastně nikdy nedotkl, viz [5, str. 231–232].



Obrázek 2.1: Setkání Koopmanse, Dantziga a Kantoroviche (zleva) v roce 1975 v IIASA.

2.2 Malá ukázka na začátek

Malá ukázka na začátek

Příklad 2.2.1

Zemědělec vlastní 1000 arů³⁹ (nebo 10 hektarů) víceméně homogenní zemědělské plochy. Na této ploše může chovat skot nebo pěstovat pšenici, kukuřici či rajčata. Jeden kus skotu zabere 4 ary, jedná se o rodinnou farmu, takže k dispozici je 12000 pracovních hodin, které mohou být využity kdykoli a jakkoli v průběhu celého roku. Následující tabulka ukazuje podrobnosti týkající se zemědělské produkce

	skot	pšenice	kukuřice	rajčata
chovatelné/pěstovatelné množství na 1 aru	1/4	50 bušlů ⁴⁰	80 bušlů	500 Kg
potřebná úhrnná roční pracovní doba	40 hod/ks	10 hod/ar	12 hod/ar	25 hod/ar
zisk	32000 Kč/kus	100 Kč/bušl	120 Kč/bušl	20 Kč/bušl

Navíc je nutné ponechat navíc alespoň část odpovídající 20 % půdy z celkové obhospodařované plochy pro skot a nejvýše 30 % může být využito pro pěstování rajčat, přičemž poměr mezi velikostí půdy určené k pěstování pšenice a neobhospodařované plochy nesmí překročit 2 : 1. Jak rozdělit zemědělskou půdu, aby bylo dosaženo maximálního zisku

při splnění všech daných omezení?

Řešení. Označme symbolem x_1 velikost plochy určené pro chov skotu, x_2 pro pěstování pšenice, x_3 pro kukuřici a x_4 pro rajčata. Pak účelová funkce určující dosažený zisk je

$$f(x_1, x_2, x_3, x_4) = 32000 x_1/4 + 100 \cdot 50 x_2 + 120 \cdot 80 x_3 + 20 \cdot 500 x_4$$

při omezení daném velikostí pozemku

$$x_1 + x_2 + x_3 + x_4 \leq 1000,$$

časovou náročností a dostupnému množství pracovních hodin

$$40 x_1/4 + 10 x_2 + 12 x_3 + 25 x_4 \leq 12000,$$

dodatečnými požadavky na velikost obhospodařované plochy

$$x_1 \geq 0, 2(x_1 + x_2 + x_3 + x_4) \quad \& \quad x_4 \leq 0,3 \cdot 1000 \quad \& \quad x_2 : (1000 - x_1 - x_2 - x_3 - x_4) \leq 2 : 1.$$

Pochopitelně nelze alokovat záporné množství půdy, takže pro našeho zemědělce dostáváme lineární program

$$8000 x_1 + 5000 x_2 + 9600 x_3 + 1000 x_4 \rightarrow \max$$

$$x_1 + x_2 + x_3 + x_4 \leq 1000$$

$$10 x_1 + 10 x_2 + 12 x_3 + 25 x_4 \leq 12000$$

$$8 x_1 - 2 x_2 - 2 x_3 - 2 x_4 \geq 0$$

$$x_4 \leq 300$$

$$2 x_1 + 3 x_2 + 2 x_3 + 2 x_4 \leq 2000$$

$$x_1 \geq 0 \quad \& \quad x_2 \geq 0 \quad \& \quad x_3 \geq 0 \quad \& \quad x_4 \geq 0.$$

[Jeho řešením je $x_1 = 200$, $x_2 = 0$, $x_3 = 769,23$ a $x_4 = 30,76$.]

³⁹ 1 ar je $10 \times 10 \text{ m}^2$

⁴⁰ Bušl je stará objemová a hmotnostní míra, která se používá převážně jako měrná jednotka pro obilniny na angloamerických komoditních burzách. Např. 1 bušl pšenice je cca 27,216 kg a v případě kukuřice jde o cca 25,4016 kg.

Příklad 2.2.2

(*Problém batohu*) Lupič se rozhodl vykrást sklad pražírny kávy. Zde je celkem 6 druhů kávy/kávových bodů. Zloděj má k dispozici batoh se 6 pytlí o stejném objemu jako samotný batoh, tj. 40 litrů. Tyto pytle by chtěl naplnit tak, aby se vešly do batohu a jejich hodnota byla maximální vzhledem k ceně jednotlivých druhů uvedených v následující tabulce.

	1	2	3	4	5	6
cena Kč/kg	800	450	900	700	1500	600
objem l/kg	2	0,9	1,5	2	4	1

Řešení. Pro $i \in \{1, 2, \dots, 6\}$ označme jako x_i množství i -tého druhu kávy, které zloděj dal do i -tého pytle. Pak jeho úkolem vlastně je nalezení řešení lineárního programu

$$\begin{aligned} 800x_1 + 450x_2 + 900x_3 + 700x_4 + 1500x_5 + 600x_6 &\rightarrow \max \\ 2x_1 + 0,9x_2 + 1,5x_3 + 2x_4 + 4x_5 + x_6 &\leq 40 \\ x_1 \geq 0 \ \& \ x_2 \geq 0 \ \& \ x_3 \geq 0 \ \& \ x_4 \geq 0 \ \& \ x_5 \geq 0 \ \& \ x_6 \geq 0. \end{aligned}$$

[Jeho řešením je volba $x_3 = 26\frac{2}{3}$ a $x_1 = x_2 = x_4 = x_5 = x_6 = 0$.] S tímto problémem se setkáme i v příští kapitole, kdy ale budeme „pomáhat“ zloději v knihkupectví.

Příklad 2.2.3

(*Rozdělení portfolia*) Investor má k dispozici kapitál ve výši 20 milionů Kč, který by chtěl investovat do nějaké kombinace dluhopisů, cenných papírů, termínovaného vkladu, spořicího účtu, nemovitého majetku a zlata. Předpokládaná úroková míra, míra rizika a předpokládaný růst hodnoty investice jsou shrnuty v následující tabulce.

	výnos/úroky [% , počítáno ročně]	riziko	předpokládaný růst [%]
dluhopisy	5	3	0
cenné papíry	2	10	7
termínovaný vklad	4	2	0
spořicí účet	3	1	0
nemovitosti	0	5	7
zlato	0	20	11

Např. nemovitosti nemají žádný výnos (nejedná se o pronájem), ale jejich hodnota by měla růst o 7 % ročně. Na druhou stranu cenné papíry se průměrně zúročí o 2 % a navíc jejich cena by měla vzrůst o 7 %, tj. celkový roční výnos je 9 %. Cílem je maximalizovat očekávanou hodnotu investice v průběhu let za podmínek:

- (i) Z celkového objemu investic musí být nejméně 30 % v dluhopisech, nejvýše 10 % v cenných papírech a nejvýše 10 % v termínovaných vkladech a/nebo spořicích účtech;
- (ii) Nejvýše 50 % z objemu investic do nemovitostí může být získáno zpět formou hypotéky s úrokem 6 %. Nejvýše však 3 miliony Kč.
- (iii) Průměrné riziko vztahované na objem investovaných peněz nesmí převýšit 9/2.
- (iv) Průměrný roční úrok vzhledem k objemu investovaných peněz musí být alespoň 2,5 %.
- (v) Do zlata nelze investovat více než 2 miliony Kč nebo 8 % dostupných financí (podle toho, co je menší).

Řešení. Máme jediný zdroj – peníze, které můžeme rozdělit do 6 druhů investic. Pro $i \in \{1, 2, \dots, 6\}$ označme x_i jako množství investované do i -tého typu a x_7 bude představovat

objem vypůjčených peněz. Potom jde o maximalizaci funkce

$$1,05x_1 + 1,09x_2 + 1,04x_3 + 1,03x_4 + 1,07x_5 + 1,11x_6 - 1,06x_7$$

za podmínek

$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \leq 20\,000\,000 + x_7$$

$$x_1 \geq 0,3 \cdot (x_1 + x_2 + x_3 + x_4 + x_5 + x_6)$$

$$x_2 \leq 0,1 \cdot (x_1 + x_2 + x_3 + x_4 + x_5 + x_6)$$

$$x_3 + x_4 \leq 0,1 \cdot (x_1 + x_2 + x_3 + x_4 + x_5 + x_6)$$

$$x_7 \leq 0,5x_5$$

$$x_7 \leq 3\,000\,000$$

$$\frac{3x_1 + 10x_2 + 2x_3 + x_4 + 5x_5 + 20x_6}{x_1 + x_2 + x_3 + x_4 + x_5 + x_6} \leq \frac{9}{2}$$

$$\frac{5x_1 + 2x_2 + 4x_3 + 3x_4}{x_1 + x_2 + x_3 + x_4 + x_5 + x_6} \geq \frac{5}{2}$$

$$x_6 \leq 2\,000\,000$$

$$x_6 \leq 0,08 \cdot (20\,000\,000 + x_7)$$

$$x_1 \geq 0 \quad \& \quad x_2 \geq 0 \quad \& \quad x_3 \geq 0 \quad \& \quad x_4 \geq 0 \quad \& \quad x_5 \geq 0 \quad \& \quad x_6 \geq 0 \quad \& \quad x_7 \geq 0,$$

což už neměl být problém přepsat do tvaru lineárního programu. [Řešením je $x_1 = 8\,740\,000$, $x_2 = 2\,300\,000$, $x_3 = x_4 = 0$, $x_5 = 9\,568\,000$, $x_6 = 92\,000$ a $x_7 = 3\,000\,000$.]

Příklad 2.2.4

(*Optimální výrobní program*) Vyrábíme n různých výrobků s pomocí m -tice různých zdrojů. Na výrobu jednoho kusu j -tého produktu potřebujeme a_{ij} jednotek i -tého zdroje. Zdroje jsou však omezené a my máme k dispozici pouze b_i jednotek i -tého zdroje. Zisk z výroby 1 kusu j -tého výrobku je c_j . Vyrobitme-li celkem x_j kusů j -tého výrobku, pak lineární program pro maximalizaci zisku je

$$\sum_{j=1}^n x_j c_j \rightarrow \max$$

$$Ax \leq b \quad \& \quad x \geq 0.$$

Příklad 2.2.5

(*Směšovací problém*) Máme n -tici základních surovin, ze kterých máme namíchat výrobek předepsaného složení, přičemž samozřejmě chceme minimalizovat surovinové náklady. Množství spotřebovaných jednotek j -té suroviny označme x_j a odpovídající jednotkovou cenu jako c_j . Požadované složení je přesně popsáno vektorem b , jehož složky b_i jsou rovny požadovanému obsahu i -té látky ve výsledné směsi. Jednotkové množství j -té suroviny obsahuje a_{ij} jednotek i -té látky. Cílem tedy je nalezení řešení lineárního

programu

$$\sum_{j=1}^n x_j c_j \rightarrow \max$$

$$Ax = b \quad \& \quad x \geq 0.$$

Příklad 2.2.6

(*Dopravní problém*) Máme m -tici výrobců a n -tici spotřebitelů. Snažíme se rozvést zboží od výrobců je spotřebitelům s minimálními náklady při jistých omezeních.

⁴¹ Toto je věta, kterou jsem slychal snad v každé hodině matematiky na druhém stupni základní školy. Jakkoli se nám to v té době zdálo přehnané, uvidíme napříč celým textem, že v této oblasti matematiky to platí dvojnásob.

$$\langle c, x \rangle \rightarrow \min \tag{2.2.1}$$

$$x \in X := \{x \in \mathbb{R}^n \mid \langle a_i, x \rangle \leq b_i, i = 1, \dots, k \ \& \ \langle a_i, x \rangle = b_i, i = k + 1, \dots, m \ \& \ x_1 \geq 0 \ \& \dots \ \& \ x_n \geq 0\} \tag{2.2.2}$$

2.3 Základem všeho je slušný náčrtek⁴¹

Úlohy v \mathbb{R}^2 (a s mnohem větší mírou úsilí a představivosti i v \mathbb{R}^3) můžeme snadno vyřešit s pomocí jednoduchého obrázku. V takovém případě jsou obvykle jednotlivá omezení ve tvaru nerovností, neboť případná rovnost by úlohu výrazně zjednodušila. Tato omezení vymezují poloroviny s hraničními přímkami s normálovým vektorem a_i a vrstevnice účelové funkce $c_1 x_1 + c_2 x_2$ odpovídají přímkám s normálovým vektorem c . Řešení úlohy (2.2.1) pak znamená nalézt vrstevnici na nejnížší možné úrovni, která má ještě neprázdný průnik s množinou X .

Příklad 2.3.1

Uvažme úlohu

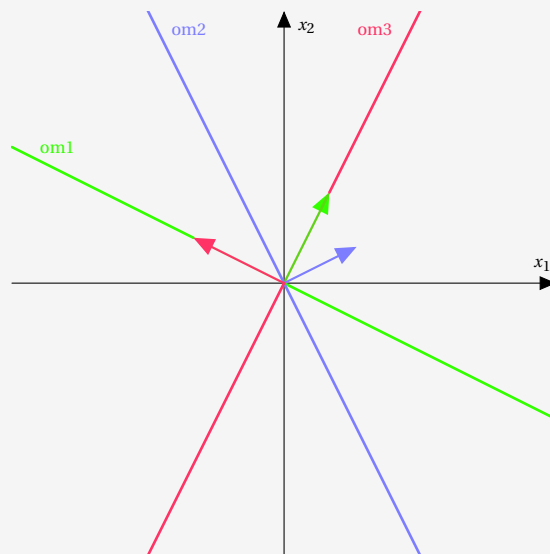
$$-x_1 - x_2 \rightarrow \min$$

$$x_1 + 2x_2 \leq 4 \quad \& \quad 4x_1 + 2x_2 \leq 12 \quad \& \quad -2x_1 + x_2 \leq 1 \quad \& \quad x_1 \geq 0 \quad \& \quad x_2 \geq 0.$$

Přímky a jejich normálové vektory určující první tři omezení jsou zobrazeny na Obrázku 2.2, zatímco pro poslední dvě podmínky dostaneme souřadné osy.

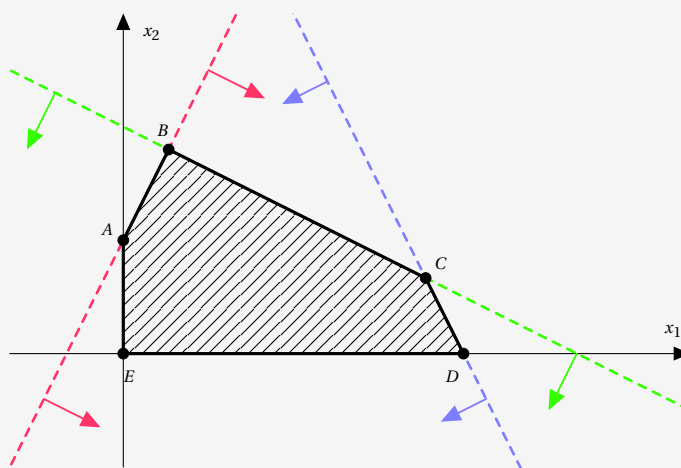
⁴² Samozřejmě mnohem praktičtější je spočítat přímo průsečíky těchto přímek se souřadnými osami, čímž dostaneme pro první omezení body $[4, 0]$ a $[0, 2]$, pro druhé omezení $[3, 0]$ a $[0, 6]$, pro třetí omezení $[-1/2, 0]$ a $[0, 1]$.

⁴³ Je-li to možné, pak je neuvěřitelnější otestovat bod $[0, 0]$.



Obrázek 2.2: Přímky určené levými stranami jednotlivých omezení (tj. $om1: x_1 + 2x_2 = 0$, $om2: 4x_1 + 2x_2 = 0$, $om3: -2x_1 + x_2 = 0$) a jejich normálové vektory.

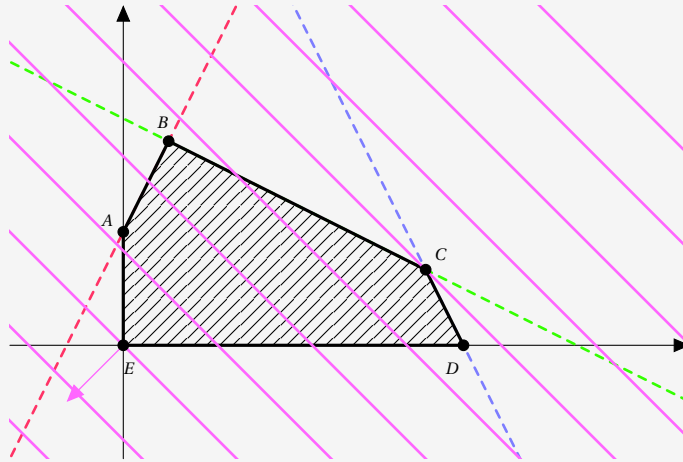
Hodnoty pravých stran pak určují posun těchto přímek (v případě kladného čísla se přímka posune ve směru normálového vektoru⁴²). Tím dostaneme hraniční přímky rozdělující \mathbb{R}^2 vždy na dvě části a potřebujeme určit, která z nich vyhovuje dané nerovnosti. K tomu stačí vybrat libovolný bod neležící na hraniční přímce⁴³ a otestovat, zda je pro něj nerovnost splněna či nikoliv. Tyto vyhovující poloroviny jsou na Obrázku 2.3 vyznačeny šipkami. Nesmíme také zapomenout na požadavek nezápornosti obou proměnných, čímž dostaneme přípustnou množinu X jako průnik jednotlivých polorovin a prvního kvadrantu. Výsledkem je pak pětiúhelník z Obrázku 2.3.



Obrázek 2.3: Přípustná množina.

Vrstevnice účelové funkce jsou rovnoběžné přímky $-x_1 - x_2 = \alpha$ určené tímž normálovým vektorem, které se pro $\alpha \rightarrow \infty$ „pohybují“ jihozápadním směrem (odpovídajícímu právě onomu normálovému vektoru)⁴⁴, viz Obrázek 2.4.

⁴⁴ Díky rovnoběžnosti nám stačí zakreslit pouze jednu vrstevnici a ostatní pak pouze „zkopírovat“. Např. pro volbu $\alpha = c_1 c_2$ (součin koeficientů účelové funkce) bude odpovídající vrstevnice vždy procházet body $[0, c_1]$ a $[c_2, 0]$.



Obrázek 2.4: Kompletní vizualizace dané úlohy se zvýrazněním normálového vektoru vrstevnic.

Z tohoto obrázku pak snadno vidíme, že řešením dané úlohy je bod C , který odpovídá průsečíku přímek určující první a druhé omezení, tj. vyhovuje dvojici rovnic $x_1 + 2x_2 = 4$ a $4x_1 + 2x_2 = 12$. Snadným výpočtem zjistíme, že $C = [8/3, 2/3]$ a odpovídající hodnota úlohy je $f^* = -10/3$. Jak by se změnilo řešení a Obrázek 2.4, kdyby účelová funkce byla $x_1 + x_2$?

Při méně přesném náčrtku dané úlohy (což je samozřejmě mnohem častější případ) se může stát, že není zcela zřejmé, ve kterém bodě nastává řešení. Obvykle dokážeme identifikovat hranu, na které bude ležet řešení, ale rozhodnutí o konkrétním bodě už může být složitější. V takovém případě je možné si pomoci směrnici vrstevnic, která má vždy hodnotu $-c_1/c_2$, takže odchylka těchto přímek od kladné části osy x_1 je

$$\varphi = \begin{cases} \arctg(-c_1/c_2), & -c_1/c_2 > 0, \\ \pi + \arctg(-c_1/c_2), & -c_1/c_2 < 0. \end{cases}$$

V prvním případě pak bude $\varphi \in (0, \pi/2)$, zatímco ve druhém dostaneme $\varphi \in (\pi/2, \pi)$. Kdyby $c_1 = 0$, pak jsou vrstevnice rovnoběžné s osou x_1 , zatímco pro $c_2 = 0$ budou rovnoběžné s osou x_2 . Podobným způsobem určíme i sklon hraniční přímky vymežující množinu X a jejich porovnáním získáme hledaný bod, přičemž zde je dobré mít na paměti, že větší hodnota podílu $|c_1/c_2|$ znamená větší sklon neboli úhel φ blíže $\pi/2$. Ovšem nelze *a priori* říci, zda větší sklon je „lepší“ nebo „horší“. Vždy totiž záleží na vzájemné poloze příslušné hrany a vrstevnic a také na směru jejich pohybu. Toto si můžeme ilustrovat na následujícím příkladě.

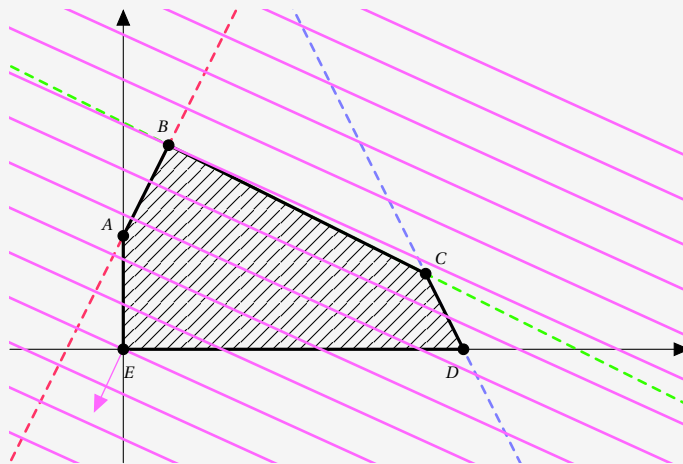
Příklad 2.3.2

Uvažme úlohu

$$-5x_1 - 11x_2 \rightarrow \min$$

$$x_1 + 2x_2 \leq 4 \quad \& \quad 4x_1 + 2x_2 \leq 12 \quad \& \quad -2x_1 + x_2 \leq 1 \quad \& \quad x_1 \geq 0 \quad \& \quad x_2 \geq 0.$$

Jelikož přípustná množina je stejná jako v předchozím příkladě, můžeme využít Obrázek 2.3 a zakreslit do něj vrstevnice odpovídající současné účelové funkci, viz Obrázek 2.5.



Obrázek 2.5: Kompletní vizualizace dané úlohy se zvýrazněním normálového vektoru vrstevnic.

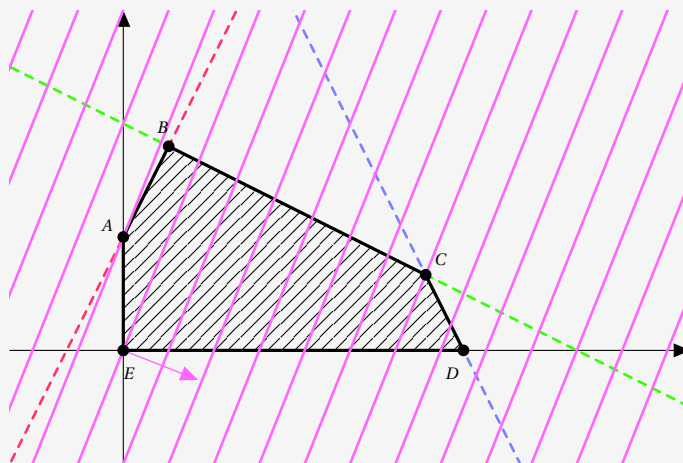
Odtud je vidět, že řešením je bod B . Ovšem při běžném náčrtku to až tak zřejmé být nemusí a můžeme být na pochybách, zda řešením je bod B nebo C (tyto jediné dva potenciální kandidáty bychom měli poznat i při velmi hrubém náčrtku). Protože pro vrstevnice máme $c_1 = -5$ a $c_2 = -11$, bude směrnice těchto přímek rovna $\arctg(-5/11) \approx 155,6^\circ$. Hrana přípustné množiny s body B a C odpovídá přímce z prvního omezení, takže její směrnice je $\arctg(-1/2) \approx 153,3^\circ$. To znamená, že sklon této hrany je o něco málo strmější než vrstevnic (však také $|-5/11| < |-1/2|$), což potvrzuje náš původní závěr ohlední řešení v bodě B .

Uvažme ještě také úlohu

$$5x_1 - 2x_2 \rightarrow \min$$

$$x_1 + 2x_2 \leq 4 \quad \& \quad 4x_1 + 2x_2 \leq 12 \quad \& \quad -2x_1 + x_2 \leq 1 \quad \& \quad x_1 \geq 0 \quad \& \quad x_2 \geq 0.$$

Opět máme tutéž přípustnou množinu, takže dostáváme Obrázek 2.6 níže.

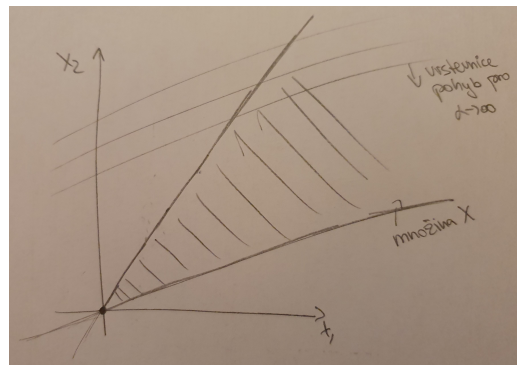


Obrázek 2.6: Kompletní vizualizace dané úlohy se zvýrazněním normálového vektoru vrstevnic.

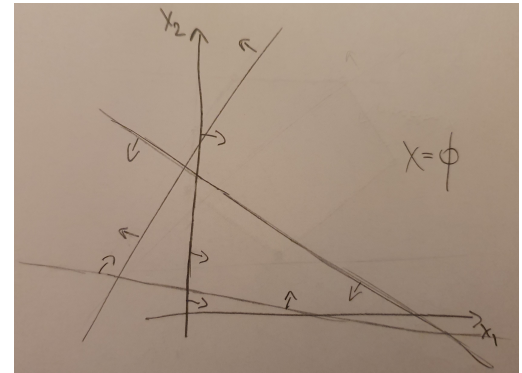
V tomto případě vidíme, že řešením je bod A . Pojd'me si jej potvrdit i pomocí výpočtu směrníc odpovídajících přímek. Bez ohledu na preciznost náčrtku by mělo být zřejmé, že řešením je buď bod A nebo B . Tato hrana odpovídá třetí omezení a směrnice této

přímky je $\arctg(2) \approx 63,4^\circ$. Směrnice vrstevnic je $\arctg(5/2) \approx 68,2^\circ$, takže jejich sklon je o trochu strmější než pro hranu přípustné množiny (však také $5/2 > 2$). Proto řešením úlohy je skutečně bod A .

Všechny předchozí úlohy měly jediné řešení. To samozřejmě neplatí obecně a úplně stejně se může stát, že řešení vůbec neexistuje – a to buď kvůli neohraničenosti účelové funkce nebo kvůli tomu, že $X = \emptyset$, viz Obrázky 2.7.a–2.7.b.

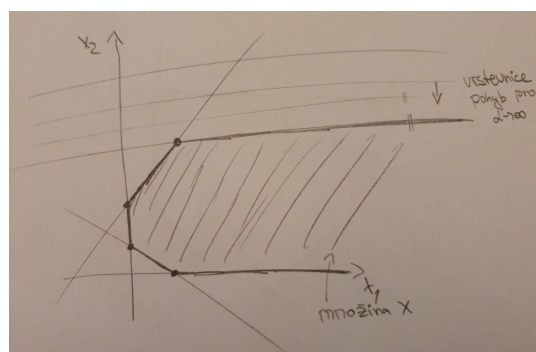


Obrázek 2.7.a: Neohraničená úloha.

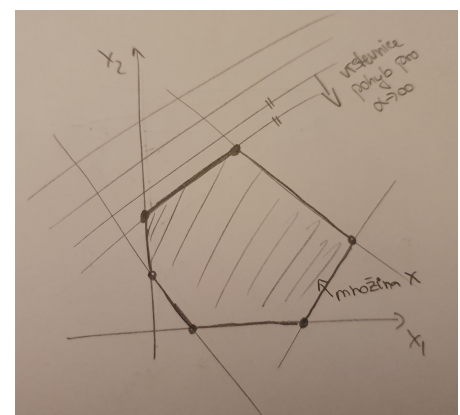


Obrázek 2.7.b: Nepřípustná úloha.

Úloha také může mít nekonečně mnoho řešení. I zde ale mohou nastat dvě principiálně odlišné situace – buď je množina řešení neohraničená (přímka nebo polopřímka) nebo je ohraničená (úsečka), viz Obrázky 2.8.a–2.8.b. K tomuto je ale každopádně nutné, aby normálový vektor vrstevnic a hrany přípustné množiny (tj. jedné z vymežujících přímek) byly lineárně závislé (tj. jeden nějakým nenulovým násobkem druhého).

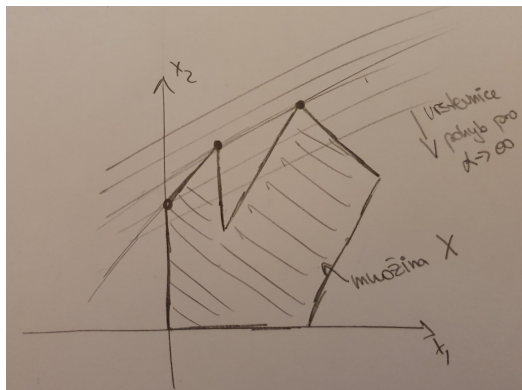


Obrázek 2.8.a: Nekonečně mnoho řešení (polopřímka).



Obrázek 2.8.b: Nekonečně mnoho řešení (úsečka).

V obou předchozích případech byla množina řešení dokonce nepočtená. V této souvislosti se nabízí zcela přirozená otázka: je možné, aby úloha lineárního programování měla pouze konečný počet řešení větší než jedna? Viz např. Obrázek 2.9.



Obrázek 2.9: Úloha lineárního programování může mít právě 2 různá řešení?

At' už řešení bylo jediné nebo jich bylo nekonečně mnoho, jedna velmi důležitá vlastnost všechny tyto případy spojovala. Vidíte ji na obrázcích? Ano, řešení vždy nastalo v nějakém vrcholu mnohoúhelníku (a bylo-li to dokonce ve dvou, pak řešením byly i body ležící na celé hraně spojující tyto body). Náhoda? Mohlo by to být v \mathbb{R}^2 jinak? A v \mathbb{R}^n ? To zjistíme v další části. Ted' se ale podíváme, co je přípustná množina X z (2.2.1) v obecném případě. Tato množina je určena pomocí soustavy rovností a neostrých nerovností daných afinními funkcemi a podmínkami na znaménko některých proměnných. Co to znamená? Každá z rovností v soustavě $A_1x = b_1$, tj. $\langle a_i, x \rangle = b_i$ pro $i = 1, \dots, k$, určuje nadrovinu v \mathbb{R}^n s normálovým vektorem a_i , takže takové soustavě vyhovují body ležící v průniku všech nadrovin. Pokud žádné dvě z nich nejsou rovnoběžné⁴⁵ ani „žádná z nich není lineární kombinací ostatních“⁴⁶ (viz raději také začátek dalšího odstavce), pak s každou rovnostní podmínkou dojde ke snížení dimenze původního prostoru \mathbb{R}^n o jeden stupeň, takže dané k -tici rovností vyhovuje $n - k$ dimenzionální nadrovina (afinní množina). Oproti tomu každá z daných nerovnostních podmínek v soustavě $A_2x \leq b_2$, tj. $\langle a_i, x \rangle \leq b_i$ pro $i = k + 1, \dots, m$, udává tzv. *poloprostor* v \mathbb{R}^n , což je jedna ze dvou částí \mathbb{R}^n ohraničených nadrovinou $\langle a_i, x \rangle = b_i$ s normálovým vektorem a_i . Průnik těchto poloprostorů samozřejmě může být neohraničený (přímka, polopřímka, pás, kužel apod.) nebo ohraničený, v kterémžto případě půjde o nějaký (mnohdy nepravidelný) mnohostěn. Konečně požadavek nezápornosti některých proměnných znamená, že se musíme omezit pouze na část \mathbb{R}^n ohraničenou odpovídajícími osami. Množina X je pak průnikem těchto tří množin, tj. typicky ta část $n - k$ dimenzionální nadroviny ležící v nějakém (ohraničeném/nehrazeném) mnohostěnu. Vrstevnice účelové funkce jsou pro každé $\alpha \in \mathbb{R}$ neprázdné množiny $\{x \in \mathbb{R}^n \mid \langle c, x \rangle = \alpha\}$. Lépe řečeno, jedná se o rovnoběžné nadroviny s normálovým vektorem c , které se pro $\alpha \rightarrow \infty$ „pohybují“ ve směru vektoru c . Tyto vrstevnice vyplní celé \mathbb{R}^n a řešení úlohy (2.2.1) opět znamená nalezení bodu (příp. bodů) z X , který leží na vrstevnici s nejmenší možnou hodnotou α , tj. hledáme vrstevnici na nejnížší možné úrovni, která má ještě neprázdný průnik s množinou X . Takový bod může být jediná nebo jich může být nekonečně (nespočetně) mnoho a nebo nemusí existovat žádný.

Tento náš intuitivní pohled na množinu X můžeme popsat také *konvexním* jazykem, který hraje zcela klíčovou roli v celé oblasti matematického programování a my se jej podrobně naučíme později v Kapitole ???. Jelikož každé omezení je dáno pomocí afinní funkce (a tudíž je určeno nějakou nadrovinou), musí být odpovídající množina konvexní⁴⁷. Díky tomu je množina X (jakožto průnik konvexních množin) nutně také konvexní – ovšem ne jen tak ledajaká. Každou rovnost totiž můžeme nahradit dvojicí nerovností a celkový počet všech omezení je konečný, takže se na

⁴⁵ \triangle Rovnoběžnost obvykle nezahrnuje totožnost (splývání), takže v takovém případě soustava nebude mít řešení.

⁴⁶ V takovém případě buď nadrovina splývá (je totožná) s jinou nebo sice nespývá, ale její průnik s ostatními je stejný jako průnik ostatních mezi sebou, tj. tato nadrovina vlastně žádné nové omezení nepřidává a její případné odebrání nemá žádný vliv na podobu přípustné množiny.

⁴⁷ Konvexní množina je taková, že s libovolnými dvěma body obsahuje i celou úsečku určenou těmito body (např. úsečka, trojúhelník, čtverec, kruh, koule atd.)

⁴⁸ Velmi specifickým polyedrem je i prázdná množina, kterou a priori z našich úvah nevyklučujeme.

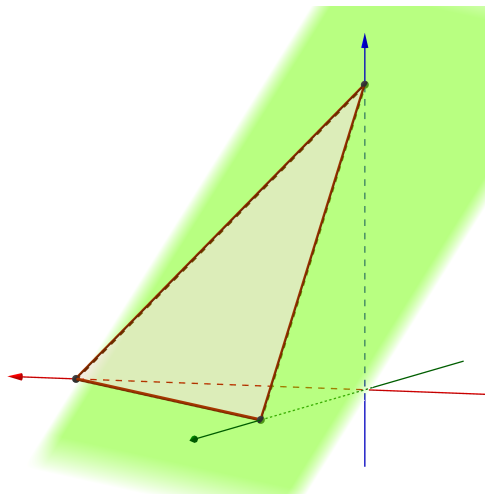
⁴⁹ tj. nejmenší konvexní množina obsahující dané body

⁵⁰ tj. žádná trojice neleží na jedné přímce, žádná čtveřice bodů neleží v jedné rovině, atd.

množinu X můžeme podívat také jako na průnik konečného počtu poloprostorů neboli na *polyedr*⁴⁸. Je-li tento polyedr ohraničený, jedná se o tzv. *polytop* či *simplex*, což je *konvexní obal*⁴⁹ $k + 1$ *afinně nezávislých* bodů⁵⁰, které představují vrcholy množiny X , tj.

$$X = \left\{ x \in \mathbb{R}^n \mid \sum_{i=1}^{k+1} \lambda_i u_i, \lambda_i \in [0, 1] \text{ a } \sum_{i=1}^{k+1} \lambda_i = 1 \right\},$$

kde u_1, \dots, u_{k+1} jsou jednotlivé vrcholy simplexu. Např. množina X na Obrázku 2.10 je tzv. *standardní* (nebo *jednotkový*) *2-simplex* určený vrcholy $u_1 = [1, 0, 0]$, $u_2 = [0, 1, 0]$ a $u_3 = [0, 0, 1]$.



Obrázek 2.10: Množina X pro $A_1 = (1, 1, 1)$ a $b_1 = 1$, zatímco $A_1 = 0$, $b_1 = 0$ (tj. žádné nerovnosti) a $s = 3$ (tj. všechny proměnné jsou nezáporné).

⁵¹ V tomto kontextu rozumíme 0-úhelníkem prázdnou množinu, 1-úhelníkem jednobodovou množinu a 2-úhelníkem úsečku

V \mathbb{R}^2 bude množina X vždy nějaký konvexní ℓ -úhelník, který má v ohraničeném případě $\ell \leq m + n$ hran a vrcholů⁵¹, zatímco v neohraničeném případě bude mít $\ell - 1$ hran a $\ell - 2$ vrcholů.

K čemu to vlastně potřebujeme? Možná k ničemu a později to již nevyužijeme, protože naše možnosti grafického znázornění úloh lineární programování (stejně jako kterýchkoliv jiných) končí u \mathbb{R}^3 . Proto chceme vybudovat algoritmus čistě analyticky (abstraktně) a nezávisle na nějaké vizualizaci. Je navíc asi celkem přirozené očekávat, že praktické úlohy budou obsahovat (mnohem) více než tři proměnné. To by ale bylo až příliš příkré hodnocení. Uvedený geometrický popis má přeci jen velmi důležitou dvojroli

- (i) *intuice*: s jeho pomocí můžeme na základě úloh v \mathbb{R}^2 či \mathbb{R}^3 získat představu o možném řešení obecných úloh, která by se nám mohla hodit při hledání řešení obecných úloh;
- (ii) *etymologie*: v roce 1948 popsal Danzig svůj algoritmus Motzkinovi, který v tom spatřil jakési systematické procházení vrcholů simplexu, díky čemuž získal tento algoritmus svůj název – *simplexový algoritmus*.

Než se pustíme do hlubší analýzy obecných úloh lineárního programování zodpovíme si otázku samotné existence řešení takových úloh. K tomu budeme potřebovat jeden důsledek následující velmi důležitého tvrzení, které si v tuto chvíli uvedeme bez důkazu. Ne že bychom jej nezvládli –

mohli bych si zde pro něj *ad hoc* vybudovat potřebný aparát, avšak bude lepší s tím počkat až do Kapitoly ??, kde si vše uvedeme se všemi podrobnostmi a v širších souvislostech, takže samotný důkaz pak už bude „snadný“. Nicméně si ukážeme alespoň jednoduchý obrázek, který velmi dobře ilustruje podstatu tvrzení v \mathbb{R}^2 .

VĚTA 2.3.3 FARKASOVA– MINKOWSKÉHO

Nechť je dána matice $A \in \mathbb{R}^{m \times n}$ a vektor $b \in \mathbb{R}^m$. Pak buď existuje řešení soustavy

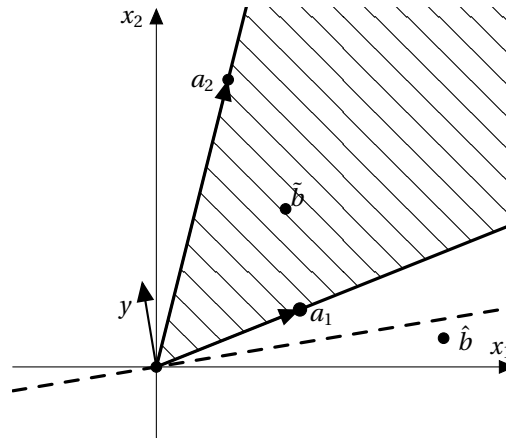
$$Ax = b \quad \& \quad x \geq 0, \quad (2.3.1)$$

nebo existuje řešení

$$A^T y \geq 0 \quad \& \quad \langle y, b \rangle < 0. \quad (2.3.2)$$

⁵² *Kužel* je množina, která s každým svým bodem obsahuje i celou polopřímku vycházející z počátku a procházející tímto bodem.

Uvažme vektory $a_1, a_2 \in \mathbb{R}^2$ tvořící sloupce matice A a bod $b \in \mathbb{R}^2$. Pak splnění soustavy (2.3.1) pro nějaké $x^* \geq 0$ znamená, že bod b patří do *kuželu*⁵² vymezeného polopřímkami vycházejícími z počátku a se směrovými vektory a_1, a_2 (tj. tyto polopřímky prochází odpovídajícími body a_1 a a_2). Naopak, je-li pro nějaké y^* splněna soustava (2.3.2), pak přímka procházející počátkem a s normálovým vektorem y^* *odděluje* kužel a bod b . Je tedy zřejmé, že bod b má právě jednu z těchto vlastností, viz Obrázek 2.11.



Obrázek 2.11: Geometrický pohled na Farkasovu–Minkowského větu pro $b = \tilde{b}$ a $b = \hat{b}$.

Pro nás je ale v tuto chvíli důležitý především následující důsledek Farkasovy–Minkowského věty.

DŮSLEDEK 2.3.4

Nechť je dána matice $A \in \mathbb{R}^{m \times n}$ a vektor $b \in \mathbb{R}^n$. Pak buď existuje řešení soustavy

$$Ax \leq b \quad \& \quad x \in \mathbb{R}^n, \quad (2.3.3)$$

nebo existuje řešení

$$A^T y = 0 \quad \& \quad \langle y, b \rangle < 0 \quad \& \quad y \geq 0. \quad (2.3.4)$$

Důkaz. Definujme matici $\tilde{A} := (A \ b) \in \mathbb{R}^{m \times (n+1)}$ a vektory $\tilde{b} := (0 \ -1)^T \in \mathbb{R}^{n+1}$ pro nulový vektor

$0 \in \mathbb{R}^n$ a také $\tilde{x} := \begin{pmatrix} x \\ \lambda \end{pmatrix} \in \mathbb{R}^{n+1}$ pro $\lambda \in \mathbb{R}$. Pak výše uvedené systémy (2.3.3)–(2.3.4) lze postupně zapsat jako

$$\tilde{A}\tilde{x} \geq 0 \quad \& \quad \langle \tilde{x}, \tilde{b} \rangle < 0 \quad (2.3.5)$$

$$\tilde{A}^\top y = \tilde{b} \quad \& \quad y \geq 0. \quad (2.3.6)$$

Vskutku, je-li $\tilde{x}^* = \begin{pmatrix} x^* \\ \lambda^* \end{pmatrix}$ řešením (2.3.5), pak

$$Ax^* + \lambda^* b \leq 0 \quad \& \quad \lambda^* > 0 \quad \text{neboli} \quad -\frac{1}{\lambda^*} Ax^* \leq b,$$

tj. $-\frac{1}{\lambda^*} x^*$ je řešením úlohy (2.3.3). Naopak, je-li x^* řešením (2.3.3), pak $\tilde{x}^* = \begin{pmatrix} -x^* \\ 1 \end{pmatrix}$ splňuje (2.3.5). Podobně, pokud y^* splňuje (2.3.6), pak $A^\top y^* = 0$, $b^\top y^* = -1$ a $y^* \geq 0$, takže y^* jistě vyhovuje i soustavě (2.3.4). Konečně, je-li y^* řešením (2.3.4), pak pro $-\frac{1}{\langle y^*, b \rangle} y^*$ platí (2.3.6). To tedy znamená (odmyslíme-li si záměnu označení x a y), že úloha (2.3.3) je ekvivalentní s úlohou typu (2.3.2) a problém (2.3.4) s úlohou typu (2.3.1). Pak tvrzení o řešitelnost právě jednoho ze systémů (2.3.3)–(2.3.4) ihned vyplývá z Farkasovy–Minkowského věty 2.3.3. ■

Nyní už máme vše potřebné k tomu, abychom si dokázali větu o řešitelnosti úloh lineárního programování, která říká, že hledání takového řešení může selhat pouze z očividných důvodů.

VĚTA 2.3.5

Je-li v úloze lineárního programování v základním tvaru

$$\langle c, x \rangle \rightarrow \min \quad \& \quad x \in X := \{x \in \mathbb{R}^n \mid Ax \geq b\}$$

pro matici $A \in \mathbb{R}^{m \times n}$ a vektory $b \in \mathbb{R}^m$ a $c \in \mathbb{R}^n$ přípustná množina X neprázdná a funkce $f(x) := \langle c, x \rangle$ zdola ohraničená na X , pak má tato úloha alespoň jedno řešení.

Ještě před samotným důkazem tvrzení jedna velmi důležitá poznámka: Úloha ve Větě 2.3.5 je sice dána v základním tvaru, ale my již víme, že je ekvivalentní s úlohou v kanonickém tvaru, takže totéž tvrzení nutně platí také pro úlohu lineárního programování v kanonickém tvaru...

Důkaz Věty 2.3.5. Necht' $X \neq \emptyset$ a účelová funkce je ohraničená na X , tj. existuje konečné číslo $f^* := \inf_{x \in X} f(x) > -\infty$. Pripusťme, že daná úloha není řešitelná. To znamená, že neexistuje řešení soustavy

$$Ax \geq b \quad \& \quad \langle c, x \rangle \leq f^*,$$

což je ekvivalentní s neexistencí řešení soustavy

$$\begin{pmatrix} -A \\ c^\top \end{pmatrix} x \leq \begin{pmatrix} -b \\ f^* \end{pmatrix}, \quad x \in \mathbb{R}^n.$$

Proto podle Důsledku 2.3.4 musí existovat $y \in \mathbb{R}^m$ a $\lambda \in \mathbb{R}$ taková, že dvojice $\begin{pmatrix} y \\ \lambda \end{pmatrix}$ řeší systém

$$\begin{pmatrix} -A^\top & c \end{pmatrix} \begin{pmatrix} y \\ \lambda \end{pmatrix} = 0 \quad \& \quad \left\langle \begin{pmatrix} y \\ \lambda \end{pmatrix}, \begin{pmatrix} -b \\ f^* \end{pmatrix} \right\rangle < 0 \quad \& \quad \begin{pmatrix} y \\ \lambda \end{pmatrix} \geq 0,$$

což je ekvivalentní se soustavou

$$A^\top y = \lambda c \quad \& \quad \langle y, b \rangle > \lambda f^* \quad \& \quad y \geq 0 \quad \& \quad \lambda \geq 0. \quad (2.3.7)$$

V takovém případě pak pro libovolné $x \in X$ platí

$$\lambda \langle c, x \rangle = \langle \lambda c, x \rangle \stackrel{(2.3.7)}{=} \langle A^T y, x \rangle = \langle y, Ax \rangle \stackrel{x \in X}{\geq} \langle y, b \rangle \stackrel{(2.3.7)}{>} \lambda f^*,$$

z čehož vyplývá, že $\lambda > 0$ (hodnota $\lambda = 0$ není možná vzhledem k ostré nerovnosti) a současně $\langle c, x \rangle \geq \frac{\langle y, b \rangle}{\lambda} > f^*$. Jenže odtud dostáváme spor $f > f$, neboť by vzhledem k libovlnosti $x \in X$ muselo platit

$$f^* = \inf_{x \in X} \langle c, x \rangle \geq \frac{\langle y, b \rangle}{\lambda} > f^*. \quad \blacksquare$$

2.4 Teoretické základy aneb být, či nebýt BPB

Dále se budeme věnovat problému (2.2.1) s množinou X v kanonickém tvaru, tj.

$$\langle c, x \rangle \rightarrow \min \quad \& \quad x \in X := \{x \in \mathbb{R}^n \mid Ax = b \ \& \ x \geq 0\}, \quad (\text{LP})$$

pro danou matici $A \in \mathbb{R}^{m \times n}$ a vektor $b \in \mathbb{R}^m$.

Samozřejmě v úloze (LP) má smysl omezit se pouze na situaci, kdy soustava $Ax = b$ má alespoň jedno řešení, protože v opačném případě je nutně $X = \emptyset$ a problém (LP) je tzv. *nepřípustný*. Jelikož můžeme bez jakékoli újmy na obecnosti předpokládat, že žádná z rovnic v soustavě $Ax = b$ není nadbytečná, pak podle *Rouchéovy–Capelliho věty*⁵³ je nutně $\text{rank } A = m \leq n$. Ovšem zajímavý je především případ $m < n$, neboť pro $m = n$ je soustava $Ax = b$ jednoznačně řešitelná, takže množina X bude buď zase prázdná (pokud ono jediné řešení nesplňuje druhou podmínku $x \geq 0$) nebo bude jednobodová (a tento bod bude automaticky naším hledaným řešením daného problému lineárního programování). Nicméně my nebudeme v našem výkladu explicitně vyžadovat ostrou nerovnost mezi m a n , neboť navzdory jednoduchému vzorci $x = A^{-1}b$ může být nalezení jediného přípustného bodu výpočetně velmi náročné, např. je-li $m = n = 10^6$.⁵⁴ Navíc je potřeba mít na paměti, že úloha může být nepřipustná i pro $m < n$, protože prvky množiny X musí splňovat také podmínku nezápornosti, viz Obrázek 2.12 a 2.13.

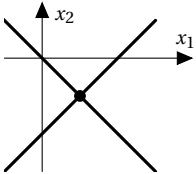
Pro náš algoritmus budeme ještě potřebovat, aby složky vektoru b byly nezáporné, tj. $b \geq 0$. Toho ale lze velmi snadno dosáhnout pouhým vynásobením odpovídajících řádků soustavy $Ax = b$ číslem -1 . Celkem tedy v úloze (LP) můžeme bez újmy na obecnosti předpokládat, že

- *sloupce* matice A jsou $a_1, \dots, a_n \in \mathbb{R}^m$, tj. $A = (a_1, a_2, \dots, a_n)$ (pouze označení);
- $m \leq n$ a $\text{rank } A = m$ (tj. řádky matice A jsou lineárně nezávislé);
- $b \geq 0$ (pouze technický požadavek).

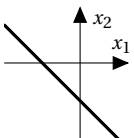
A co je množina X v tomto případě? Je to konvexní podmnožina \mathbb{R}^n vytvořená průnikem kladného orthantu v \mathbb{R}^2 (kvadrantu v \mathbb{R}^2 a oktantu v \mathbb{R}^3) určeného podmínkou $x \geq 0$ a m -tice nadrovin určených každou z rovnic v soustavě $Ax = b$. Tato podmnožina může být ohraničená (polytop/simplex) jako na Obrázku 2.10 nebo neohraničená jako na Obrázku 2.14 níže (ta horní hrana je pouze „virtuální“).

⁵³ (mnohdy nepřesně označované jako *Frobeniova věta*): soustava $Ax = b$ má alespoň jedno řešení právě tehdy, když hodnota matice soustavy je stejná jako hodnota rozšířené matice soustavy, tj. $\text{rank } A = \text{rank}(A \mid b)$. Je-li dokonce $\text{rank } A = n$, pak je řešení jediné, zatímco pro $\text{rank } A < n$ je řešení nekonečně mnoho a tvoří afinní podprostor dimenze $n - \text{rank } A$.

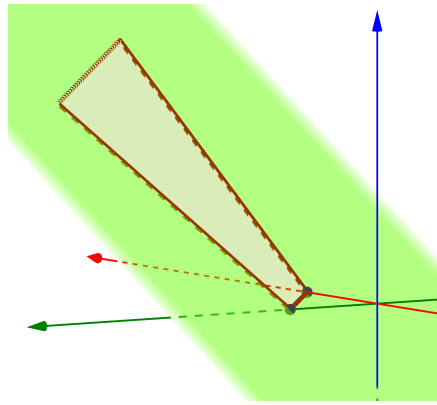
⁵⁴ takové úlohy jsou zcela běžné v počítačové tomografii, a pokud matice A nemá nějakou velmi speciální strukturu (např. velké množství nulových prvků), pak je nalezení řešení prakticky nemožné.



Obrázek 2.12: Soustava $x_1 - x_2 = 2$ a $x_1 + x_2 = 0$ má jediné řešení $x^* = [1, -1]$, které ovšem nesplňuje podmínku nezápornosti, tj. $X = \emptyset$



Obrázek 2.13: Soustava $x_1 + x_2 = 1$ má nekonečně mnoho řešení, ovšem žádná z nich nesplňuje podmínku nezápornosti, tj. $X = \emptyset$

Obrázek 2.14: Množina X z (LP) pro $A = (1, 1, -1)$ a $b = 3$.

K tomu ale budeme potřebovat některé další teoretické poznatky o množině X a účelové funkci.

DEFINICE 2.4.1

Pro libovolný bod $x = [x_1, \dots, x_n] \in \mathbb{R}^n$ definujeme jeho *nosič* jako množinu

$$J(x) := \{j \in \{1, \dots, n\} \mid x_j > 0\},$$

která obsahuje indexy nenulových souřadnic bodu $x \geq 0$.

Je-li navíc $x \in X$, pak jej nazveme *bazickým přípustným bodem* (dále obvykle jen *BPB*) úlohy (LP), jestliže vektory $\{a_j \mid j \in J(x)\}$ jsou lineárně nezávislé.

Lze se setkat také s terminologií

Příklad 2.4.2

Uvažme množinu X určenou maticí $A = \begin{pmatrix} 1 & 5 & 3 & 4 & 6 \\ 0 & 1 & 3 & 5 & 6 \end{pmatrix} \in \mathbb{R}^{2 \times 5}$, která očividně splňuje podmínku $\text{rank } A = 2$, a vektorem $b = \begin{pmatrix} 14 \\ 7 \end{pmatrix} \in \mathbb{R}^2$. Pak například bod $x = [0, 2, 0, 1, 0] \in X$ je BPB, neboť $J(x) = \{2, 4\}$ a odpovídající vektory $a_2 = \begin{pmatrix} 5 \\ 1 \end{pmatrix}$, $a_4 = \begin{pmatrix} 4 \\ 5 \end{pmatrix}$ jsou lineárně nezávislé. Na druhou stranu volbou $b = \begin{pmatrix} 27 \\ 27 \end{pmatrix}$ bod $\bar{x} = [0, 0, 1, 0, 4] \in X$ není BPB, neboť $J(\bar{x}) = \{3, 5\}$ a odpovídající vektory $a_3 = \begin{pmatrix} 3 \\ 3 \end{pmatrix}$, $a_5 = \begin{pmatrix} 6 \\ 6 \end{pmatrix}$ jsou zjevně lineárně závislé.

Jak ale vlastně BPB vypadají? A jak je najít? To jsou klíčové otázky, na které se nyní pokusíme najít odpověď. Nejdříve ukážeme „geometrický“ popis BPB. Již jsme se zmínili, že každá konvexní množina obsahuje všechny úsečky určené body z této množiny, tj. pro libovolné $x, x_2 \in X$ a $\lambda \in [0, 1]$ máme $\lambda x_1 + (1 - \lambda) x_2 \in X$. To ale znamená, že (s výjimkou prázdné a jednobodové množiny) většina bodů v množině X leží „uvnitř“ nějaké úsečky s krajními body z X . Ovšem to nejsou všechny – spolu s těmito body tam jsou body, pro které žádná taková úsečka neexistuje, tj. nelze ani „o kousek“ prodloužit žádnou úsečku vedoucí z množiny X za tento bod, aniž by toto prodloužení neopustilo množinu X . Toto jsou pro nás extrémně důležité body.

DEFINICE 2.4.2A

Nechť množina $Y \subseteq \mathbb{R}^n$ je konvexní. Pak body, které nelze vyjádřit ve tvaru $\lambda \bar{y} + (1 - \lambda) \bar{y}$ pro nějaká $\bar{y}, \bar{y} \in Y$ splňující $\bar{y} \neq \bar{y}$ a nějaké $\lambda \in (0, 1)$ nazýváme *extrémními* (nebo *krajními*) body množiny Y .

Jedním z typických příkladů konvexní množiny je n -úhelník. Jeho extrémní body pak jsou jednotlivé vrcholy, viz Obrázek 2.15. Ovšem extrémní body mohou vypadat i trochu jinak. Uvažme např. kruh. Má extrémní body? A jaké to jsou? Ano, je to celý obvod kruhu, tj. hraniční kružnice.

VĚTA 2.4.3

Bod $x \in X$ je BPB právě tehdy, když je extrémním bodem množiny X .

Důkaz. „ \implies “ Nejdříve předpokládejme, že x je BPB, ale není extrémním bodem množiny X , tj. existují $\tilde{x}, \hat{x} \in X$ a $\lambda \in (0, 1)$ taková, že $\tilde{x} \neq \hat{x}$ a $x = \lambda \tilde{x} + (1 - \lambda) \hat{x}$. Pak nutně $\tilde{x}_j \neq \hat{x}_j$ pro $j \notin J(x)$, protože pro tyto indexy máme $0 = x_j = \lambda \tilde{x}_j + (1 - \lambda) \hat{x}_j$ a na pravé straně jsou pouze nezáporná čísla. Proto platí

$$b = \sum_{j=1}^n \tilde{x}_j a_j = \sum_{j \in J(x)} \tilde{x}_j a_j = \sum_{j \in J(x)} \hat{x}_j a_j, \quad \text{tj.} \quad 0 = \sum_{j \in J(x)} (\tilde{x}_j - \hat{x}_j) a_j. \quad (2.4.1)$$

Jenže x je BPB, takže vektory $\{a_j \mid j \in J(x)\}$ lineárně nezávislé, a tudíž rovnost (2.4.1) je splněna tehdy a jenom tehdy, když $\tilde{x}_j - \hat{x}_j = 0$, tj. $\tilde{x}_j = \hat{x}_j$. To ale znamená, že $\tilde{x} = \hat{x}$, což je spor ℓ .

„ \Leftarrow “ Důkaz opačného směru se provede podobně. Necht' $x \in X$ je extrémním bodem množiny X , ale současně není BPB, tj. vektory $\{a_j \mid j \in J(x)\}$ jsou lineárně závislé. Pak tedy existují čísla $\{c_j \mid j \in J(x)\}$ taková, že alespoň jedno je nenulové a $\sum_{j \in J(x)} c_j a_j = 0$. Položme $c_j := 0$ pro $j \notin J(x)$ a $c := (c_1, \dots, c_n)^\top$. Pak platí $A c = \sum_{j=1}^n c_j a_j = 0$, a tedy $A(x \pm \alpha c) = Ax \pm \alpha A c = b$ pro libovolné $\alpha \geq 0$. Navíc pro dostatečně malá $\alpha \geq 0$ platí také $x \pm \alpha c \geq 0$ (perturbujeme pouze kladné souřadnice bodu x ; kdyby $x = 0$, pak nutně $c = 0$ a vše je triviální), tj. $x \pm \alpha c \in X$. Jenže současně platí

$$x = \frac{1}{2}(x + \alpha c) + \frac{1}{2}(x - \alpha c),$$

což je spor s extrémností bodu x , neboť jsem jej vyjádřili jako konvexní kombinaci dvou různých (vyjma $c = 0$) bodů z X . ■

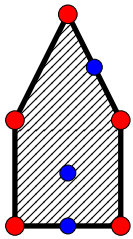
Než si ukážeme jakou roli hrají BPB v řešení úlohy (LP), podíváme na jejich existenci, počet a jak by se daly najít.

VĚTA 2.4.4

Necht' $X \neq \emptyset$. Pak množina všech BPB množiny $X \neq \emptyset$ je neprázdná a navíc pro libovolné $x \in X$ existuje BPB \tilde{x} takový, že $J(\tilde{x}) \subseteq J(x)$.

Důkaz. Vzhledem k předpokladu $X \neq \emptyset$ je první část *de facto* pouze přímým důsledkem druhé části, která zaručuje existenci alespoň jednoho BPB. Dokážeme tedy pouze druhou část tvrzení. Necht' $x \in X$ je libovolné. Pak $b = \sum_{j=1}^n x_j a_j$ a naším cílem je najít BPB $\tilde{x} \in X$ s nejvýše stejným počtem nenulových prvků, tj. pro $\tilde{x} \geq 0$ takové, že $b = \sum_{j \in J(x)} \tilde{x}_j a_j$, $J(\tilde{x}) \subseteq J(x)$ a vektory $\{a_j \mid j \in J(\tilde{x})\}$ jsou lineárně nezávislé. Je-li $b = 0$, pak všechny tyto požadavky splňuje $\tilde{x} := 0$, neboť $A\tilde{x} = b = 0$ a $J(\tilde{x}) = \emptyset$. Pro dále uvažujme pouze $b \neq 0$:

- (i) Jsou-li vektory $\{a_j \mid j \in J(x)\}$ lineárně nezávislé, pak bod x už je BPB, a volba $\tilde{x} := x$ bude jistě splňovat všechny požadavky.



Obrázek 2.15: Konvexní množina X její všechny extrémní body (červené) a některé „neextrémní“ body (modré).

- (ii) Jsou-li vektory $\{a_j \mid j \in J(x)\}$ lineárně závislé, pak existují čísla $\{\alpha_j \mid j \in J(x)\}$ taková, že ne všechna jsou nulová a $\sum_{j \in J(x)} \alpha_j a_j = 0$, takže pro libovolné $\gamma \in \mathbb{R}$ platí

$$b = \sum_{j \in J(x)} (x_j - \gamma \alpha_j) a_j \quad (2.4.2)$$

Navíc bez újmy na obecnosti můžeme předpokládat, že alespoň pro jedno $j \in J(x)$ je $\alpha_j > 0$ (v opačném případě stačí u všech α_j změnit znaménko), takže jistě existuje číslo

$$\gamma^* := \min \left\{ \frac{x_j}{\alpha_j} \mid \alpha_j > 0, j \in J(x) \right\} > 0.$$

Potom jistě $x_j - \gamma^* \alpha_j \geq 0$, přičemž alespoň pro jedno $j \in J(x)$ nastane rovnost. Proto bod \bar{x} se složkami $\bar{x}_j := x_j - \gamma^* \alpha_j$ pro $j \in J(x)$ a $\bar{x}_j := 0$ pro $j \notin J(x)$ je díky volbě γ^* a rovnosti (2.4.2) aplikované pro γ^* jistě přípustný (tj. $\bar{x} \in X$) a splňuje $|J(\bar{x})| \leq |J(x)| - 1$. Je-li bod \bar{x} už BPB, jsme hotovi. V opačném případě můžeme proces opakovat tak dlouho, dokud

- všechny složky \bar{x} nebudou nulové – to by ale vzhledem k jeho přípustnosti bylo možné pouze v případě $b = 0$, což už jsme vyloučili $\cancel{!}$;
- vektory $\{a_j \mid j \in J(\bar{x})\}$ budou lineárně nezávislé a v takovém případě je důkaz hotov, neboť máme BPB bod splňující dokonce $J(\bar{x}) \subset J(x)$. ■

Už tedy víme, že pro libovolnou přípustnou množinu $X \neq \emptyset$ vždy existuje alespoň jeden BPB. Ale kolik jich je přesně? Kolik vlastně může mít daný polyedr/simplex extrémních bodů? Jde to vůbec *a priori* spočítat? Toto je jedna z klíčových otázek samotného simplexového algoritmu a k její odpovědi budeme ještě jiný (ekvivalentní) popis extrémních bodů z následujícího lemmatu. V něm využijeme značení $A_B := (a_j)_{j \in B}$ pro matici vzniklou z matice A vypuštěním sloupců s indexy mimo danou indexovou množinu B ⁵⁵. Podobně budeme používat značení x_B pro vektor vytvořený z x tak, že v něm vynecháme složky na pozicích s indexy $\{1, 2, \dots, n\} \setminus B$.

⁵⁵ Při takovéto interpretaci musí být index v množině B seřazený vzestupně. Pro libovolné uspořádání prvků v B , pak A_B znamená matici, ve které první sloupec odpovídá sloupci matice A s indexem rovným prvnímu prvku v B , druhý sloupec je určen druhým indexem v B atd.

LEMMA 2.4.5

Bod x je BPB množiny X právě tehdy, když existuje m -prvková množina $B \subseteq \{1, 2, \dots, n\}$ taková, že

- čtvercová $m \times m$ matice A_B je regulární, tj. sloupce $\{a_j \mid j \in B\}$ jsou lineárně nezávislé;
- $x_j = 0$ pro $j \in \{1, 2, \dots, n\} \setminus B$.

Důkaz. „ \Leftarrow “ Je-li $x \in X$ a platí-li obě podmínky, pak $J(x) \subseteq B$, a tudíž vektory a_j jsou lineárně nezávislé i pro $j \in J(x)$, tj. x je BPB.

„ \Rightarrow “ Necht' $x \in X$ je BPB. Je-li $|J(x)| = m$, pak indexová množina $B := J(x)$ jistě splňuje obě dané podmínky. Je-li $|J(x)| < m$, pak množinu B vytvoříme postupným přidáváním $m - |J(x)|$ indexů do $J(x)$ tak, že všechny vektory $\{a_j \mid j \in B\}$ budou lineárně nezávislé. To provedeme následujícím postupem. Začneme volbou $B := J(x)$. Jelikož $\text{rank } A = m \leq n$ (tj. existuje m lineárně nezávislých sloupců matice A), musí existovat (alespoň jeden) sloupec a_i , který nelze získat jako vhodnou lineární kombinaci sloupců $\{a_j \mid j \in B\}$. Tento index i přidáme do množiny B a celý krok zopakujeme s touto novou množinou B . Ve chvíli, kdy již takový krok není možný, tj. všechny sloupce matice A již jsou v lineárním obalu množiny $\{a_j \mid j \in B\}$, pak tyto sloupce tvoří bázi lineárního obalu množiny $\{a_j \mid j \in \{1, \dots, n\}\}$, takže nutně $|B| = m$, jak bylo potřeba. ■

Tato charakterizace nám dává velmi jednoduchý (ale pracný) návod na hledání *bazických bodů*: stačí vzít indexovou množinu $B \subseteq \{1, 2, \dots, n\}$ určující regulární matici A_B a vypočítat $x_B := A_B^{-1}b$, což po doplnění nulovými hodnotami pro indexy $z \in \{1, 2, \dots, n\} \setminus B$ dává bod $x \in \mathbb{R}^n$ vyhovující podmínkám z Lemmatu 2.4.5. Je-li navíc $x \geq 0$, pak máme dokonce BPB $x \in X$. Ovšem může se stát, že tato dodatečná podmínka není splněna, takže pro danou indexovou množinu neexistuje žádný BPB. Každopádně m -prvkových podmnožin n -prvkové množiny existuje pouze konečný počet rovný binomickému číslu $\binom{n}{m} = \frac{n!}{m!(n-m)!}$, dává horní odhad počtu BPB množiny X , tj. je jich *pouze konečně mnoho*.

VĚTA 2.4.6

Množina všech BPB je jednoznačně určena indexovými množinami B z Lemma 2.4.5, tj. pro každou m -prvkovou množinu $B \subseteq \{1, 2, \dots, n\}$ určující regulární matici A_B existuje nejvýše jeden BPB $x \in X$.

Důkaz. Necht' $B \subseteq \{1, 2, \dots, n\}$ je m -prvková množina, pro kterou A_B je regulární $m \times m$ matice. Ukážeme, že existuje nejvýše jeden BPB $x \in X$ určený právě touto množinou B . Takový bod x musí splňovat $x_j = 0$ pro $j \in \{1, 2, \dots, n\} \setminus B$ a současně

$$b = Ax = \sum_{j \in B} x_j a_j + \sum_{i \in \{1, 2, \dots, n\} \setminus B} x_i a_i = \sum_{j \in B} x_j a_j = A_B x_B. \quad (2.4.3)$$

Ovšem tato soustava má díky regulárnosti matice A_B jediné řešení $x_B = A_B^{-1}b$, čímž je vektor x jednoznačně určen. Pokud navíc $x_B \geq 0$, pak toto x je jediným BPB příslušným indexové množině B , zatímco v opačném případě plyne, že BPB pro B neexistuje. ■



Ovšem pozor: toto tvrzení o jednoznačném propojení mezi BPB v X a indexovými množinami B neříká vůbec nic o *jedno-jednoznačnosti* příslušného BPB $x \in X$, tj. různým indexovým množinám B může příslušet jeden a ten stejný BPB. Nicméně to poukazuje na jistou *bazickou* vlastnost množiny B , která je umocněna i tím, že vektory $\{a_j \mid j \in B\}$ tvoří bázi \mathbb{R}^m .

DEFINICE 2.4.7

Množinu B popsanou v Lemmatu 2.4.5 nazveme *bází* BPB $x \in X$. V takovém případě hovoříme o tom, že indexy $j \in B$ jsou v *bázi bodu* x a indexy $j \in \{1, 2, \dots, n\} \setminus B$ jsou *mimo bázi bodu* x .

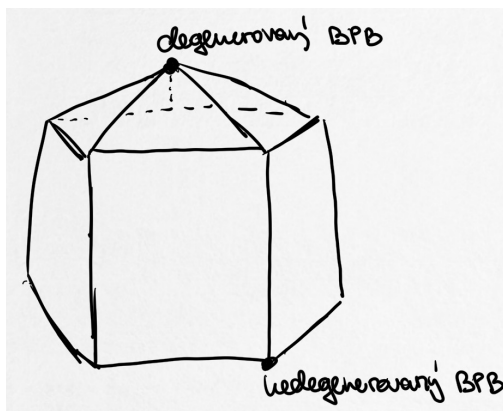
BPB $x \in X$ nazveme *nedegenerovaný*, jestliže $x_j > 0$ pro všechna $j \in B$, zatímco v případě existence alespoň jednoho indexu $j \in B$ s vlastností $x_j = 0$ hovoříme o *degenerovaném* bodě. Jsou-li všechny BPB množiny X nedegenerované, řekneme, že úloha (LP) je *nedegenerovaná*. V opačném případě nazveme úlohu (LP) *degenerovanou*.



BPB $x \in X$ je tedy nedegenerovaný právě tehdy, když $|J(x)| = m$ neboli $J(x) = B$. Navíc z definice ihned vyplývá, že nedegenerovaný BPB $x \in X$ je vždy jednoznačně určen množinou B (různé báze nemohou dát stejný nedegenerovaný BPB vzhledem k druhé podmínce v Lemmatu 2.4.5), tj. množina $\{a_j \mid j \in B\}$ tvoří v jistém smyslu „bázi“ BPB $x \in X$ – samozřejmě to není totéž báze v lineární algebře, neboť zde pro různé BPB potřebujeme různé množiny B . Je-li BPB $x \in X$ degenerovaný, pak jej *může* být možné získat pomocí dvou (a více) různých množin B (to ale záleží na lineární /ne-/závislosti vektorů $\{a_j \mid j \in \{1, 2, \dots, n\} \setminus B\}$ a vektoru a_i odpovídajícího indexu $i \in B$ s $x_i = 0$). Každopádně platí, že úloha (LP) je degenerovaná právě tehdy, když vektor b lze

⁵⁶ Neboli úloha (LP) je nedegenerovaná právě tehdy, když vektor b lze vyjádřit jako lineární kombinaci s nenulovými koeficienty pomocí m sloupců matice A .

vyjádřit jako lineární kombinaci s nezápornými koeficienty pomocí nejvýše $m - 1$ sloupců matice A ⁵⁶. Vskutku, je-li úloha degenerovaná, pak existuje $x \in X$ a báze B tak, že $b = A_B x_B$ a $x_j = 0$ pro alespoň jedno $j \in B$, tj. stačí nám nejvýše $m - 1$ sloupců matice A . Naopak, stačí-li nejvýše $m - 1$ sloupců matice A , pak doplněním indexů na m -prvkovou množinu B při zachování lineární nezávislosti a nulovosti složek x , dostaneme degenerovaný BPB. To mimo jiné znamená, že degenerovaný bod můžeme charakterizovat také tak, že vznikl průnikem více než n nadrovin, viz také Obrázek 2.16.



Obrázek 2.16: Množina X určená podmínkami $Ax \leq b$ s degenerovaným bodem, který není důsledkem nadbytečného omezení.

Jestliže matice A obsahuje všechny sloupce $m \times m$ jednotkové matice (v libovolném pořadí), pak nalezení alespoň jednoho BPB je celkem snadné: vezmeme bázi B tvořenou indexy těchto sloupců $\{e_i = (\underbrace{0, \dots, 0}_{i-1}, 1, 0, \dots, 0)^T \mid i \in \{1, \dots, m\}\}$ a položíme $x_j = 0$ pro $j \in \{1, 2, \dots, m\} \setminus B$, zatímco pro $j \in B$ s odpovídajícím sloupcem e_i pro nějaké $i \in \{1, \dots, m\}$ bereme x_j jako i -tou složku vektoru b .

Příklad 2.4.8

Určeme BPB s bázi $B = \{3, 4\}$, $B = \{1, 4\}$ a $B = \{1, 2\}$ pro množinu X vymezenou podmínkami

$$\begin{aligned} 2x_1 + x_2 + x_3 = 1 \quad & \& \quad x_1 + 3x_2 + x_4 = 2 \\ x_1 \geq 0 \quad & \& \quad x_2 \geq 0 \quad & \& \quad x_3 \geq 0 \quad & \& \quad x_4 \geq 0. \end{aligned}$$

V tomto případě máme

$$A = \begin{pmatrix} 2 & 1 & 1 & 0 \\ 1 & 3 & 0 & 1 \end{pmatrix} \quad \& \quad b = \begin{pmatrix} 1 \\ 2 \end{pmatrix}.$$

Pro bázi $B = \{3, 4\}$ máme $A_B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, takže $x_B = A_B^{-1} b = b$, tj. odpovídají BPB je

$$x = [0, 0, 1, 2].$$

Pro bázi $B = \{1, 4\}$ máme $A_B = \begin{pmatrix} 2 & 0 \\ 1 & 1 \end{pmatrix}$, takže⁵⁷ $A_B^{-1} = \frac{1}{2} \begin{pmatrix} 1 & 0 \\ -1 & 2 \end{pmatrix}$. Proto $x_B = A_B^{-1} b = \begin{pmatrix} 1/2 \\ 3/2 \end{pmatrix}$, tj. odpovídají BPB je

$$x = [1/2, 0, 0, 3/2].$$

⁵⁷ Zde pouze stačí využít známého vzorce $\begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} = \frac{1}{ad-bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$.

Konečně pro $B = \{1, 2\}$ máme $A_B = \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix}$, takže $A_B^{-1} = \frac{1}{5} \begin{pmatrix} 3 & -1 \\ -1 & 2 \end{pmatrix}$. Proto $x_B = A_B^{-1} b = \begin{pmatrix} 1/5 \\ 3/5 \end{pmatrix}$, tj. odpovídají BPB je

$$x = [1/5, 3/5, 0, 0].$$

Kdybychom upravili druhou rovnost v zadání tak, že by matice A tentokrát byla

$$A = \begin{pmatrix} 2 & 1 & 1 & 0 \\ 1 & -3 & 0 & 1 \end{pmatrix},$$

pak bychom pro $B = \{3, 4\}$ dostali tentýž BPB, zatímco pro $B = \{1, 2\}$ bychom měli $A_B = \begin{pmatrix} 2 & 1 \\ 1 & -3 \end{pmatrix}$ a $x_B = \begin{pmatrix} 5/7 \\ -3/7 \end{pmatrix}$, takže odpovídající bod je

$$x = [5/7, -3/7, 0, 0],$$

což je sice bazický bod, ovšem není přípustný! Tj. pro tuto bázi neexistuje BPB. V obou případech byly dané množiny nedegenerované, neboť vektor b nebyl násobkem žádného ze sloupců matice A (tj. je jej možné získat jako lineární kombinaci nejméně dvou sloupců matice A). Příkladem degenerované množiny je třeba

$$\begin{aligned} x_1 - 2x_2 - x_3 + 9x_4 + x_5 &= 0 & x_1 - 3x_2 - x_3 + 3x_4 + x_6 &= 0 & x_3 + x_7 &= 1 \\ x_1 \geq 0 & \& x_2 \geq 0 & \& x_3 \geq 0 & \& x_4 \geq 0 & \& x_5 \geq 0 & \& x_6 \geq 0 & \& x_7 \geq 0. \end{aligned}$$

V takovém případě máme

$$A = \begin{pmatrix} 1 & -2 & -1 & 9 & 1 & 0 & 0 \\ 1 & -3 & -1 & 3 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} \quad \& \quad b = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

Pro takovou množinu je totiž BPB např. $x = [0, \dots, 0, 1]$, pro který platí $J(x) = \{7\}$, zatímco $B = \{5, 6, 7\}$. Je také zřejmé, že pro vektor b nepotřebujeme tři sloupce matice A – stačí nám stačí pouze jeden (ten poslední).

Jistě jste si již všimli k tomu, zda nějaký bod je či není BPB, jsme vůbec nepotřebovali účelovou funkci úlohy (LP), ale šlo jen o „geometrii“ samotné množiny X . Tak proč jsme tomu tak podrobně věnovali? Protože ve skutečnosti existuje velmi úzká souvislost mezi BPB a řešením úlohy (LP), jak nám ukazuje poslední tvrzení tohoto odstavce.

VĚTA 2.4.9

Jestliže úloha (LP) je řešitelná, pak existuje BPB, který je řešením.



Pozor: tato věta rozhodně neříká, že každé řešení úlohy (LP) je BPB či extrémní bod. Kdyby např. řešením byla celá hrana nějakého polyedru/simplexu, pak BPB jsou pouze její koncové body (vrcholy), ovšem ostatní body již BPB nejsou. Ovšem je-li úloha jednoznačně řešitelná, pak tím řešením nutně musí být nějaký BPB. Odtud a z Věty 2.4.6 dostáváme konečně-krokový (ačkoli zcela nepraktický) algoritmus (využívající opravdu velmi hrubé síly) pro řešení úlohy (LP): stačí uvážit všechny m -prvkové podmnožiny $B \subseteq \{1, 2, \dots, n\}$, pro každou z nich najít příslušný BPB (pokud existuje) a spočítat hodnotu účelové funkce v těchto bodech. Např. $n = 2m$ musíme uvá-

žit $\binom{2^m}{m} \approx 4^m$ podmnožin. Tato hodnota pro rostoucí m roste exponenciálně (tj. s každým zvýšením m o 1 dojde ke zčtyřnásobení počtu množin B), což je z praktického hlediska nepřijatelné. Navíc tento „algoritmus“ dává správný výsledek pouze v případě řešitelné úlohy (LP), protože případná neohraničenost dané úlohy na jeho výpočtu nic nemění. My ale směřujeme k mnohem efektivnějšímu algoritmu, ve kterém není procházení BPB náhodné a který funguje správně i pro neohraničené úlohy (tj. v průběhu výpočtu indikuje neohraničenost účelové funkce na X).

Důkaz Věty 2.4.9. Důkaz si rozdělíme na několik kroků, které se v komplikovanější podobě objeví také v simplexové metodě. Proto můžeme tento důkaz považovat za jistou přípravu. Připusťme, že platí následující tvrzení

$$\left. \begin{array}{l} \text{Je-li účelová funkce úlohy (LP) zdola ohraničená, pak pro každé } x_0 \in X \text{ existuje BPB} \\ \bar{x} \in X \text{ se stejnou nebo menší funkční hodnotou účelové funkce, tj. } c^\top \bar{x} \leq c^\top x_0. \end{array} \right\} (*)$$

Potom z konečného počtu BPB plyne, že v některém z nich musí účelová funkce nabýt minimální hodnoty, a tudíž být řešením úlohy (LP).

Musíme tedy dokázat tvrzení (*). Uvažme proto libovolné $x_0 \in X$. Mezi všemi $x \in X$ takovými, že $c^\top x \leq c^\top x_0$ vybereme to, které má největší možný počet nulových složek, a označíme jej jako \bar{x} (tento bod samozřejmě nemusí být určen jednoznačně). Jsou-li vektory $\{a_j \mid j \in J(\bar{x})\}$ lineárně nezávislé, jsme hotovi. V opačném případě musí pro matici $\bar{A} := (a_j)_{j \in J(\bar{x})}$ existovat vektor $\bar{v} \in \mathbb{R}^{|J(\bar{x})|} \setminus \{0\}$ takový, že $\bar{A}\bar{v} = 0$. Doplníme vektor v nulovými hodnotami na pozicích $\{1, 2, \dots, n\} \setminus J(\bar{x})$ a výsledek jako $v \in \mathbb{R}^n$. Pak $Av = \bar{A}\bar{v} = 0$ a předpokládáme, že v navíc splňuje následující dvě podmínky, jejichž platnost si za chvíli ověříme v poslední části důkazu:

- (i) $c^\top v \leq 0$;
- (ii) existuje $j \in J(\bar{x})$ takové, že $v_j < 0$.

Pro $t \geq 0$ definujme vektor $x(t) := \bar{x} + tv$. Pak platí $Ax(t) = A\bar{x} + tAv = A\bar{x} = b$ pro všechna $t \geq 0$, protože $\bar{x} \in X$. Navíc $x(0) = \bar{x}$ má všechny složky s indexy $j \in J(\bar{x})$ kladné a ostatní nulové. Pro j -tou složku vektoru $x(t)$ máme $x_j(t) = \bar{x}_j + tv_j$, což v případě platnosti (ii) dává $x_j(t) < 0$ pro dostatečně velké $t > 0$. Začneme-li s $t = 0$ a necháme-li t růst, pak hodnoty $x_j(t)$ s indexem odpovídajícím $v_j < 0$ jsou klesající a v jistém okamžiku t_1 dosáhnou nulové hodnoty. V tuto chvíli má ale $x(t_1)$ stále ještě nezáporné hodnoty, takže je přípustné a současně má o jednu nulovou složku více než \bar{x} . Ovšem $x(t_1)$ také splňuje $c^\top x(t_1) = c^\top \bar{x} + t_1 c^\top v \leq c^\top x_0$, čímž dostáváme spor s předpokladem, že \bar{x} obsahuje největší možný počet nulových složek mezi všemi $x \in X$ splňujícími $c^\top x \leq c^\top x_0$.

Nyní již jen zbývá dokázat vlastnosti (i) a (ii). Pokud $c^\top v = 0$, pak (i) je určitě splněna a vlastnost (ii) dostaneme případnou změnou všech znamének ve t , neboť $v \neq 0$. Necht' tedy dále $c^\top v \neq 0$. Opět případnou změnou znamének (je-li tento součin kladný) dosáhneme $c^\top v < 0$, tj. platí (i). Kdyby (ii) neplatilo, tj. $v \geq 0$, pak $x(t) = \bar{x} + tv \geq 0$ pro všechna $t \geq 0$, a tedy $x(t) \in X$ pro všechna $t \geq 0$. Jenže v takovém případě je hodnota účelové funkce pro $x(t)$ rovna $c^\top x(t) = c^\top \bar{x} + t c^\top v \rightarrow \infty$ a s rostoucím $t \rightarrow \infty$ bude $c^\top x(t) \rightarrow -\infty$, tj. úloha (LP) je zdola neohraničená, a tedy nemá řešení, což je spor s předpoklady tvrzení (*). Vzhledem k libovolnosti x_0 a \bar{x} je tímto celé tvrzení dokázáno. ■

2.5 Dualita aneb peníze až na prvním místě

Ted' už sice máme vše potřebné k vybudování již několikrát avizovaného algoritmu, ale předtím si ještě uděláme krátkou a z praktického pohledu velmi důležitou odbočku. Na úlohy lineárního programování se dá totiž podívat i z trochu jiného úhlu pohledu. Co tím máme na mysli? Než si to ukážeme na samotné úloze lineárního programování, uvažme klasickou soustavu lineárních rovnic

$$Bx = f \quad (2.5.1)$$

pro nějakou matici B a vektor f (vhodných rozměrů) a připusťme, že nás v tuto chvíli nezajímá celé řešení soustavy (2.5.1), ale pouze nějaká jeho část určená vektorem g , tj.

$$\Phi(f) := g^T x.$$

Např. pro $g = e_i$ dostaneme i -tou složku řešení nebo volba $g = (1, \dots, 1)^T$ dává součet složek celého řešení atd. Jak potom závisí hodnota $\Phi(f)$ na samotném f ? Co se stane při změně f ? Uvážíme-li místo soustavy (2.5.1) úlohu

$$B^T v = g,$$

pak platí

$$\Phi(f) = g^T x = (B^T v)^T x = v^T Bx = v^T f.$$

Tedy se znalostí řešení nové soustavy (2.5) máme jasně danou závislost hodnoty $\Phi(f)$ na pravé straně f . Proto úlohu (2.5) můžeme považovat za tzv. *duální*⁵⁸ (též *adjungovanou*) k původní soustavě (2.5.1). Jak ji ale získat? V tuto chvíli to byl spíše *Deus ex machina*, ale v dalším příkladu to již uvidíme se všemi podrobnostmi.

Uvažme proto nyní úlohu lineárního programování

$$\begin{aligned} 4x_1 + 14x_2 &\rightarrow \min, \\ x_1 + 2x_2 &\geq 3, \quad x_1 + 4x_2 \geq 4, \\ 3x_1 + x_2 &\geq 3, \quad x_1 \geq 0, \quad x_2 \geq 0. \end{aligned}$$

Tuto úlohu můžeme celkem snadno vyřešit pomocí obrázku a najít $f^* = 15$ v bodě $x^* = [2, 1/2]$. Ale pojďme to zkusit jinak. Jestliže x_1, x_2 má být řešením této úlohy, pak z prvního omezení a podmínky nezápornosti plyne

$$4x_1 + 14x_2 = 4(x_1 + 3\frac{1}{2}x_2) \geq 4(x_1 + 2x_2) \geq 4 \cdot 3 = 12,$$

tj. $f^* \geq 12$. Můžeme dostat i lepší odhad? Ano, pomocí druhého a třetí omezení obdržíme

$$4x_1 + 14x_2 \geq 3(x_1 + 4x_2) + \frac{1}{3}(3x_1 + x_2) \geq 3 \cdot 4 + \frac{1}{3} \cdot 3 = 13,$$

tj. $f^* \geq 13$, a pomocí prvního a druhého omezení dokonce dostaneme

$$4x_1 + 14x_2 \geq 3(x_1 + 4x_2) + 1(x_1 + 2x_2) \geq 3 \cdot 4 + 1 \cdot 3 = 15,$$

tj. $f^* \geq 15$. Jak dobrý odhad můžeme takovým způsobem vůbec získat? A co to vlastně znamená „takovým způsobem“? Začněme od konce. Pomocí jednotlivých omezení se snažíme odvodit nerovnost ve tvaru

$$d_1 x_1 + d_2 x_2 \geq h,$$

⁵⁸ Označení „duální“ je klasický matematický termín, se kterým se můžeme setkat v řadě oblastí. Jak ovšem označit původní úlohu? Tuto otázku si položil William Orchard-Hays někdy kolem roku 1954 a otec zakladatele lineárního programování Tobias Dantzig navrhl označení „primární“. Přírozené? Možná...

kde $d_1 \leq 4$, $d_2 \leq 14$ a číslo h je největší možné. Pak můžeme tvrdit, že pro všechna $x_1, x_2 \geq 0$ máme

$$4x_1 + 14x_2 \geq d_1 x_1 + d_2 x_2 \geq h,$$

a tudíž h je nějaká dolní hranice minimální hodnoty účelové funkce na X . Jak takovou nerovnost ale získat? Zkombinujeme všechny tři nerovnosti do tvaru podobnému nerovnosti výše s nějakými nezápornými koeficienty y_1, y_2, y_3 (nezápornost je potřeba, aby znaménko nerovnosti nemohlo být opačné). Potom

$$(1 \cdot y_1 + 1 \cdot y_2 + 3 \cdot y_3) x_1 + (2 \cdot y_1 + 4 \cdot y_2 + 1 \cdot y_3) x_2 = y_1 (x_1 + 2x_2) + y_2 (x_1 + 4x_2) + y_3 (3x_1 + x_2) \geq 3y_1 + 4y_2 + 3y_3,$$

tedy máme

$$d_1 := y_1 + y_2 + 3y_3, \quad d_2 := 2y_1 + 4y_2 + y_3, \quad h := 3y_1 + 4y_2 + 3y_3.$$

Už jen zbývá zajistit, aby h bylo největší možné a d_1, d_2 splňovala daná omezení. To nás ale přivádí k další úloze lineárního programování

$$\begin{aligned} 3y_1 + 4y_2 + 3y_3 &\rightarrow \max, \\ y_1 + y_2 + 3y_3 &\leq 4, \quad 2y_1 + 4y_2 + y_3 \leq 14, \quad y_1 \geq 0, \quad y_2 \geq 0, \quad y_3 \geq 0. \end{aligned}$$

Toto je tzv. *duální úloha* k úloze (2.5.1), která ohraničuje původní problém zdola v tom smyslu, že každý přípustný bod $[y_1, y_2, y_3]$ duální úlohy dává dolní hranici minima účelové funkce původní úlohy.

Duální úlohou k obecné úloze lineárního programování

$$\begin{aligned} \langle c, x \rangle &\rightarrow \min, \\ \sum_{j=1}^n a_{ij} x_j &\geq b_j, \quad i = 1, \dots, k, \\ \sum_{j=1}^n a_{ij} x_j &= b_j, \quad i = k+1, \dots, m, \\ x_j &\geq 0, \quad j = 1, \dots, s, \end{aligned}$$

je maximalizační úloha lineárního programování tvaru

$$\begin{aligned} \langle y, b \rangle &\rightarrow \max, \\ \sum_{i=1}^m a_{ij} y_i &\leq c_j, \quad j = 1, \dots, s, \\ \sum_{i=1}^m a_{ij} y_i &= c_j, \quad j = s+1, \dots, n, \\ y_i &\geq 0, \quad i = 1, \dots, k. \end{aligned}$$

Zejména tedy duální úlohou k úloze (LP) je úloha v základním tvaru, tj.

$$\langle y, b \rangle \rightarrow \max \quad \& \quad y \in Y := \{x \in \mathbb{R}^n \mid A^T y \leq c\}. \quad (\text{LP}_D)$$

Úlohy (LP) a (LP_D) můžeme současně zapsat do (velmi elegantní a symetrické) *Tuckerovy tabulky*

	$x_1 \geq 0$	$x_2 \geq 0$	\dots	$x_n \geq 0$	
y_1	a_{11}	a_{12}	\dots	a_{1n}	$= b_1$
y_2	a_{21}	a_{22}	\dots	a_{2n}	$= b_2$
\vdots	\vdots	\vdots	\ddots	\vdots	\vdots
y_m	a_{m1}	a_{m2}	\dots	a_{mn}	$= b_m$
	$\leq c_1$	$\leq c_2$	\dots	$\leq c_n$	

kteřá v tom nejobecnějším případě má tvar

Primární úloha									
		$x_1 \geq 0$	$x_2 \geq 0$	\dots	$x_s \geq 0$	x_{s+1}	\dots	x_n	
Duální úloha	$y_1 \geq 0$	a_{11}	a_{12}	\dots	a_{1s}	$a_{1,s+1}$	\dots	a_{1n}	$\geq b_1$
	$y_2 \geq 0$	a_{21}	a_{22}	\dots	a_{2s}	$a_{2,s+1}$	\dots	a_{2n}	$\geq b_2$
	\vdots	\vdots	\vdots	\ddots	\vdots	\vdots	\dots	\vdots	\vdots
	$y_k \geq 0$	a_{k1}	a_{k2}	\dots	a_{ks}	$a_{k,s+1}$	\dots	a_{kn}	$\geq b_k$
	y_{k+1}	$a_{k+1,1}$	$a_{k+1,2}$	\dots	$a_{k+1,s}$	$a_{k+1,s+1}$	\dots	$a_{k+1,n}$	$= b_{k+1}$
	\vdots	\vdots	\vdots	\ddots	\vdots	\vdots	\ddots	\vdots	\vdots
	y_m	a_{m1}	a_{m2}	\dots	a_{ms}	$a_{m,s+1}$	\dots	a_{mn}	$= b_m$
		$\leq c_1$	$\leq c_2$	\dots	$\leq c_s$	$= c_{s+1}$	\dots	$= c_n$	$\begin{array}{l} y^T b \rightarrow \max \\ c^T x \rightarrow \min \end{array}$

Dříve zmíněné ohraničení dané duální úlohou je vlastně obsahem tzv. *slabé věty o dualitě*. Její důkaz stejně jako důkaz její silnější verze zde uvádět nebudeme, neboť jsou přímým důsledkem obecnějšího tvrzení z teorie matematického programování.

VĚTA 2.5.1 SLABÁ VĚTA O DUALITĚ

Pro každý přípustný bod $x \in X$ úlohy (LP) a každý přípustný bod $y \in Y$ duální úlohy (LP_D) platí

$$\langle c, x \rangle \geq \langle y, b \rangle. \quad (2.5.2)$$

Jinými slovy, je-li y přípustný bod pro (LP_D), pak hodnota $\langle y, b \rangle$ udává dolní hranici minima účelové funkce úlohy (LP). Zejména je-li (LP) neohrazená (zdola), úloha (LP_D) musí být nepřípustná. Naopak, je-li úloha (LP_D) neohrazená (shora), pak primární úloha (LP) musí být nepřípustná.

Nerovnost (2.5.2) ve skutečnosti ukazuje, že omezení hodnot primární a duální úlohy je vzájemné, tj. f^* nemůže být menší než hodnota duální úlohy a zároveň naopak. Ovšem jak jsou tyto dvě hodnoty „vzdálené“ neboli jaký je tzv. *duální rozdíl*⁵⁹? Nulový nebo může být i nenulový?

⁵⁹ angl. *duality gap*

Odpověď na tuto otázku dává tzv. *Silná věta o dualitě*, ale ještě před ní si můžeme zformulovat jeden velmi jednoduchý (ale ne příliš praktický) důsledek plynoucí ihned z (2.5.2). Pokud se nám totiž podaří najít dvojici bodů $x^* \in X$ a $y^* \in Y$ takových, že hodnoty odpovídajících účelových funkcí jsou stejné, tj. $\langle c, x^* \rangle = \langle y^*, b \rangle$, pak jsme našli řešení obou úloh, neboť účelové funkce v těchto dvou bodech nabývají své největší a nejmenší hodnoty, kterou mohou nabýt mezi všemi $x \in X$ a $y \in Y$.

DŮSLEDEK 2.5.2 CERTIFIKÁT OPTIMALITY

Pokud pro nějaký přípustný bod $x^* \in X$ úlohy (LP) a nějaký přípustný bod $y^* \in Y$ úlohy (LP_D) nastane v (2.5.2) rovnost, tj. $\langle c, x^* \rangle = \langle y^*, b \rangle$, pak x^* a y^* jsou řešeními „svých“ úloh.

V úloze (LP) mohou nastat tři možnosti:

- (i) přípustná a ohraničená (tj. řešitelná);
- (ii) nepřípustná ($X = \emptyset$);
- (iii) neohraničená ($f^* = -\infty$)

a ty stejné možnosti mohou nastat i pro duální úlohu (LP_D). To dohromady dává celkem 9 možností. Tři z nich nejsou možné kvůli Slabé větě o dualitě (neohraničenost obou úloh i neohraničenost jedné úlohy a současně řešitelnost druhé úlohy). Navíc lze ukázat, že ani nepřípustnost jedné úlohy a řešitelnost druhé úlohy nemůže nastat. Zbyly nám tedy pouze 4 možnosti, které jsou shrnuty v následující tabulce a zároveň popsány v následující větě. Ve skutečnosti ze Silné věty o dualitě plyne pouze to, že primární úloha nemůže být řešitelná a současně duální úloha nepřípustná. To, že nemůže nastat ani symetrická varianta s řešitelnou duální úlohou a nepřípustnou primární úlohou, plyne z toho, že duální úloha je opět úlohou lineárního programování, takže tabulka musí být symetrická.

DÚ \ PÚ	NP ($f^* = \infty$)	PaO	NO ($f^* = -\infty$)
NP ($\varphi^* = -\infty$)	✓	✗	✓
PaO	✗	✓	✗
NO ($\varphi^* = \infty$)	✓	✗	✗

VĚTA 2.5.3 SILNÁ VĚTA O DUALITĚ

Pro úlohy (LP) a (LP_D) může nastat právě jedna z následujících možností:

- (i) obě úlohy jsou nepřípustné (tj. $X = \emptyset = Y$);
- (ii) úloha (LP) je neohraničená a úloha (LP_D) je nepřípustná;
- (iii) úloha (LP) je nepřípustná a úloha (LP_D) je neohraničená;
- (iv) obě úlohy jsou přípustné a ohraničené. Pak existují jejich řešení x^* a y^* , která navíc splňují

$$\langle c, x^* \rangle = \langle y^*, b \rangle,$$

VĚTA 2.5.3 SILNÁ VĚTA O DUALITĚ

tj. minimum dosažené v (LP) je stejné jako maximum dosažené v (LP_D) neboli hodnoty obou úloh jsou stejné.

Naše úvahy ohledně duální úlohy ale nejsou samoúčelné. Znalost řešení duální úlohy má totiž značný praktický význam, neboť poskytuje další informace o původní úloze. Uvažme opět problém optimálního jídelníčku ve standardním tvaru

$$\langle c, x \rangle \rightarrow \min \quad \& \quad Ax \geq b \quad \& \quad x \geq 0.$$

Pak duální úloha je maximalizační problém

$$\langle y, b \rangle \rightarrow \max \quad \& \quad A^T y \leq c \quad \& \quad y \geq 0.$$

Proměnné x_1, \dots, x_n z původní úlohy reprezentují množství jednotlivých surovin, ale jaký je význam duálních proměnných y_1, \dots, y_m ? To se pokusíme zjistit pomocí tzv. *rozměrové analýzy*. V duálních omezeních $\sum_{i=1}^m a_{ij} y_i \leq c_j$ je c_j jednotková cena j -suroviny a a_{ij} určuje množství i -té živiny v j -té surovině. Aby obě strany byly porovnatelné musí y_i udávat jednotkovou cenu i -té živiny. Protože b_i je minimální požadované množství i -té živiny za den, účelová funkce $\langle y, b \rangle$ pak určuje celkovou cenu požadovaných živin. V duální úloze tedy hledáme hodnoty $y = (y_1, \dots, y_m)^T$ určující ceny živin tak, aby se maximalizovala celková hodnota požadovaného příjmu živin za den za podmínky $y \geq 0$, přičemž celková hodnota živin v j -tém jídle $\sum_{i=1}^m a_{ij} y_i$ není větší než aktuální cena c_j j -tého jídla.

Ted' si ale představme, že nějaký obchodník nám nabídne přímý prodej živin bez jídla, např. ve formě vitamínových tablet (vše plyne syntetické, nenáročné na přípravu, plnohodnotně nahrazující vše potřebné atd.). Nabízí nám prodej i -té živiny za jednotkovou cenu y_i . Jak by měl správně zvolit cenu, abychom na tento obchod přistoupili? Připusťme, že nás zajímá pouze finanční stránka tohoto obchodu. Pak cena, kterou si účtuje za mix živin nahrazujících j -té jídlo by neměla být vyšší než původní cena j -tého jídla, což je právě omezení z duální úlohy. Je-li toto pravdivé pro všechna j , pak má smysl uzavřít obchod. Obchodník chce samozřejmě vydělat co nejvíce, takže by měl zvolit y_1, \dots, y_m maximalizující celkový zisk $\langle y, b \rangle$ při daných omezeních, tj. právě jako řešení duální úlohy⁶⁰. My sice vzhledem k nulovému duálnímu rozdílu mezi hodnotami primární a duální úlohy žádné peníze neušetříme, ale tento obchod pro nás může mít jiné nehmotné výhody – třeba úsporu času (i když v tomto případě to asi moc velká výhra nebude). V ekonomii se proto hodnoty y_1, \dots, y_m nazývají *stínovou cenou* jednotlivých živin. Slovo „stínová“ odráží fakt, že takový obchodník asi neexistuje nebo že by se mohlo jednat o komoditu, která na klasickém trhu neobchodovatelná. Nicméně stále v jistém smyslu vyjadřuje „cenu“ jednotlivých živin určenou tržními cenami jídel a našimi požadavky na skladbu příjmu živin.

⁶⁰ Zvolí-li jiné hodnoty, pak buď jeho potenciální zisk bude nižší nebo jím požadovaná cena bude vyšší než kolik by stál nákup surovin

Příklad 2.5.4

Uvažme řemeslníka, který k výrobě svých jediných dvou výrobků P a Q využívá dva stroje. Jednotlivé detaily jsou shrnuty v následující tabulce

	stroj A	stroj B	zisk [Kč]
výrobek P	3	2	300
výrobek Q	2	2/3	400

Stroj A nesmí běžet déle než 13 hodin za den a stroj B více než 12 hodin za den. Kolik vyrobit kusů jednotlivých výrobků, aby se maximalizoval zisk? To je řešením úlohy

$$300x_1 + 400x_2 \rightarrow \max$$

$$3x_1 + 2x_2 \leq 13 \quad \& \quad 2x_1 + 3x_2 \leq 12 \quad \& \quad x_1 \geq 0 \quad \& \quad x_2 \geq 0.$$

Řešením jsou 3 kusy výrobku P a 2 kusy výrobku Q s maximálním ziskem 1700 Kč.

Ted' ale uvažme takovouto situaci: čerstvý absolvent studia matematiky na PřF MU by chtěl začít podnikat ve stejném oboru. Ovšem k tomu je potřeba mít počáteční kapitál, který pochopitelně nemá a vzhledem k jeho věku a majetku (a taky dvakrát pozdě vrátil knihy do knihovny, takže už má záznam v registru) nelze očekávat, že by mu nějaká banka poskytla rozumný úvěr. Ale doslechl se, že náš řemeslník zvažuje odchod do důchodu. Náš mladý podnikatel se proto rozhodl chopit příležitosti a navštívit řemeslníka s touto nabídkou: pronajme si oba stroje, přičemž za hodinu provozu stroje A zaplatí y_1 Kč a za hodinu provozu stroje B zaplatí y_2 Kč. Současně zachová stávající druh produkce a výrobků, které bude prodávat sám tak, aby mu to pokrylo alespoň nájem. Nápad se řemeslníkovi pochopitelně líbil, neboť již nebude muset řešit běžné problémy drobného živnostníka (daně, EET atd.) a bude mít čas na vnoučata. Současně je výhodný i pro mladého absolventa, neboť nemusí drazé kupovat nové stroje. Nicméně řemeslník je stále nedůvěřivý, a tak je nutné využít dosažené vzdělání a přesvědčit jej o výhodnosti celé nabídky.

Pro výrobu výrobku P jsou potřeba 3 hodiny na stroji A a 2 hodiny na stroji B, tedy za něj zaplatí nájem

$$3y_1 + 2y_2.$$

Současně však tento nájem bude alespoň roven jeho aktuálnímu zisku z prodeje tohoto výrobku, tj.

$$3y_1 + 2y_2 \geq 300.$$

Podobně pro výrobek Q dostaneme

$$2y_1 + 3y_2 \geq 400.$$

Řemeslníkovi se návrh líbil, ale nelíbí se mu možnost pronájmu pouze na část dne. Chtěl by pronajmout stroje na celou jejich provozní dobu. Jakou nabídku má mladík udělat, aby řemeslník souhlasil (on ví, že jeho cenní zisk je 1700 Kč)?

Mladý absolvent se proto znovu zamyslel. Ví, že bude muset zaplatit $13y_1 + 12y_2$, ovšem nechce platit více než je nutné. Takže hodnoty y_1 a y_2 musí být řešením úlohy

$$13y_1 + 12y_2 \rightarrow \min$$

$$3y_1 + 2y_2 \geq 300 \quad \& \quad 2y_1 + 3y_2 \geq 400 \quad \& \quad y_1 \geq 0 \quad \& \quad y_2 \geq 0.$$

Má tato úloha řešení? Ze Silné věty o dualitě plyne, že ano. Není těžké se přesvědčit, že tím je volba $y_1 = 20$ a $y_2 = 120$ s platbou 1700 Kč za den.

⁶¹ Rozdíl mezi duální a primární je především v typu hledaného extrému. Ovšem již velmi dobře víme, že přechod mezi minimalizací a maximalizací je pouze o změně znaménka

Dualitu lze přiblížit i z fyzikálního pohledu. Tentokrát ale začneme z druhé strany⁶¹, tj.

$$\langle c, x \rangle \rightarrow \max \quad \& \quad Ax \leq b$$

s $n = 3$ a nechť tato úloha je řešitelná (tj. přípustná a ohraničená). Bod $x \in \mathbb{R}^3$ je prvkem třídimen-

⁶² To je jedna ze dvou částí \mathbb{R}^3 vymezena danou rovinou

zionaálního prostoru a vektor c chápeme jako gravitační vektor (ten tedy směřuje dolů – ovšem ne nutně v kolmém směru). Každá nerovnost v $Ax \leq b$ určuje nějaký *poloprostor*⁶² a jejich průnik je zdola ohraničený konvexní polyedr. Každá dvoudimenzionální stěna polyedru je určena jednou rovností $\langle a_i, x \rangle = b_i$, kde vektory a_1, \dots, a_m tentokrát bereme jako transponované řádky matice A . Označme stěnu určenou rovností $\langle a_i, x \rangle = b_i$ jako S_i (ovšem ne každá nerovnost v soustavě $Ax \leq b$ musí odpovídat nějaké stěně, takže S_i nemusí být definováno pro všechna $i = 1, \dots, m$).

Představme si, že hranice polyedru jsou vyrobeny z lepenky. Vezmeme miniaturní ocelovou kuličku a umístíme ji někam dovnitř polyedru. Kulička dopadne na stěnu a skutálí do nejnižší položeného vrcholu (proto potřebujeme ohraničenost), příp. zůstane stát na vodorovné stěně. Tuto pozici označíme x^* . Zde působí největší síla, takže se jedná o řešení dané maximalizační úlohy. V této stabilní pozici se kulička dotýká několika dvoudimenzionálních stěn (obvykle tří). Označme D množinu indexů i takových, že se kulička dotýká S_i . Pro $i \in D$ tedy máme $\langle a_i, x^* \rangle = b_i$. Gravitační síla F na kuličku je úměrná vektoru c . Tato síla je rozložena mezi síly, jimiž kulička působí (tlačí) na jednotlivé stěny, kterých se dotýká. Síla F_i , kterou kulička působí na stěnu S_i je kolmá na S_i s směrem ven z polyedru (zanedbáme-li tření), viz Obrázek XXX

Síly působící na kuličku jsou v rovnováze, tj. $F = \sum_{i \in D} F_i$. Normálový vektor stěny S_i nasměrovaný ven z polyedru je právě a_i , takže síle F_i je úměrná a_i . To znamená, že pro nějaká nezáporná čísla y_i^* musí platit

$$\sum_{i \in D} y_i^* a_i = c.$$

Položíme-li $y_i^* = 0$ pro $i \in \{1, 2, \dots, m\} \setminus D$, můžeme psát $\sum_{i=1}^m y_i^* a_i = A^T y^* = c$, tj. y^* je přípustným bodem duální úlohy

$$\langle y, b \rangle \rightarrow \min \quad \& \quad A^T y = c \quad \& \quad y \geq 0.$$

Uvažme součin $\langle y^*, Ax^* - b \rangle$. Pro $i \in \{1, 2, \dots, m\} \setminus D$ je i -tá složka vektoru y^* rovna 0, zatímco pro $i \in D$ je i -tá složka $Ax^* - b$ rovna 0. Tedy tento skalární součin je roven nule, tj. $\langle y^*, b \rangle = \langle y^*, Ax^* \rangle = \langle A^T y^*, x^* \rangle = \langle c, x^* \rangle$. Celkem tedy x^* je přípustný bod pro primární úlohu, y^* je přípustný bod pro duální úlohu a $\langle y^*, b \rangle = \langle c, x^* \rangle$. Pak z Certifikátu optimality (Důsledek 2.5.2) plyne, že y^* je řešením duální úlohy.

2.6 Simplexový algoritmus

Geometrický základ metody využívající tzv. *simplexovou tabulku* jsme již naznačili. Nyní se na to podíváme více analyticky (a mnohem méně geometricky). Během postupu budeme generovat posloupnost BPB $x^{[0]}, x^{[1]}, \dots, x^{[p]} \in X$, přičemž v p -tém kroku zjistíme, zda

- (i) úloha (LP) je neohraničená;
- (ii) existuje BPB $x^{[p+1]}$ takový, že $\langle c, x^{[p+1]} \rangle < \langle c, x^{[p]} \rangle$, tj. tento bod dává lepší (menší) hodnotu, přičemž souřadnice tohoto bodu explicitně vypočteme;
- (iii) bod $x^{[p]}$ je řešením úlohy (LP).

Předpokládejme, že úloha (LP) je přípustná (v opačném případě se nám ani nepodaří samotný algoritmus spustit, tj. najít výchozí BPB $x^{[0]}$), a mějme již posloupnost BPB $x^{[0]}, x^{[1]}, \dots, x^{[p]}$. Pro

jednoduchost označme $J := J(x)$ nosič bodu $:= x^{[p]}$ a (nějakou) jeho bázi $B := B(x)$. Pak při značení $x = (x_1, \dots, x_n)^\top$ a $c = (c_1, \dots, c_n)^\top$ platí

$$\sum_{j \in B} x_j a_j = \sum_{j \in J} x_j a_j = b \quad (2.6.1)$$

a označme hodnotu účelové funkce v bodě x jako z_0 , tj.

$$z_0 := \sum_{j \in B} c_j x_j = \sum_{j \in J} c_j x_j. \quad (2.6.2)$$

Jelikož vektory $\{a_j \mid j \in B\}$ jsou lineárně nezávislé, libovolný vektor a_1, \dots, a_n může být vyjádřen jejich pomocí s koeficienty λ_{jk} , tj.

$$a_k = \sum_{j \in B} \lambda_{jk} a_j, \quad k = 1, \dots, n \quad (2.6.3)$$

a definujeme čísla

$$z_k := \sum_{j \in B} \lambda_{jk} c_j, \quad k = 1, \dots, n. \quad (2.6.4)$$

Ve vyjádření (2.6.3) je zřejmé, že $\lambda_{kk} = 1$ pro všechna $k \in B$ a $\lambda_{jk} = 0$ pro $j \in B \setminus \{k\}$. Proto též $z_k = c_k$ pro všechna $k \in B$.



Jaký je význam koeficientu λ_{jk} ? Pro jednoduchost přeznačme pořadí vektorů a_1, \dots, a_n tak, že báze B našeho bodu x je $B = \{1, \dots, m\}$ a definujme matici

$$L := \begin{pmatrix} \lambda_{11} & \dots & \lambda_{1n} \\ \vdots & \ddots & \vdots \\ \lambda_{m1} & \dots & \lambda_{mn} \end{pmatrix} = (\lambda_{jk})_{\substack{j=1, \dots, m \\ k=1, \dots, n}} \in \mathbb{R}^{m \times n}. \quad (2.6.5)$$

Rovnost (2.6.3) vlastně znamená

$$A = (a_1, \dots, a_n) = \underbrace{(a_1, \dots, a_m)}_{A_B} \begin{pmatrix} \lambda_{11} & \dots & \lambda_{1n} \\ \vdots & \ddots & \vdots \\ \lambda_{m1} & \dots & \lambda_{mn} \end{pmatrix} = A_B \cdot L,$$

tj.

$$L = A_B^{-1} A \quad \text{neboli} \quad \begin{pmatrix} \lambda_{11} & \dots & \lambda_{1n} \\ \vdots & \ddots & \vdots \\ \lambda_{m1} & \dots & \lambda_{mn} \end{pmatrix} = A_B^{-1} (a_1, \dots, a_m). \quad (2.6.6)$$

Pak k -tý sloupec matice L dostaneme vynásobením e_k zprava, tj.

$$\begin{pmatrix} \lambda_{1k} \\ \vdots \\ \lambda_{mk} \end{pmatrix} = A_B^{-1} \begin{pmatrix} a_{1k} \\ \vdots \\ a_{mk} \end{pmatrix} = A_B^{-1} a_k,$$

pak z_k definované v (2.6.4) můžeme vyjádřit jako

$$z_k = c_1 \lambda_{1k} + c_2 \lambda_{2k} + \dots + c_m \lambda_{mk} = (c_1, \dots, c_m) \begin{pmatrix} \lambda_{1k} \\ \vdots \\ \lambda_{mk} \end{pmatrix} = c_B^\top A_B^{-1} a_k.$$

Odtud také snadno nahlédneme, že pro libovolné $y \in X$ platí

$$\langle c, x \rangle = \langle c, y \rangle + \sum_{j \notin B} (z_j - c_j) y_j. \quad (2.6.7)$$

Vskutku, je-li $y \in X$, pak $Ay = b$ neboli pro $N := \{1, \dots, n\} \setminus B$ máme $A_B y_B + A_N y_N = b$, takže

$$y_B = A_B^{-1} b - A_B^{-1} A_N y_N = A_B^{-1} b - A_B^{-1} \sum_{j \in N} y_j a_j,$$

z čehož plyne

$$\begin{aligned} \langle c, y \rangle &= \langle c_B, y_B \rangle + \langle c_N, y_N \rangle = c_B^\top (A_B^{-1} b - A_B^{-1} \sum_{j \in N} y_j a_j) + c_N^\top y_N = \\ &= c_B^\top A_B^{-1} b - (c_B^\top A_B^{-1} \sum_{j \in N} y_j a_j - \sum_{j \in N} c_j y_j) = c_B^\top x_B - \sum_{j \in N} (c_B^\top A_B^{-1} a_j - c_j) y_j = \\ &= \langle c, x \rangle - \sum_{j \in N} (z_j - c_j) y_j. \end{aligned}$$

Potom ale k tomu, aby x bylo řešením úlohy (LP), je vzhledem k nezápornosti a libovlnosti složek y_j nutná nekladnost sumy na pravé straně. Vzhledem ke znaménku rozdílu $z_k - c_k$ a čísla λ_{jk} a řešitelnosti úlohy (LP) jsou pro nás zajímavé pouze 3 situace, které si nyní rozebereme. Jejich důkazy jsou založeny na rovnostech (2.6.1)–(2.6.4), ze kterých pro libovolné $\alpha \in \mathbb{R}$ odečtením α -násobku rovnosti (2.6.3) od rovnosti (2.6.1) dostaneme

$$\sum_{j \in B} (x_j - \alpha \lambda_{jk}) a_j + \alpha a_k = b \quad \text{pro všechna } k = 1, \dots, n \quad (2.6.8)$$

a podobně odečtením α -násobku (2.6.4) od (2.6.2) a přidáním αc_k na obě strany získáme

$$\sum_{j \in B} (x_j - \alpha \lambda_{jk}) c_j + \alpha c_k = z_0 - \alpha (z_k - c_k) \quad \text{pro všechna } k = 1, \dots, n. \quad (2.6.9)$$

VĚTA 2.6.1

Jestliže existuje $s \in \{1, 2, \dots, n\} \setminus B$ takový, že $z_s - c_s > 0$ a $\lambda_{js} \leq 0$ pro všechna $j \in B$, pak úloha (LP) je neohraničená.

Důkaz. Necht' index $s \in \{1, 2, \dots, n\} \setminus B$ je takový, že $z_s - c_s > 0$ a současně $\lambda_{js} \geq 0$ pro všechna $j \in B$. Pak pro libovolné $\alpha > 0$ je bod

$$x(\alpha) := \begin{cases} x_j - \alpha \lambda_{js}, & j \in B, \\ \alpha, & j = s, \\ 0, & j \notin B \cup \{s\} \end{cases}$$

je přípustný, tj. $x(\alpha) \in X$, neboť $m+1$ jeho složek je kladných pro $B \cup \{s\}$, ostatní jsou nulové a z rovnosti (2.6.8) aplikované pro $k = s$ plyne $Ax(\alpha) = \sum_{j \in B \cup \{s\}} x_j(\alpha) a_j = b$. Jelikož α může být libovolné kladné číslo, dostaneme dále z (2.6.9) pro $k = s$, že

$$\langle c, x(\alpha) \rangle = \sum_{j \in B \cup \{s\}} c_j x_j(\alpha) = z_0 - \alpha \underbrace{(z_s - c_s)}_{>0} \rightarrow -\infty \quad \text{pro } \alpha \rightarrow \infty,$$

tj. úloha (LP) je neohraničená. ■

Další možnost nás přivede k situaci, kdy existuje „lepší“ BPB. Uvidíme, že báze tohoto bodu $x^{[p+1]}$ a bodu $x^{[p]}$ se liší pouze v jediném indexu. Takovým bodům se říká *sousední*. Ponecháme na čtenáři, aby si rozmyslel, že v případě degenerované úlohy (LP) existují body, které sousedí samy se sebou.

VĚTA 2.6.2

Nechť úloha (LP) je nedegenerovaná. Existuje-li index $s \in \{1, 2, \dots, n\} \setminus B (= J)$ takový, že $z_s - c_s > 0$ a současně $\lambda_{ks} > 0$ pro nějaké $k \in B$, pak existuje nový BPB $\hat{x} \in X$, pro který je hodnota účelové funkce menší než v bodě x .

Důkaz. Necht' pro nějaká $s \in \{1, 2, \dots, n\} \setminus B$ a $k \in B$ jsou splněny předpoklady věty. Pak díky nedegenerovanosti dané úlohy jistě existuje číslo $\alpha > 0$ takové, že $x_j - \alpha \lambda_{js} \geq 0$ pro všechna $j \in B$. Vskutku, pro $\lambda_{js} \leq 0$ je uvedená nerovnost splněna triviálně, zatímco v případě $\lambda_{js} > 0$ je potřeba vzít $\alpha \in (0, x_j / \lambda_{js}]$. Jelikož $\lambda_{ks} > 0$ je splněno nejméně pro jeden index j a nejvýše pro n indexů, můžeme definovat číslo

$$\hat{\alpha} := \min \left\{ \frac{x_j}{\lambda_{js}} \mid j \in B \text{ \& } \lambda_{js} > 0 \right\} > 0.$$

Označíme-li index, ve kterém nastává toto minimum, jako $r \in B$, pak pro něj platí $x_r - \hat{\alpha} \lambda_{rs} = 0$. Navíc index r je určen jednoznačně, neboť v opačném případě bychom s využitím (2.6.8) pro $k = s$ a $\alpha = \hat{\alpha}$ získali vyjádření vektoru b jako nějakou lineární kombinaci nejvýše $(m - 1)$ -tice vektorů $\{a_j \mid j \in (B \setminus \{r_1, r_2\}) \cup \{s\}\}$, což je spor \neq .

Jestliže se nyní omezíme na $\alpha \in (0, \hat{\alpha})$ a definujeme-li bod $x(\alpha)$ se složkami

$$x_j(\alpha) := \begin{cases} x_j - \alpha \lambda_{js}, & j \in B, \\ \alpha, & j = s, \\ 0, & j \notin B \cup \{s\}, \end{cases} \quad (2.6.10)$$

pak z první části a rovnosti (2.6.8) pro $k = s$ vyplývá, že $x(\alpha) \in X$. Navíc hodnota účelové funkce v takovém bodě podle (2.6.9) pro $k = s$ splňuje

$$z(\alpha) := \langle c, x(\alpha) \rangle = z_0 - \alpha \underbrace{(z_s - c_s)}_{>0} < z_0, \quad (2.6.11)$$

tj. je menší než v bodě x . Ovšem bod $x(\alpha)$ nemusí být BPB. To se stane pouze a jenom pro $\alpha = \hat{\alpha}$ vedoucí k $x_r(\hat{\alpha}) = 0$, neboť v opačném případě bychom vždy měli $x(\alpha)$ s $(m + 1)$ -ticí nezáporných složek. Proto $\hat{x} := x(\hat{\alpha})$ je hledaným „lepší“ BPB. ■

Nastane-li situace popsaná v právě dokázané Větě 2.6.2, pak za další člen naší posloupnosti BPB bereme $x^{[p+1]} := x(\hat{\alpha})$ s bází $B^{[p+1]} = (B^{[p]} \setminus \{r\}) \cup \{s\}$, přičemž se mluví o tom, že index r (příp. vektor a_r) vystupuje z báze a současně index s (příp. vektor a_s) vstupuje/přibíráme do báze. Navíc z (2.6.11) lze snadno vidět, že právě tato volba $x(\hat{\alpha})$ dává největší možné snížení hodnoty účelové funkce mezi všemi body $x(\alpha)$ pro $\alpha \in (0, \hat{\alpha})$. Jelikož víme, že BPB je pouze konečný počet, bude tento algoritmus za předpokladu nedegenerovanosti úlohy (LP) vždy konečnokrokový. Jestliže je ale úloha (LP) degenerovaná, pak se může stát, že $\hat{\alpha} = 0$ (pokud by $x_j = 0$ pro nějaké $j \in B$), takže by nastalo $x(\hat{\alpha}) = x^{[p]}$, a tudíž by došlo pouze ke změně báze BPB $x^{[p]}$. To by mohlo vést až k zacyklení celého algoritmu, čemuž se budeme podrobněji věnovat ve druhé části následujícího odstavce.

VĚTA 2.6.3

Nechť úloha (LP) je nedegenerovaná. Bod x je řešením úlohy (LP) právě tehdy, když $z_k - c_k \leq 0$ pro všechna $k \in \{1, \dots, n\} \setminus B$.

Důkaz. „ \implies “ Vzhledem k předpokladu negenerovanosti úlohy (LP) plyne nutnost uvedené podmínky z tvrzení Vět 2.6.1 a 2.6.2.

„ \Leftarrow “ Necht' $y = (y_1, \dots, y_n)^T \neq x$ je jiný přípustný bod množiny X , tj. $Ay = b$ a $y \geq 0$. Protože $z_k - c_k \leq 0$ pro všechna $k \in \{1, \dots, n\} \setminus B$ a současně $z_j - c_j = 0$ pro každé $j \in B$ dle (2.6.4), platí $z_k - c_k \leq 0$ pro všechna $k \in \{1, \dots, n\}$. Proto

$$\langle y, z \rangle = \sum_{i=1}^n y_i z_i \leq \sum_{i=1}^n y_i c_i = \langle y, c \rangle =: z^*.$$

Navíc dosazením vyjádření (2.6.3) do rovnosti $Ay = b$ dostaneme

$$b = Ay = \sum_{i=1}^n y_i a_i = \sum_{i=1}^n y_i \left(\sum_{j \in B} \lambda_{ji} a_j \right) = \sum_{i=1}^n y_i (\lambda_{j_1 i} a_{j_1} + \lambda_{j_2 i} a_{j_2} + \dots + \lambda_{j_m i} a_{j_m})$$

neboli

$$b = \sum_{j \in B} \left(\sum_{i=1}^n \lambda_{ji} y_i \right) a_j. \quad (2.6.12)$$

Podobně dosazením (2.6.4) do nerovnosti $\langle y, z \rangle \leq z^*$ získáme

$$\langle y, z \rangle = \sum_{i=1}^n y_i z_i = \sum_{i=1}^n y_i \left(\sum_{j \in B} \lambda_{ji} c_j \right) \leq z^*$$

neboli

$$\sum_{j \in B} \left(\sum_{i=1}^n \lambda_{ji} y_i \right) c_j \leq z^*. \quad (2.6.13)$$

Protože vektory $\{a_j \mid j \in B\}$ tvoří bázi \mathbb{R}^m a protože vyjádření libovolného vektoru pomocí báze musí být jednoznačné, koeficienty odpovídajících vektorů v rovnostech (2.6.1) a (2.6.12) si musí být rovny, tj. $\sum_{i=1}^n \lambda_{ji} y_i = x_j$ pro všechna $j \in B$. Pak ale z (2.6.13) plyne

$$z^* \geq \sum_{j \in B} \left(\sum_{i=1}^n \lambda_{ji} y_i \right) c_j = \sum_{j \in B} x_j c_j = z_0.$$

Jelikož $y \in X$ bylo voleno libovolně, vyplývá z poslední nerovnosti, že bod x je řešením úlohy (LP). ■

Všímavý čtenář jistě postřehl, že pro důkaz implikace „ \Leftarrow “ v předchozím tvrzení nebyl požadavek nedegenerovanosti dané úlohy potřeba. To znamená, že podmínka „ $z_k - c_k \leq 0$ pro všechna $k \in \{1, \dots, n\} \setminus B$ “ je postačující pro nalezení řešení libovolné úlohy (LP). Ovšem v případě řešitelnosti mohou nastat dvě diametrálně odlišné situace podle jednoznačnosti řešení: řešení se realizuje buď v jediném bodě (ten je nutně BPB) nebo ve více bodech (z nichž alespoň musí být BPB), v kterémžto případě jich bude nutně nespočetně mnoho (neboť množina řešení musí být konvexní, takže řešením bude i každý bod ležící na úsečce spojující dvě řešení). Odlišnost těchto situací není ale zajímavá pouze z matematického pohledu, protože i v praxi jistě mohou nastat situace, kdy přeci jen některé řešení bude výhodnější než jiné, ačkoli dává stejnou minimální hodnotu účelové funkce⁶³.

⁶³ Např. kdyby šlo o výrobu založenou na nákupu některých zdrojů od externích dodavatelů, tak bychom mohli preferovat řešení založené na dodavateli, s nímž se nám lépe obchoduje nebo který by nám následně mohl nabídnout i nějakou slevu. To jsou ale faktory, které nelze a priori zahrnout do samotné formulace problému.

DŮSLEDEK 2.6.4

Jestliže $z_k - c_k < 0$ pro všechna $k \in \{1, \dots, n\} \setminus B$, pak x je jediné řešení úlohy (LP).

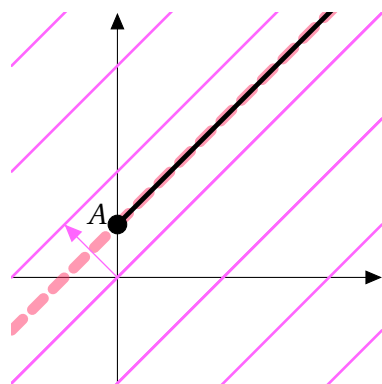
Důkaz. Jelikož $z_k - c_k < 0$ pro všechny indexy $k \in \{1, \dots, n\} \setminus B$, plyne z Věty 2.6.3, že x je řešením úlohy (LP). Předpokládejme, že řešení dané úlohy není jednoznačné, tj. existuje další přípustný

bod $y \in X$ splňující $x \neq y$ a $\langle c, y \rangle = \langle c, x \rangle$. Pak nutně $y_j > 0$ pro nějaký index $j \in \{1, \dots, n\} \setminus B$. Kdyby tomu tak nebylo (tj. $y_j = 0$ pro všechna $j \in \{1, \dots, n\} \setminus B$), pak by množina B také byla bází bodu y , což by vzhledem jednoznačnosti BPB nutně znamenalo $x = y$, a tedy bychom dostali spor ξ , viz Větu 2.4.6. Potom ale z (2.6.7) vyplývá

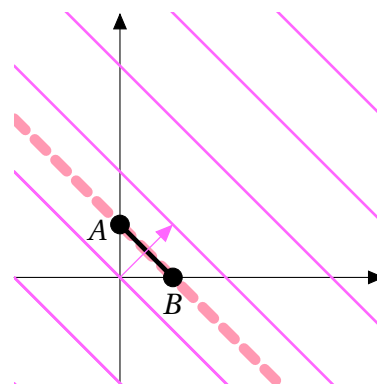
$$\langle c, y \rangle = \langle c, x \rangle - \underbrace{\sum_{j \notin B} (z_j - c_j)}_{< 0 \ \forall j \notin B} \underbrace{y_j}_{\substack{\geq 0 \ \forall j \notin B \\ > 0 \ \text{pro nějaké } j \notin B}} > \langle c, x \rangle,$$

což je opět spor ξ , tj. y nemůže být řešením, čímž je jednoznačnost dokázána. ■

Všimněte si, že předchozí tvrzení je platné bez ohledu na degenerovanost úlohy (LP). Na druhou stranu pro následující důsledek je již nedegenerovanost potřeba. V opačném případě bychom se totiž mohli dostat do situace, ve které bychom za alternativní řešení považovali tentýž degenerovaný BPB vyjádřený pouze pomocí různých bází. Z důkazu navíc uvidíme, že v případě rovnosti $z_s - c_s = 0$ pro nějaké $s \in \{1, \dots, n\} \setminus B$ mohou nastat dvě různé situace podle toho, zda alternativní řešení je BPB či nikoli, jak je vidět na Obrázcích 2.17.a–2.17.b. Rozdíl mezi těmito situacemi lze popsat také „v řeči bází“: jde totiž o to, zda po nalezení řešení úlohy (LP) ještě existuje nějaký index $r \in B$, na jehož úkor by index s vstoupil do báze.



Obrázek 2.17.a: Úloha $-x_1 + x_2 \rightarrow \min$ pro $-x_1 + x_2 = 1$, $x_1 \geq 0$ a $x_2 \geq 0$, která má nespočetně mnoho řešení, ale pouze jedno z nich je BPB (tím je bod A).



Obrázek 2.17.b: Úloha $x_1 + x_2 \rightarrow \min$ pro $x_1 + x_2 = 1$, $x_1 \geq 0$ a $x_2 \geq 0$, která má nespočetně mnoho řešení a dva z nich jsou BPB (body A a B).

DŮSLEDEK 2.6.5

Nechť úloha (LP) je nedegenerovaná. Jestliže x je řešením úlohy (LP) a je-li $z_s - c_s = 0$ pro nějaké $s \in \{1, \dots, n\} \setminus B$, pak existuje i jiné (alternativní) řešení této úlohy. Je-li navíc $\lambda_{ks} > 0$ pro nějaké $k \in B$, pak řešením je i jiný BPB, zatímco v opačném případě je x jediným BPB, který řeší úlohu (LP).

Důkaz. Tvrzení víceméně ihned vyplývá z konstrukce bodu $x(\alpha)$ v důkazu Věty 2.6.2, neboť nerovnost $z_s - c_s > 0$ jsme využili až v jeho závěru. Vskutku, je-li $z_s - c_s = 0$ pro nějaké $s \in \{1, \dots, n\} \setminus B$ a současně $\lambda_{ks} > 0$ pro nějaké $k \in B$, pak lze zkonstruovat nový BPB $x(\bar{\alpha})$ stejně jako v (2.6.10) s tím rozdílem, že tentokrát v (2.6.11) nastane rovnost, tj. hodnoty účelové funkce v BPB x a $x(\bar{\alpha})$ jsou totožné. Kdyby ale bylo $z_s - c_s = 0$ pro nějaké $s \in \{1, \dots, n\} \setminus B$ a současně $\lambda_{js} \leq 0$ pro všechna $j \in B$, pak by bod $x(\alpha)$ z (2.6.10) stále byl přípustným bodem pro všechna $\alpha \geq 0$ a splňoval by $\langle c, x \rangle = \langle c, x(\alpha) \rangle$, ale neexistovala by hodnota $\bar{\alpha} > 0$, pro kterou by se z něj stal BPB. ■

2.7 Simplexová tabulka

Samotný výpočet řešení úlohy (LP) pomocí simplexového algoritmu je „organizován“ do tzv. *simplexové tabulky* a, jak se za chvíli přesvědčíme, jeho průběh velmi připomíná *Gaussovu eliminaci* – ovšem s řízeným výběrem *pivota* (či *hlavního/klíčového prvku*) a „drobným“ omezením při úpravách dalších řádků. V literatuře narazíte na různé tvary této tabulky, které se liší především uspořádáním jednotlivých sloupců či řádků a také mírou podrobnosti uvedených údajů. Vždy z nich ale lze vyčíst všechny informace potřebné pro výpočet.

Abychom náš výklad zbytečně nekomplikovali, zdá se nám nejvýhodnější začít v prvním kroku s velmi podrobnou tabulkou o velikosti $(n + 4) \times (m + 4)$, jak můžeme vidět v Tabulce 2.3 níže. K ní ještě ale musíme přidat několik vysvětlení:

- (i) první řádek je pouze informativní popis a v praktických výpočtech není potřeba;
- (ii) červeně jsou uvedena symbolická označení jednotlivých sloupců/řádků, zde nic nedosazujeme;
- (iii) zelenou barvou jsou uvedeny proměnné s bazickými indexy, zde místo b_1, \dots, b_m dosadíme konkrétní hodnoty;
- (iv) černou barvou jsou uvedeny proměnné či složky různých vektorů/matic a při praktickém výpočtu místo nich uvedeme konkrétní číselné hodnoty.
- (v) šedé buňky zůstávají prázdné.

V dalších krocích již není potřeba takto podrobná tabulka a dále se pracuje s tzv. *redukovanou simplexovou tabulkou*, ve které již není potřeba vypisovat sloupec označený c_B a řádky označené c a z_k . Někteří autoři také kvůli úspoře času/barvy či kvůli fyzické námaze ignorují při vyplňování tabulky nulové hodnoty, takže jejich tabulka občas působí neúplně. My tento úzus aplikovat nebudeme, neboť by se nám při ručním výpočtu mohla tabulka snadno stát velmi nepřehlednou (a při výpočtu na počítači se tím nemusíme moc trápit).

hodnoty bazických složek vektoru c	označení proměnných odpovídajících indexům v bázi	označení proměnných a odpovídající prvky matice L				hodnota bazických složek vektoru x	podíl $\frac{x_B}{\lambda_{j,s}}$ (pro vhodné s)
c_B	x_B	x_1	x_2	\dots	x_n	x_B	x_B / λ
c_{b_1}	x_{b_1}	$\lambda_{b_1,1}$	$\lambda_{b_1,2}$	\dots	$\lambda_{b_1,n}$	x_{b_1}	$x_{b_1} / \lambda_{b_1,s}$
c_{b_2}	x_{b_2}	$\lambda_{b_2,1}$	$\lambda_{b_2,2}$	\dots	$\lambda_{b_2,n}$	x_{b_2}	$x_{b_2} / \lambda_{b_2,s}$
\vdots	\vdots	\vdots	\vdots	\ddots	\vdots	\vdots	\vdots
c_{b_m}	x_{b_m}	$\lambda_{b_m,1}$	$\lambda_{b_m,2}$	\dots	$\lambda_{b_m,n}$	x_{b_m}	$x_{b_m} / \lambda_{b_m,s}$
	c	c_1	c_2	\dots	c_n		
	z_k	z_1	z_2	\dots	z_n		
	$z_k - c_k$	$z_1 - c_1$	$z_2 - c_2$	\dots	$z_n - c_n$		

Tabulka 2.3: Výchozí simplexová tabulka

Příklad 2.7.1

Uvažme úlohu

$$\begin{aligned} x_1 - x_2 + 2x_3 + 3x_4 &\rightarrow \min \\ 2x_1 + x_2 + x_3 &= 1 \quad \& \quad x_1 + 3x_2 + x_4 = 2 \\ x_1 \geq 0 \quad \& \quad x_2 \geq 0 \quad \& \quad x_3 \geq 0 \quad \& \quad x_4 \geq 0. \end{aligned}$$

Tato úloha je již v kanonickém tvaru a odpovídá úloze (LP) s $n = 4$, $m = 2$ a volbou

$$c = \begin{pmatrix} 1 & -1 & 2 & 3 \end{pmatrix}^T \quad \& \quad A = \begin{pmatrix} 2 & 1 & 1 & 0 \\ 1 & 3 & 0 & 1 \end{pmatrix} \quad \& \quad b = \begin{pmatrix} 1 & 2 \end{pmatrix}^T.$$

Jelikož vektor b lze získat pouze pomocí dvou (libovolných) sloupců matice A , jedná se o nede degenerovanou úlohu. Zvolíme-li bázi $B = \{3, 4\}$, pak $A_B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, takže podle (2.4.3) máme $x_B = A_B^{-1} b = b$ a v souladu s (2.6.6) dostaneme $L = A_B^{-1} A = A$. Proto výchozí simplexová tabulka má následující podobu (poslední sloupec s x_B/λ zatím necháme být, neboť jeho výpočet úzce souvisí s prvním krokem simplexového algoritmu).

c_B	x_B	x_1	x_2	\dots	x_n	x_B	x_B/λ
2	x_3	2	1	1	0	1	$x_3/\lambda_{3,s}$
3	x_4	1	3	0	2	2	$x_4/\lambda_{4,s}$
	c	1	-1	2	3		
	z_k	7	11	2	3		
	$z_k - c_k$	6	12	0	0	8	

Tabulka 2.4: Výchozí simplexová tabulka k Příkladu 2.7.1

A teď již samotný popis celého simplexového algoritmu:

- (1) Danou úlohu lineárního programování převedeme do kanonického tvaru (LP), tj. vhodnými úpravami z rovností uděláme nerovnosti a proměnné bez omezení na znaménko nahradíme novými proměnnými omezeními na znaménko atd.
- (2) Pokud nějaká složka vektoru b je záporná, vynásobíme příslušnou rovnici -1 .
- (3) Nalezneme výchozí BPB a určíme jeho bázi B . Obsahuje-li matice A všechna sloupce jednotkové matice, pak za bázi bereme odpovídající indexy. V opačném případě využijeme tzv. *dvoufázovou simplexovou metodu*, které se budeme věnovat v následující Odstavci 2.8. Zde také zjistíme případnou nepřipustnost úlohy, neboť se nám nepodaří najít výchozí BPB.
- (4) Sestavíme výchozí simplexovou tabulku pro zvolený BPB.
- (5) Jestliže
 - (i) $z_k - c_k \leq 0$ pro všechna $k \notin B$, je výpočet hotov. Zvolený BPB je řešením úlohy. KONEC.
 - (ii) $z_s - c_s > 0$ pro nějaké $s \notin B$ a současně
 - (ii_a) $\lambda_{js} \leq 0$ pro všechna $j \in B$, pak úloha nemá řešení. KONEC.

(ii_b) $\lambda_{js} > 0$ pro nějaké $j \in B$, pak existuje nový BPB s „lepší“ hodnotou účelové funkce, který najdeme v dalších krocích.

(6) Určíme *klíčový sloupec*:

- (i) pokud $z_s - c_s > 0$ pro jediný index $s \notin B$, pak je to sloupec odpovídající proměnné x_s .
- (ii) pokud $z_s - c_s > 0$ pro více indexů $s \notin B$, pak je to sloupec odpovídající proměnné x_s , pro kterou má rozdíl $z_s - c_s$ největší hodnotu, tj.

$$z_s - c_s = \max \{z_j - c_j \mid j \notin B \text{ \& } z_j - c_j > 0\}.$$

Kdyby i toto nastalo ve více indexech, vybereme ten nejmenší (tj. ten, který je v simplexové tabulce nejvíce vlevo neboli pravidlo *první volby*), viz Poznámka 2.7.2 níže.

Index odpovídající klíčovému sloupci vstupuje do báze. Současně z báze vystupuje ten index r , pro který

$$\frac{x_r}{\lambda_{rs}} = \min \left\{ \frac{x_j}{\lambda_{js}} \mid j \in B \text{ \& } \lambda_{js} > 0 \right\} \quad (2.7.1)$$

(k tomu využijeme poslední sloupec simplexové tabulky), čímž dostaneme tzv. *klíčový řádek*. Takto získáváme novou bázi $B \setminus \{r\} \cup \{s\}$ a prvek λ_{rs} v simplexové tabulce je tzv. *klíčový prvek* nebo *pivot* (průnik klíčového řádku a sloupce).

(7) Sestavíme novou simplexovou tabulku (bez řádku odpovídajících c a z_k) odpovídající bázi $B \setminus \{r\} \cup \{s\}$. Nemusíme ale vše přepočítávat od začátku. Stačí upravit stávající tabulku podle následujících pravidel (viz Větu 2.7.3 níže)

- (i) klíčový řádek vynásobíme číslem $1/\lambda_{rs}$, abychom na pozici klíčového prvku dostali 1;
- (ii) v ostatních řádcích klíčového sloupce chceme mít 0, takže od každého ze zbývajících $m - 1$ řádků pro x_B odečteme klíčový řádek vynásobený číslem $\lambda_{js}/\lambda_{rs}$;
- (iii) podobnou úpravu provedeme i pro řádek $z_k - c_k$, tj. klíčový řádek vynásobíme $(z_s - c_s)/\lambda_{rs}$;
- (iv) ve sloupci označeném x_B s bazickými proměnnými místo x_r napíšeme x_s a ostatní bazické proměnné ponecháme stejné.

(8) Pokračujeme krokem 5.

Poznámka 2.7.2. Volba klíčového sloupce popsána v kroku (6)(ii) je tzv. *Dantzigovo pravidlo* (nebo *maximálního relativního ocenění*, které je motivováno tím, že hodnota $z_s - c_s$ jsou tzv. *redukované náklady*, tj. vyjadřuje jednotkovou míru změny (poklesu) účelové funkce při změně v proměnné x_k . Koeficienty v účelové funkci totiž reprezentují jednotkové náklady „aktivit“ určených proměnnými. Redukované náklady pak indikují, jak moc by se musel redukovat k -tý koeficient účelové funkce, aby současná „aktivita“ reprezentovaná k -tou proměnnou byla nákladově efektivní (tzn., že by měla kladnou hodnotu v optimálním řešení). Existují samozřejmě i jiné postupy, např.

- *lexikografické pravidlo*, ve kterém ihned bereme nejmenší index (tj. ten nejvíce vlevo). Toto asi není příliš efektivní, ale je velmi jednoduché.⁶⁴
- *pravidlo největšího vylepšení*, ve kterém při dané volbě docílíme maximální možné změny hodnoty účelové funkce.
- *pravidlo nejvzdálenějšího vrcholu*, ve kterém se chceme z aktuálního BPB přesunout do nejvzdálenějšího BPB (neboli vrcholu simplexu).

⁶⁴ TO NENÍ ÚPLNĚ PŘESNÉ!!

Poslední dvě alternativní pravidla jsou výpočetně velmi náročná, neboť potřebujeme porovnat určité hodnoty ve všech možných nových BPB.

Jestliže úloha (LP) není neohraničená ani nepřipustná a současně zvolený výchozí BPB není jejím řešením, pak v souladu s Větou 2.6.2 zkonstruujeme nový BPB. Ovšem tento bod již nemůžeme počítat pomocí rovnosti (2.4.3), považovat jej za nový výchozí BPB a vracet se až k bodu (4) s konstrukcí nové výchozí simplexové tabulky. Všechny tyto informace totiž dává nová (simplexová) tabulka získaná dle pravidel popsaných v bodě (7), takže můžeme pokračovat přímo bodem (5). Korektnost tohoto postupu zaručuje potvrzuje následující věta, která nevyžaduje nedegenerovanost dané úlohy. Navíc body (6)–(7) lze aplikovat i v případě, kdy $z_s - c_s = 0$ a současně $\lambda_{ks} > 0$ pro nějaké $k \in B$, čímž nalezneme jiný bod se stejnou funkční hodnotou. Ten je buď alternativním řešením nebo součástí cyklu (je-li úloha degenerovaná).

VĚTA 2.7.3

Nechť x je BPB úlohy (LP) s bází B . Jestliže $z_s - c_s \geq 0$ a současně $\lambda_{ks} > 0$ pro nějakou dvojici indexů $s \in \{1, 2, \dots, n\} \setminus B$ a $k \in B$, pak tabulka získaná podle pravidel popsaných v bodě (7) simplexového algoritmu je simplexovou tabulkou pro nový BPB \hat{x} s bází $\hat{B} := B \setminus \{r\} \cup \{s\}$ získanou v bodě (6).

Důkaz. Z Věty 2.6.2 víme, že novým bazickým bodem je

$$\hat{x}_j := \begin{cases} x_j - \frac{\lambda_{js}}{\lambda_{rs}} x_r, & j \in B, \\ \frac{x_r}{\lambda_{rs}}, & j = s, \\ 0, & j \notin B \cup \{s\}, \end{cases}$$

pro vhodný index $r \in B$, přičemž $\hat{x}_r = 0$ a současně $\lambda_{rs} > 0$, viz (2.7.1). Jelikož

$$a_s \stackrel{(2.6.3)}{=} \sum_{j \in B} \lambda_{js} a_j = \sum_{j \in B \setminus \{r\}} \lambda_{js} a_j + \lambda_{rs} a_r,$$

plyne odtud

$$a_r = \frac{1}{\lambda_{rs}} a_s - \sum_{j \in B \setminus \{r\}} \frac{\lambda_{js}}{\lambda_{rs}} a_j. \quad (2.7.2)$$

Proto můžeme libovolný vektor a_1, \dots, a_n vyjádřit jako

$$\begin{aligned} a_k &\stackrel{(2.6.3)}{=} \sum_{j \in B \setminus \{r\}} \lambda_{jk} a_j + \lambda_{rk} a_r = \\ &\stackrel{(2.7.2)}{=} \sum_{j \in B \setminus \{r\}} \lambda_{jk} a_j + \lambda_{rk} \left(\frac{1}{\lambda_{rs}} a_s - \sum_{j \in B \setminus \{r\}} \frac{\lambda_{js}}{\lambda_{rs}} a_j \right) = \\ &= \sum_{j \in B \setminus \{r\}} \left(\lambda_{jk} - \lambda_{js} \frac{\lambda_{rk}}{\lambda_{rs}} \right) a_j + \frac{\lambda_{rk}}{\lambda_{rs}} a_s, \quad k = 1, \dots, n, \end{aligned}$$

neboli dostáváme vyjádření

$$a_k = \sum_{j \in \hat{B}} \hat{\lambda}_{jk} a_j, \quad k = 1, \dots, n,$$

s koeficienty

$$\hat{\lambda}_{jk} = \begin{cases} \lambda_{jk} - \lambda_{js} \frac{\lambda_{rk}}{\lambda_{rs}}, & j \in B \setminus \{r\}, k \in \{1, \dots, n\}, \\ \frac{\lambda_{rk}}{\lambda_{rs}}, & j = s, k \in \{1, \dots, n\}, \end{cases}$$

určujícími novou matici \widehat{L} odpovídající bázi \widehat{B} , viz (2.6.5). Pak ale $\widehat{\lambda}_{ss} = 1$ a $\widehat{\lambda}_{js} = 0$ pro $j \in \widehat{B} \setminus \{s\}$, což odpovídá právě krokům popsaným v simplexovém algoritmu v bodě (7)(i)–(ii). Nyní ještě zbývá určit $\widehat{z}_k - c_k$ a $\langle c, \widehat{x} \rangle$. Poněvadž platí



$$\begin{aligned} \widehat{z}_k - c_k &\stackrel{(2.6.4)}{=} \sum_{j \in \widehat{B}} \widehat{\lambda}_{jk} c_j - c_k = \sum_{j \in B \setminus \{r\}} \left(\lambda_{jk} - \lambda_{js} \frac{\lambda_{rk}}{\lambda_{rs}} \right) c_j + \frac{\lambda_{rk}}{\lambda_{rs}} c_s - c_k = \\ &= \left(\sum_{j \in B} \lambda_{jk} c_j - \lambda_{rk} c_r \right) - \frac{\lambda_{rk}}{\lambda_{rs}} \left(\sum_{j \in B} \lambda_{js} c_s - \lambda_{rs} c_r \right) + \frac{\lambda_{rk}}{\lambda_{rs}} c_s - c_k = \\ &= \sum_{j \in B} \lambda_{jk} c_j - c_k - \frac{\lambda_{rk}}{\lambda_{rs}} \left(\sum_{j \in B} \lambda_{js} c_s - c_s \right) = \\ &= (z_k - c_k) - \frac{z_s - c_s}{\lambda_{rs}} \lambda_{rk} \end{aligned}$$

a současně také

$$\begin{aligned} \langle c, \widehat{x} \rangle &= \sum_{j \in \widehat{B}} c_j \widehat{x}_j = \sum_{j \in B \setminus \{r\}} \left(x_j - \frac{\lambda_{js}}{\lambda_{rs}} x_r \right) c_j + \frac{x_r}{\lambda_{rs}} c_s = \\ &= \left(\sum_{j \in B} c_j x_j - c_r x_r \right) - \frac{x_r}{\lambda_{rs}} \left(\sum_{j \in B} \lambda_{js} c_j - \lambda_{rs} c_r - c_s \right) = \\ &= \sum_{j \in B} c_j x_j - \frac{x_r}{\lambda_{rs}} \left(\sum_{j \in B} \lambda_{js} c_j - c_s \right) = \\ &= \langle c, x \rangle - \frac{z_s - c_s}{\lambda_{rs}} x_r, \end{aligned}$$

jsou hodnoty $\widehat{z}_k - c_k$ a $\langle c, \widehat{x} \rangle$ zcela v souladu s výpočtem popsaným v simplexovém algoritmu v bodě (7)(iii). ■

Příklad 2.7.4

Nedegenerovaná úloha v kanonickém tvaru:   <https://goo.gl/7H15Uy>



Příklad 2.7.5

Nedegenerovaná úloha:   <https://goo.gl/7H15Uy>



Příklad 2.7.6

Neohraničená úloha:   <https://goo.gl/7H15Uy>

Příklad 2.7.7

Alternativní řešení #1:   <https://goo.gl/7H15Uy>

Příklad 2.7.8

Alternativní řešení #2:   <https://goo.gl/7H15Uy>

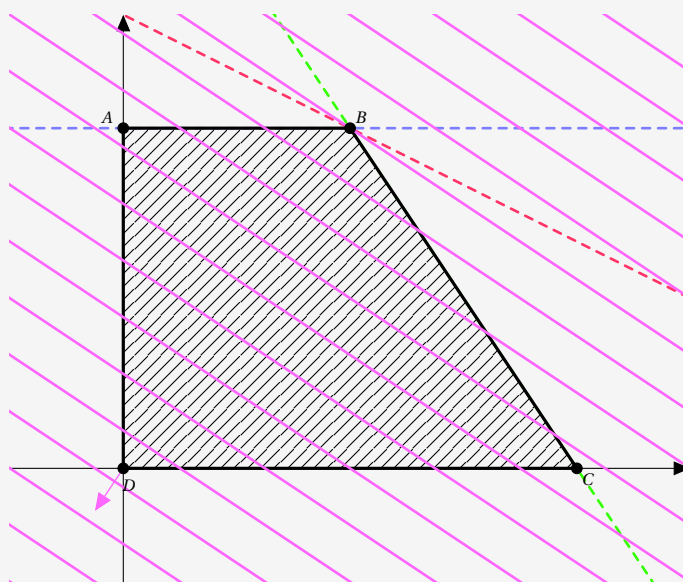
Příklad 2.7.9

S pomocí simplexového algoritmu vyřešme úlohu

$$\begin{aligned} -4x_1 - 6x_2 &\rightarrow \min \\ 6x_1 + 4x_2 &\leq 24 \quad \& \quad 5x_1 + 10x_2 \leq 40 \\ x_1 &\geq 0 \quad \& \quad 0 \leq x_2 \leq 3. \end{aligned}$$

Řešení. Převodem do kanonického tvaru dostaneme úlohu

$$\begin{aligned} -4x_1 - 6x_2 &\rightarrow \min \\ 6x_1 + 4x_2 + s_3 &= 24 \\ 5x_1 + 10x_2 + s_4 &= 40 \\ x_2 + s_5 &= 3 \\ x_1 \geq 0 \quad \& \quad x_2 \geq 0 \quad \& \quad s_3 \geq 0 \quad \& \quad s_4 \geq 0 \quad \& \quad s_5 \geq 0. \end{aligned}$$



Obrázek 2.18: Kompletní vizualizace dané úlohy se zvýrazněním normálového vektoru vrstevnic.

Odpovídající matice A a vektor b jsou

$$A = \begin{pmatrix} 6 & 4 & 1 & 0 & 0 \\ 5 & 10 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}, \quad b = \begin{pmatrix} 24 \\ 40 \\ 3 \end{pmatrix},$$

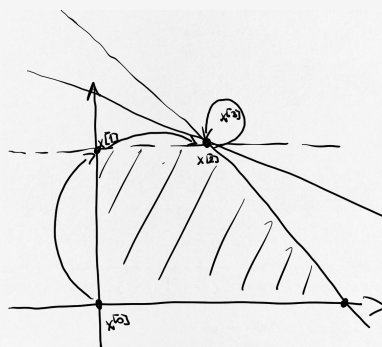
z čehož vidíme, že úloha je degenerovaná, neboť vektor b lze získat jako lineární kombinaci pouze dvou sloupců matice A , viz také Obrázek 2.18. Za výchozí BPB může vzít $x^{[0]} = [0, 0, 24, 40, 3]$ s bází $B(x^{[0]}) = \{s_3, s_4, s_5\}$ a můžeme pustit do samotného výpočtu.

c_B	x_B	x_1	x_2	s_3	s_4	s_5	x_B	x_B/λ
0	s_3	6	4	1	0	0	24	6
0	s_4	5	10	0	1	0	40	4
0	s_5	0	①	0	0	1	3	3 →
	c_k	-4	-6	0	0	0		
	z_k	0	0	0	0	0		
	$z_k - c_k$	4	6 ↑	0	0	0	0	
	s_3	6	0	1	0	-4	12	2
	s_4	⑤	0	0	1	-10	10	2 →
	x_2	0	1	0	0	1	3	✘
	$z_k - c_k$	4 ↑	0	0	0	-6	-18	
	s_3	0	0	1	-6/5	⑧	0	0 →
	x_1	1	0	0	1/5	-2	2	✘
	x_2	0	1	0	0	1	3	3
	$z_k - c_k$	0	0	0	-4/5	2 ↑	-26	
	s_5	0	0	1/8	-6/40	1	0	
	x_1	1	0	1/4	-1/10	0	2	
	x_2	0	1	-1/8	3/20	0	3	
	$z_k - c_k$	0	0	-1/4	-1/2	0	-26	

Hned ve druhém kroku jsme dostali možnost výběru pro volbu klíčového řádku, což je důsledek degenerovanosti stejně jako nulová hodnota s_3 ve třetím kroku. Postupně jsme prošly tyto BPB

$$x^{[0]} = [0, 0, 24, 40, 3] \rightsquigarrow x^{[1]} = [0, 3, 12, 10, 0] \rightsquigarrow x^{[2]} = [2, 3, 0, 0, 0] \rightsquigarrow x^{[3]} = [2, 3, 0, 0, 0].$$

V předposledním kroku jsme se dostali do degenerovaného BPB. Naštěstí již po jednom dalším kroku jsme „dorazili do řešení“. Ovšem to je fakticky stejné jako v předposledním kroku, pouze jsme změnili bázi B , viz také Obrázek 2.19.



Obrázek 2.19: Vizualizace průběhu simplexového algoritmu v původních proměnných.

⁶⁵ angl. *stalling*

Takto vzniká tzv. *cyklus*, který v případě konečné délky (jako před chvílí) způsobí pouze *od-
dálání*⁶⁵ nalezení řešení. Jenže situace může být i mnohem horší než v předchozím příkladě a může dojít i k nekonečné smyčce, skrze níž se nikdy nedostaneme k řešení. Toto ale není v rozporu s tvrzením Věty 2.6.2, která *de facto* zaručuje konvergenci simplexové metody. V degenerovaném případě se totiž může stát, že navzdory změně báze a nalezení nového BPB *nedojde* ke „zlepšení“ hodnoty účelové funkce. Důkaz konvergence simplexové metody pochází od G. Dantzig a je (nejspíše) z let 1947–1948. Je samozřejmě snadné zkonstruovat degenerovanou úlohu (LP), ovšem docílit současně nějakého cyklu při řešení simplexovou metodou již vyžaduje nemalé úsilí. První takový příklad pochází od Hoffmana z roku 1951 (ačkoli byl publikován až 1953):

$$\frac{\cos \varphi - 1}{\cos \varphi} x_1 + w x_2 + 2w x_4 + 4 \sin^2 \varphi x_5 + w(2 - 4 \cos^2 \varphi) x_6 + 4 \sin^2 \varphi x_7 + w(1 - 2 \cos \varphi) x_8 \rightarrow \min$$

za podmínek

$$x_9 = 1$$

$$\cos \varphi x_1 - w \cos \varphi x_2 + \cos 2\varphi x_3 - 2w \cos^2 \varphi x_4 + \cos 2\varphi x_5 +$$

$$+ 2w \cos^2 \varphi x_6 + \cos \varphi x_7 + w \cos \varphi x_8 + x_{10} = 0$$

$$\frac{\operatorname{tg} \varphi \sin \varphi}{w} x_1 + \cos \varphi x_2 + \frac{\operatorname{tg} \varphi \sin 2\varphi}{w} x_3 + \cos 2\varphi x_4 -$$

$$- \frac{2 \sin^2 \varphi}{w} x_5 + \cos 2\varphi x_6 - \frac{\operatorname{tg} \varphi \sin \varphi}{w} x_7 + \cos \varphi x_8 + x_{11} = 0$$

$$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0, x_5 \geq 0, x_6 \geq 0, x_7 \geq 0, x_8 \geq 0, x_9 \geq 0, x_{10} \geq 0, x_{11} \geq 0,$$

kde $\varphi = \frac{2\pi}{5}$ a w je libovolné číslo větší než $\frac{1-\cos \varphi}{1-2\cos \varphi}$. Potom dostaneme cyklus délky 10 (tj. první a desátá tabulka jsou totožné) bez toho, aniž by simplexový algoritmus ukázal, že výchozí BPB s $x^{[0]} = [0, \dots, 0, 1, 0, 0]$ a všechny další body dávají minimální (tj. nulovou) hodnotu účelové funkce. Při výpočtu se využívá skutečnost, že pro $\varphi = \frac{2\pi}{5}$ platí (i) $\cos 2\varphi = \cos 3\varphi$, (ii) $\sin 2\varphi = -\sin 3\varphi$ a (iii) $\cos 2\varphi + \cos \varphi = \cos \varphi \cos 2\varphi$. Také bychom měli poznamenat, že hlavní rolí prvního omezení $x_9 = 1$ je výrazné zjednodušení nalezení výchozího BPB. Navíc se při degenerovaných úlohách stává, že máme více možností pro volbu klíčového řádku, takže pro dosažení zmíněného cyklu je nutné použít pravidlo *první volby*, tj. volí se ten řádek, který je v simplexové tabulce výše. Lee se v článku z roku 1997 snaží objasnit genezi tohoto příkladu z algebraického i geometrického

pohledu. Matice 2×2 vytvořená z prvního a druhého sloupce ve druhém a třetím omezení

$$A = \begin{pmatrix} \cos \varphi & -w \cos \varphi \\ \frac{\operatorname{tg} \varphi \sin \varphi}{w} & \cos \varphi \end{pmatrix}$$

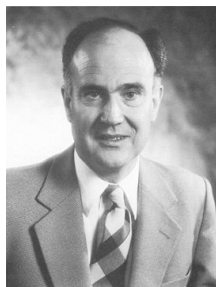
tvoří tzv. *cyklickou množinu*: A^2 odpovídá třetímu a čtvrtému sloupci, A^3 pátému a šestému sloupci, A^4 sedmému a osmému sloupci a konečně $A^5 = I$. Takové matice konečného řádu lze využít ke konstrukci dalších cyklických problémů.

Po tomto Hoffmanově příkladu byla zveřejněna ještě celá řada dalších (a značně jednodušších) příkladů zacyklení simplexového algoritmu. Avšak Hoffmanův příklad je kromě prvenství unikátní také v jiném ohledu – koeficienty jsou dány pomocí trigonometrických funkcí. Je samozřejmě zcela přirozenou otázkou, jak dlouhý/krátký cyklus můžeme získat v závislosti na velikosti dané úlohy lineárního programování. V uvedeném příkladě máme 3 omezení (kromě podmínek na znaménko jednotlivých proměnných) a 11 proměnných. Lze jich mít i méně? Dá se ukázat, že pro úlohu ve tvaru

$$\langle c, x \rangle \rightarrow \min \quad \& \quad Iy + Ax = b \quad \& \quad x \geq 0 \quad \& \quad y \geq 0$$

s $n \times n$ jednotkovou maticí I a $A \in \mathbb{R}^{m \times (n-m)}$, kde n je celkový počet proměnných, $x \in \mathbb{R}^{n-m}$, $y \in \mathbb{R}^m$ a $m < n$, je možné získat cyklus v případě, že $m \geq 2$, $n \geq m + 3$ a $n \geq 6$. Pro zacyklení v neoptimálním bodě je dokonce potřeba $m \geq 3$, $n \geq m + 3$ a $n \geq 7$. Úloha (LP) s pouhými 2 nebazickými proměnnými se nikdy nezacyklí. Pro dosažení cyklu v takové úloze je potřeba mít alespoň 6 proměnných, 2 rovnice a alespoň 3 nebazické proměnné. Pak nejkratší délka cyklu je 6. My si tuto situaci ilustrujeme na příkladu z roku 1955 od Evelyny Beale⁶⁶, ve kterém jsou 3 rovnice, 7 proměnných a dojde v něm právě k cyklu délky 6 v neoptimálním bodě (tedy vzhledem k předchozímu komentáři máme „nejmenší“ možnou úlohu s takovým chováním).

66



Evelyn Martin Lansdowne Beale (8. září 1928 – 23. prosince 1985) byl anglický matematik a statistik a jeden z prvních průkopníků matematického programování. V roce 1979 získal za „za jeho využití matematických a statistických dovedností v průmyslových problémech a za jeho přínos k teorii matematického programování“ ocenění *Fellowship of the Royal Society*, čímž se stal doživotním členem Královské společnosti v Londýně.

⁶⁷ Robert Gary Bland (* 25. února 1948) je americký matematik, který v současnosti působí jako profesor operačního výzkumu a informačního inženýrství na Cornellově univerzitě

Příklad 2.7.10

Degenerovaná úloha: 📄 🗨️ 🖨️ <https://goo.gl/7H15Uy>

Zacyklení lze předejít různými úpravami pravidel pro výběr klíčového prvku v simplexovém algoritmu. Např.

- (i) *pravidlo náhodného výběru* (navrženo Dantzigem v roce 1963): máme-li více vhodných kandidátů pro výběr klíčového řádku či sloupce, rozhodneme se náhodně (třeba pomocí hodu kostkou);
- (ii) *Blandovo*⁶⁷ *pravidlo* (1977): v případě rovnosti bereme nejnižší možný index (pro určení klíčového řádku) a v případě klíčového sloupce volíme ten, který je nejvíce vlevo a splňuje $z_s - c_s > 0$ (tj. ignorujeme Dantzigovo pravidlo ohledně výběru nejvyšší možné hodnoty $z_s - c_s$). Toto pravidlo sice není úplně efektivní a konvergence může být velmi pomalá, ale dá se ukázat, že s tímto pravidlem nedojde k zacyklení.

A jak si s cykly poradí počítače? V námi uvažovaných pravidlech byly koeficienty pouze racionální čísla a naše výpočty byly vždy „přesné“. Je však zřejmé, že ruční výpočty jsou jiné než výpočty prováděné počítačem. V případě použití nějakého výpočetního systému nepracujícího v racionálních číslech ale v desetinných číslech s jistým počtem cifer bude nutně docházet k zao-krouhlovacím chybám (obzvláště v případě iracionálních koeficientů). V takovém případě může

docházet také k zacyklení (k tzv. *výpočetním cyklům*), což je ale principiálně něco zcela jiného než *klasické cykly*, které jsme si před chvílí popsali. Nicméně dá se očekávat, že pokud úloha projevuje klasické zacyklení, bude docházet také k výpočetním cyklům a naopak.

Ještě poznamenejme, že cykly jsou z praktického hlediska velmi vzácné (a podobně pro degenerované úlohy – kolik náhodně vygenerovaných úloh daného rozměru bude asi degenerovaných?). Např. v roce 1952 byl publikován problém jisté rafinérie, který byl degenerovaný. Avšak při jeho řešení simplexovým algoritmem žádný problém s konvergencí *nenastal*⁶⁸.

⁶⁸ V roce 1949 požádal Koopmans Dantziga, aby něco udělal s požadavkem nedegenerovanosti v důkazu konvergence simplexové metody. V ideálním případě aby zkusil dokázat tvrzení bez této podmínky, kterou Dantzig původně považoval za zcela smysluplnou. Jenže jaká je vlastně pravděpodobnost, že čtyři roviny v \mathbb{R}^3 se potkají v jednom bodě? Ale stalo se něco nečekaného. Ukázalo se totiž, že ačkoli pravděpodobnost degenerovanosti problému lineárního programování je nulová, každý praktický problém z oblasti letectva testovaný Dantzigovým oddělením byl degenerovaný. Degenerovanost se neměla stát, ale děla se. A nebyla to výjimka, ale pravidlo! Viz [21].

2.8 Hledání výchozího BPB

Simplexovou metodu a její realizaci jsme si popsali a podrobně ilustrovali v předchozích odstavcích. Všechny uvedené příklady však měly jednu společnou vlastnost: matice A z kanonického tvaru vždy obsahovala jednotkovou matici, díky čemuž jsme snadno určili výchozí BPB. V ostatních případech to ale tak snadné být nemusí. Máme sice obecný návod pro výpočet x_B a příslušných koeficientů λ_{ij} pro libovolnou bázi B , jenže takový bod nemusí být přípustný. To může při větším počtu proměnných a omezení vést k velkému počtu výpočtů, než se nám podaří najít nějaký BPB. Proto potřebujeme nějaký efektivní nástroj pro určení výchozího BPB pro simplexový algoritmus. Nejčastěji se můžeme setkat s dvěma metodami

- (i) *metoda umělé báze* (též *dvoufázová metoda*) založené na řešení pomocné úlohy lineárního programování.
- (ii) *metoda velkého M* (též *penalizační metoda*), ve které se do účelové funkce přidají pomocné proměnné s maximálně nevýhodnou cenou M . Tato metoda sice není početně výhodnější, ale je implementována ve většině výpočetních systémů.

My se nyní budeme věnovat pouze první zmíněné metodě. V *první fázi* simplexový algoritmus na problém (LP) s pomocnou účelovou funkcí. Tím určíme, zda je původní problém přípustný, v kterémžto případě současně nalezneme nějaký BPB. Tento bod vezmeme jako výchozí BPB pro simplexový algoritmus, který aplikujeme na původní problém ve *druhé fázi*. Tento postup má několik výhod:

- (i) nejsou potřeba žádné předpoklady na původní systém (může být nepřípustný, neřešitelný a dokonce by mohl mít i duplicitní podmínky, což ale nedovolujeme);
- (ii) výchozí BPB pro první fázi nalezneme snadno;
- (iii) na konci první fáze máme BPB pro původní úlohu (pokud existuje) a může víceméně přímo pokračovat druhou fází, aniž bychom museli přepočítávat koeficienty λ_{ij} .

Poté, co danou úlohu lineárního programování převedeme do kanonického tvaru (LP), tzv. *inicializace* zahájíme první fází. V ní rozšíříme soustavu $Ax = b$ o *umělé proměnné* $w_1, \dots, w_m \geq 0$ ⁶⁹ tak, že platí

$$Ax + Iw = b. \quad (2.8.1)$$

Poté aplikujeme simplexový algoritmus na úlohu

$$\left. \begin{array}{l} w_1 + \dots + w_m \rightarrow \min \\ Ax + Iw = b \quad \& \quad x_1 \geq 0 \quad \& \quad \dots \quad \& \quad x_n \geq 0 \quad \& \quad w_1 \geq 0 \quad \& \quad \dots \quad \& \quad w_m \geq 0, \end{array} \right\} \quad (2.8.2)$$


⁶⁹ Není nutné vždy zavádět všechny umělé proměnné. Pokud se proměnná x_j objevuje pouze v jedné rovnici (tj. v matici A máme kladný násobek nějakého sloupce jednotkové $m \times m$ matice, vynásobíme tuto rovnost tak, aby koeficient u x_j byl $+1$, a tuto proměnnou x_j zavedeme do báze místo w_j , které vůbec nevyužijeme. Další postup je víceméně stejný, pouze nikde nevystupuje w_j a již neplatí obecný zápis omezení (2.8.1) uvedený níže.

přičemž za výchozí BPB můžeme vzít $x = 0$ a $w = b$. Jelikož $w_1, \dots, w_m \geq 0$, úloha nemůže být neohraničená a v závislosti na nalezeném řešení x^* , w^* úlohy (2.8.2) mohou nastat pouze tři možnosti:



- (i) jestliže $\sum_{i=1}^m w_i^* > 0$, pak původní úloha je nepřipustná, tj. pro původní problém v kanonickém tvaru nelze simplexový algoritmus, neboť pro něj nenajdeme výchozí BPB. Vskutku, kdyby totiž nějaké $\bar{x} \geq 0$ bylo přípustné, pak by dvojice $x = \bar{x}$ a $w = 0$ byla přípustným bodem pro úlohu (2.8.2) s nulovou hodnotou účelové funkce, což je spor s nalezeným optimem určeným w^* .
- (ii) jestliže $\sum_{i=1}^m w_i^* = 0$ a všechny umělé proměnné jsou mimo bázi, pak nutně $w^* = 0$ a bod x^* je BPB původní úlohy (LP).
- (iii) jestliže $\sum_{i=1}^m w_i^* = 0$ a alespoň jedna umělá proměnná je v bázi (nutně s nulovou hodnotou), pak výchozí BPB pro původní úlohu (LP) dostaneme s těmito proměnnými jako součástí výchozí báze. Toto se stane, má-li původní problém (LP) nadbytečné rovnice a často také má-li degenerované řešení. Pochopitelně je ale možné chtít pokračovat s první fází tak dlouho, dokud se nám nepodaří vyloučit všechny pomocné proměnné z báze (čímž dostaneme předchozí případ), abychom minimalizovali počet proměnných v další fázi. Toto se nám podaří zejména v případě, že řádek simplexové tabulky odpovídající umělé proměnné v bázi bude obsahovat alespoň jedno kladné číslo v některém ze sloupců pro původní proměnné úlohy v kanonickém tvaru. Pak totiž můžeme udělat ještě jeden krok simplexového algoritmu, ve kterém tuto umělou proměnnou vyvedeme z báze a původní proměnnou zavedeme do báze (umělá proměnná má v tuto chvíli nulovou hodnotu, takže ve sloupci „ x_B/λ “ bude nula a my ji tak v souladu s podmínkou (2.7.1) můžeme vyloučit z báze). Jenže v případě nadbytečných omezení toto nemusí být vůbec možné. Např. kdyby řádek simplexové tabulky odpovídající takové umělé proměnné obsahoval pouze nuly ve sloupcích pro původní proměnné z kanonického tvaru. Ovšem v takovém případě můžeme tento řádek (a umělou proměnnou) dále ignorovat, neboť se v další fázi rozhodně nezmění, tj. podaří se nám odstranit jedno omezení, které je pro daný problém zcela nadbytečné.

Pokud nastal případ (ii) nebo (iii), pak pokračujeme *druhou fází*. Na jejím začátku vymažeme z finální simplexové tabulky všechny sloupce odpovídající nebazickým umělým proměnným, hodnotu pomocné účelové funkce nahradíme hodnotou účelové funkce původní úlohy (LP) v nalezeném výchozím BPB a přepočítáme řádek pro $z_k - c_k$. Tímto dostaneme výchozí simplexovou tabulku pro původní úlohu (LP) a dále pokračuje simplexovým algoritmem, přičemž pokud nám zůstaly v bázi ještě nějaké umělé proměnné (s nulovou hodnotou), pak v dalších krocích musíme hlídat, aby jejich hodnota nebyla kladná, což by rozbilo přípustnost aktuálního BPB.

Příklad 2.8.1

Dvoufázová metoda #1:    <https://goo.gl/FtYjXx>

Příklad 2.8.2

Dvoufázová metoda #1:    <https://goo.gl/yyjasa>

\triangle vektor c uvedený v čase 0:56 má být $(1, -2, 0, 0)^T$, ovšem nemá to žádný vliv na další výpočty

Příklad 2.8.3

Dvoufázová metoda #3: 📄 🗨️ 🖨️ <https://goo.gl/wuW5t8>

Příklad 2.8.4

Dvoufázová metoda #4: 📄 🗨️ 🖨️ <https://goo.gl/xoSKvB>**2.9** Lineární lomené programování

Pomocí úlohy lineárního programování umíme snadno maximalizovat zisk, tj. rozdíl mezi příjmem a náklady. Např. je-li příjem 1100 Kč a náklady 1000 Kč, pak náš zisk činí 100 Kč, což můžeme chápat jako měřítko *absolutní efektivity*⁷⁰ našeho snažení. Ovšem v některých situacích může být mnohem výhodnější sledovat a maximalizovat *relativní efektivity*, tj. podíl zisku a nákladů, což v uvedeném příkladě bylo 10 %. Reálné situace samozřejmě nejsou takto černobílé, neboť založit podnikání na aktivitě s efektivitou 30 % zní sice hezky, ale taky by se mohlo stát, že jsme tohoto čísla dosáhli při příjmu 100 Kč a nákladech 30 Kč.

Toto byla ukázka jednoduché úlohy *lineárního lomeného programování*, ve které chceme minimalizovat podíl dvou afinních funkcí na (části daného) polyedru, tj.

$$f(x) := \frac{\langle c, x \rangle + \alpha}{\langle d, x \rangle + \beta} \rightarrow \min \quad \& \quad x \in X := \{x \in \mathbb{R}^n \mid Ax \leq b \ \& \ \langle d, x \rangle + \beta > 0\} \quad (\text{LLP})$$

kde $x, c, d \in \mathbb{R}^n$, $b \in \mathbb{R}^m$, $A \in \mathbb{R}^{m \times n}$ a $\alpha, \beta \in \mathbb{R}$. Obecně nelze úlohu (LLP) uvažovat na množině $\{x \in \mathbb{R}^n \mid Ax \leq b\}$, neboť může obsahovat nulové body jmenovatele. Současně ale nelze vyloučit pouze body splňující $\langle d, x \rangle + \beta \neq 0$, protože bychom v takovém případě rozbili konvexnost přípustné množiny. Proto je v množině X zvolen dodatečný požadavek na kladnost jmenovatele, díky němuž je účelová funkce definována na celém X a současně je tato množina konvexní. Samozřejmě by bylo možné uvažovat v této podmínce i opačnou nerovnost, čímž bychom dostali druhou část polyedru.

Řešení úlohy (LLP) je ekvivalentní s úlohou lineárního programování ve tvaru

$$\left. \begin{array}{l} \langle c, y \rangle + \alpha t \rightarrow \min \\ Ay \leq bt \quad \& \quad \langle d, y \rangle + \beta t = 1 \quad \& \quad t \geq 0 \end{array} \right\} \quad (2.9.1)$$

v proměnných y a t . Vskutku, nejdříve si všimněme, že je-li $x \in X$, pak pro dvojici

$$y = \frac{x}{\langle d, x \rangle + \beta} \quad \text{a} \quad t = \frac{1}{\langle d, x \rangle + \beta} \quad (2.9.2)$$

platí, že $t > 0$ a

$$Ax - bt = \frac{1}{\underbrace{\langle d, x \rangle + \beta}_{>0}} \underbrace{(Ax - b)}_{\leq 0} \leq 0, \quad \langle d, y \rangle + \beta t = \frac{\langle d, x \rangle}{\langle d, x \rangle + \beta} + \frac{\beta}{\langle d, x \rangle + \beta} = 1,$$

tj. tato dvojice y, t je přípustný bod úlohy (2.9.1). Jelikož současně máme

$$\langle c, y \rangle + \alpha t = \frac{\langle c, x \rangle}{\langle d, x \rangle + \beta} + \frac{\alpha}{\langle d, x \rangle + \beta} = f(x),$$

⁷⁰ tak tohle asi není to nejlepší slovní spojení...

tj. hodnota účelové funkce úlohy (2.9.1) pro takovou dvojici y, t je stejná jako hodnota účelové funkce úlohy (LLP) v bodě x , takže nutně (optimální) hodnota úlohy (LLP) je rovna nebo větší než (optimální) hodnota úlohy (2.9.1). Naopak, je-li dvojice y, t s $t > 0$ přípustná pro (2.9.1), pak $x = y/t$ je přípustný bod úlohy (LLP) s toutéž hodnotou účelové funkce, protože

$$Ax = \frac{1}{t} \underbrace{Ay}_{\leq bt} \leq b \quad \text{a} \quad \langle d, x \rangle + \beta = \frac{1}{t} \langle d, y \rangle + \beta = \frac{1}{t} > 0$$

$$f(x) = f(y/t) = \frac{\frac{1}{t} \langle c, y \rangle + \alpha}{\frac{1}{t} \langle d, y \rangle + \beta} = \frac{\langle c, y \rangle + \alpha t}{\underbrace{\langle d, y \rangle + \beta t}_1} = \langle c, y \rangle + t\alpha.$$

Je-li ale dvojice y, t s $t = 0$ přípustná pro (2.9.1) a x_0 nějaký přípustný bod úlohy (LLP), pak $x = x_0 + \rho y$ je také přípustný bod úlohy (LLP) pro libovolné $\rho \geq 0$, protože

$$Ax = \underbrace{Ax_0}_{=b} + \rho \underbrace{Ay}_{\leq 0} \leq b \quad \text{a} \quad \langle d, x \rangle + \beta = \underbrace{\langle d, x_0 \rangle + \beta}_{>0} + \rho \underbrace{\langle d, y \rangle}_{=1} > 0,$$

a navíc pro účelovou funkci $f(x)$ platí

$$\begin{aligned} \lim_{\rho \rightarrow \infty} f(x) &= \lim_{\rho \rightarrow \infty} f(x_0 + \rho y) = \lim_{\rho \rightarrow \infty} \frac{\langle c, x_0 + \rho y \rangle + \alpha}{\langle d, x_0 + \rho y \rangle + \beta} = \lim_{\rho \rightarrow \infty} \frac{\langle c, x_0 \rangle + \alpha + \rho \langle c, y \rangle}{\langle d, x_0 \rangle + \beta + \rho \underbrace{\langle d, y \rangle}_{=1}} = \\ &= \lim_{\rho \rightarrow \infty} \left(\underbrace{\frac{\langle c, x_0 \rangle + \alpha}{\langle d, x_0 \rangle + \beta + \rho}}_{>0} + \frac{\langle c, y \rangle}{\frac{1}{\rho} (\langle d, x_0 \rangle + \beta) + 1} \right) = \langle c, y \rangle = \langle c, y \rangle + \alpha t, \end{aligned}$$

tj. můžeme najít přípustné body úlohy (LLP) s hodnotami účelové funkce libovolně blízkými hodnotě účelové funkce úlohy (2.9.1) v bodě y, t . Odtud vyplývá, že hodnota úlohy (LLP) je menší nebo rovna hodnotě úlohy (2.9.1)⁷¹, což dohromady s první částí potvrzuje platnost uvedené ekvivalence mezi úlohami (LLP) a (2.9.1). Transformace (2.9.2) se také někdy nazývá *Charnesova-Cooperova* podle autorů článku z roku 1962, ve kterém byla použita pro důkaz jisté analogie právě uvedené ekvivalence. Toto tvrzení lze rozšířit také na účelovou funkci

$$\max_{i=1, \dots, r} \frac{\langle c_i, x \rangle + \alpha_i}{\langle d_i, x \rangle + \beta_i}.$$

⁷¹ Nastává-li minimum úlohy (2.9.1) v bodě y, t s $t > 0$, pak v X umím najít bod se stejnou funkční hodnotou. Je-li minimum pro $t = 0$, pak v X je posloupnost bodů, pro kterou účelová funkce konverguje k téže hodnotě.

Příklad 2.9.1

(*Von Neumannův problém růstu*) Uvažme ekonomiku s n -tíci segmentů a úrovní aktivity $x_i > 0$ v i -tém segmentu a v současném časovém období, zatímco v příštím časovém období bude úroveň aktivity rovna $x_i^+ > 0$ (zajímá nás pouze jedno další období). Je spotřebováno m výrobků, které jsou v závislosti na úrovni aktivity vyráběny. Úroveň aktivity x znamená $Bx \in \mathbb{R}^m$ spotřebovaných výrobků a výrobu Ax výrobků, přičemž v dalším období nelze spotřebovat více než kolik se v současném období vyrobí, tj. $Bx^+ \leq Ax$. *Míra růstu i -tého segmentu* je pak dána podílem x_i^+ / x_i . Von Neumannův problém růstu je úloha pro nalezení aktivity x , která maximalizuje minimum míry růstu mezi všemi segmenty, tj. jedná se o úlohu lineárního lomeného programování ve tvaru

$$\min_{i=1, \dots, n} \left\{ \frac{x_i^+}{x_i} \right\} \rightarrow \max \quad \& \quad Bx^+ \leq Ax \quad \& \quad x^+ \geq 0$$

a doplněná o požadavek $\{[x, x^+] \mid x > 0\}$. Jelikož se jedná o homogenní problém v x a x^+ , je možné nahradit implicitní omezení $x > 0$ explicitní podmínkou $x \geq 1$.

KAPITOLA 3

Celočíselné programování

3.1 <i>ZLP v kostce (nebo obdélníku)</i>	96
3.2 <i>Závěrečné poznámky za milion dolarů</i>	96

SNAD ANI PEJSEK S KOČIČKOU BY PŘI PEČENÍ SVÉHO SLAVNÉHO DORTU NEDOKÁZALI NAVÁŽIT $\frac{14212}{28715}$ KG MOUKY. PODOBNĚ NEREÁLNÉ BY TO BYLO S PENĚZI. EXISTUJE ASI JEN VELMI MALÁ SKUPINA LIDÍ (NEPOCHYBNĚ ALE ZAHRNUJÍCÍ VĚTŠINU BÝVALÝCH MINISTRŮ FINANCÍ ČR), KTEŘÍ BY DOKÁZALI PŘI SKLÁDÁNÍ OPTIMÁLNÍHO STÁTNÍHO ROZPOČTU VYUŽÍT $\sqrt{2\pi + e^2}$ KČ.

3.1 ZLP v kostce (nebo obdélníku)

Úvod: 📄 🗣️ 🖥️ <https://goo.gl/n2Zs2C>

Aplikace: 📄 🗣️ 🖥️ <https://goo.gl/koL42H>

Metoda větvení a mezí – řešený příklad #1 & #2: 📄 🗣️ 🖥️ <https://goo.gl/f6jnyY>

Metoda větvení a mezí – řešený příklad #3 (binární problém): 📄 🗣️ 🖥️ <https://goo.gl/qg2kkQ>

Metoda řezů – řešený příklad: 📄 🗣️ 🖥️ <https://goo.gl/7A28hk>

⚠️ řádek pro a_4 v čase 35:00 má být 0,0,0,1,1,-3/2,1 (vždyt' vznikl pouze vynásobením řádku pro a_4 v předchozí tabulce číslem -3/2)

3.2 Závěrečné poznámky za milion dolarů

$$P \stackrel{?}{=} NP$$

KAPITOLA 4

Kvadratické programování

4.1 Úvod a motivace	98
4.2 Když stačí tužka a pravítko... ..	103
4.3 Teoretické základy	107
4.4 Hildrethova–d’Esopova metoda	111
4.5 Wolfeho metoda v krátkém tvaru	114
4.6 Wolfeho metoda v dlouhém tvaru	117

¹ Riziko portfolia není lineární funkcí vektoru jeho složek. Spíše než to je rozptýl portfolia kvadratickou funkcí jeho skladby. To boří intuici většiny analytiků a investorů. Vskutku podstata rizika může být jediným nejdůležitějším argumentem pro použití kvantitativní analýzy v řízení investic. Za tuto skutečnost nelze obviňovat ani investory, ani analytiky. Ani Harry Markowitz za to nemůže. Příroda učinila riziko kvadratickou funkcí. Markowitz to pouze objevil.

² William Forsyth Sharpe (* 16. června 1934) je americký ekonom, který působí jako emeritní profesor na Stanfordově univerzitě. V roce 1990 získal společně Harrym Markowitzem a Mertonem Millerem Cenu Švédské národní banky za rozvoj ekonomické vědy na památku Alfreda Nobela za jejich „průkopnickou práci v oblasti finanční ekonomie“. Je také jedním z tvůrců modelu oceňování kapitálových aktiv. Je autorem tzv. *Sharpeho poměru* pro měření rizikově očištěného výnosu, přispěl k rozvoji binomiální metody oceňování opcí, gradientní metody pro optimalizaci alokace aktiv.

THE RISK OF A PORTFOLIO IS NOT A LINEAR FUNCTION OF THE VECTOR OF ITS COMPONENTS. RATHER, THE VARIANCE OF A PORTFOLIO IS A QUADRATIC FUNCTION OF ITS COMPOSITION. THIS THWARTS THE INTUITION OF MOST ANALYSTS AND INVESTORS. INDEED, THE NATURE OF RISK MAY BE THE SINGLE MOST IMPORTANT ARGUMENT FOR THE USE OF QUANTITATIVE ANALYSIS IN INVESTMENT MANAGEMENT. NEITHER INVESTORS NOR ANALYSTS CAN BE BLAMED FOR THIS FACT. NOR CAN HARRY MARKOWITZ. NATURE MADE RISK A QUADRATIC FUNCTION. MARKOWITZ ONLY DISCOVERED IT.¹

WILLIAM SHARPE²

Viz Sharpeho webovou stránku <https://odkaz.page.link/sharpe1>.

4.1 Úvod a motivace

V předchozích kapitolách jsme viděli, že úlohy s lineární účelovou funkcí mají velmi široký záběr. Samozřejmě ale nejsou všeobíjající. V praktických úlohách můžeme najít i úlohy s kvadratickou účelovou funkcí, tj.

$$\left. \begin{aligned} f(x) &:= \frac{1}{2} \langle Cx, x \rangle + \langle d, x \rangle \rightarrow \min \\ x \in X &:= \{x \in P \mid \langle a_i, x \rangle \leq b_i, i = 1, \dots, k \ \& \ \langle a_i, x \rangle = b_i, i = k + 1, \dots, m\} \\ P &:= \{x \in \mathbb{R}^n \mid x_1 \geq 0 \ \& \dots \ \& \ x_s \geq 0, s \in \{0, 1, \dots, n\}\}. \end{aligned} \right\} \quad (\text{QP})$$

kde $C \in \mathbb{R}^{n \times n}$, $d \in \mathbb{R}^n$, $a_1, \dots, a_m \in \mathbb{R}^n$, $b_1, \dots, b_m \in \mathbb{R}$ jsou dané matice, vektory a čísla. Navíc nedílnou součástí úlohy (QP) je požadavek symetričnosti a pozitivní semidefinitnosti matice C , tj. $C = C^T \geq 0$. Díky tomuto požadavku máme opět úlohu konvexního programování, což nám umožňuje využít některé teoretické výsledky z Kapitoly ???. Úloha (QP) se nazývá úlohou *kvadratického programování* a jedná se o zobecnění úlohy (2.2.1) odpovídající volbě $C = 0$.

Ačkoli úlohy kvadratického programování byly ve druhé polovině 50. let a následně i v 60. letech minulého století velmi populární (z této doby jsou také základy algoritmů, které si ukážeme), není jim v dnešní době věnována tak velká pozornost jako v případě lineárního programování. Důvod je především ten, že pouze málo úloh získaných jako matematické modely optimalizačních problémů z praxe splňuje podmínku konvexnosti/konkávnosti (tj. $C \geq 0$ či $C \leq 0$). Naším cílem bude vybudování dvou algoritmů pro řešení úlohy (QP): (i) první z nich je založen na duální úloze – bude početně celkem snadný (dokonce bude sám korigovat případně početní chyby), ale bude vyžadovat $C > 0$; (ii) druhý bude založen na simplexovém algoritmu a Karushových–Kuhnových–Tuckerových podmínkách – ten bude početně náročnější, ale bude fungovat pro libovolnou matici $C \geq 0$. Ještě předtím si ale ukážeme tři klasické úlohy vedoucí na problémy kvadratického programování.

(i) Regrese (metoda nejmenších čtverců s omezeními)

Chceme najít přímku, která nejlépe (tj. v normě) aproximuje daná data, přičemž o některých parametrech a priori víme, že splňují jistá omezení. Uvažme lineární regresní model

$$y = \beta_0 + \beta_1 x_1 + \dots + \beta_k x_k,$$

kde β_0, \dots, β_k jsou *regresní koeficienty*. Označme y_i jako výsledek i -tého pozorování závislé proměnné při hodnotách x_{ij} nezávislých proměnných, tj.

$$y_i = \beta_0 + \beta_1 x_{i1} + \dots + \beta_k x_{ik} + e_i, \quad i = 1, \dots, n,$$

kde e_i je chyba, která popisuje odchylku pozorování od skutečné regrese (tj. hledané podmíněné střední hodnoty). K tomu, abychom docílili nejlepšího možného modelu, je potřeba odhadnout β_0, \dots, β_k tak, aby byla chyba nejmenší možná. Toho bychom mohli docílit pouze pomocí absolutní hodnoty³, tj. řešením úlohy

$$\sum_{i=1}^n |e_i| \rightarrow \min.$$

Taková funkce je sice konvexní a my bychom dokázali takovou úlohu převést na problém lineárního programování, jenže to by hledání koeficientů β_0, \dots, β_n výrazně zkomplikovalo (a v dobách před objevem simplexového algoritmu v podstatě znemožnilo). Obvyklým nástrojem pro řešení této úlohy je derivování, což ale v případě absolutní hodnoty naráží na její nediferencovatelnost.

³ Chceme stejně „vážit“ kladné i záporné odchylky, proto s absolutní odchylkou. Bez ní by se nám odchylky s opačnými znaménky anulovaly.

Proto se k určení koeficientů regresní přímky užívá *metoda nejmenších čtverců*, ve které chceme minimalizovat sumu druhých mocnin jednotlivých chyb, tj.

$$\sum_{i=1}^n e_i^2 = \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_{i1} - \dots - \beta_k x_{ik})^2 \rightarrow \min,$$

což je kvadratická funkce vzhledem k proměnným β_0, \dots, β_k . S využitím vektorů $\beta \in \mathbb{R}^{k+1}$, $Y \in \mathbb{R}^n$ a matici $X \in \mathbb{R}^{n \times (k+1)}$ daných jako

$$\beta := \begin{pmatrix} \beta_0 \\ \vdots \\ \beta_k \end{pmatrix}, \quad Y := \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}, \quad X := \begin{pmatrix} 1 & x_{11} & \dots & x_{1k} \\ 1 & x_{21} & \dots & x_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \dots & x_{nk} \end{pmatrix}$$

lze tuto úlohu zapsat ve formě minimalizace euklidovské normy

$$\|Y - X\beta\|^2 \rightarrow \min \quad \text{neboli} \quad (Y - X\beta)^\top (Y - X\beta) = \beta^\top X^\top X \beta - \beta^\top X^\top Y - Y^\top X \beta + Y^\top Y \rightarrow \min,$$

což po zanedbání konstantního členu $Y^\top Y$ odpovídá účelové funkci z (QP) s volbou $C = 2X^\top X$ a $d = -2Y^\top X$, která jistě splňuje podmínku pozitivní semidefinitnosti matice C . V praxi je však mnohdy nutné přidat lineární omezení pro některé koeficienty β_i , tj. např.

$$\sum_{j=0}^k a_{rj} \beta_j = b_r, \quad r = 1, \dots, m$$

$$\beta_0 \geq 0 \quad \& \quad \beta_1 \geq 0 \quad \& \quad \dots \quad \& \quad \beta_k \geq 0$$

pro daná čísla $a_{10}, \dots, a_{mk} \in \mathbb{R}$, což je přesně úloha (QP) v kanonickém tvaru. Nebo případně můžeme požadovat, aby hodnoty β_j byly pouze v nějakém omezeném rozsahu, tj. $\beta_j \in [\alpha_j, \gamma_j]$.

Příklad 4.1.1

Uvažme m -tici demografických skupin, které chceme oslovit reklamou, přičemž požadovaný počet shlédnutí/zaujetí/„líbítek“ v každé skupině je určen vektorem $v^{\text{pož}}$ s kladnými koeficienty. K tomuto účelu využijeme n -tici kanálů vhodných pro šíření reklamy (různé webové platformy, televize, rádio, tisk, ...). Matice $\mathcal{R} \in \mathbb{R}^{m \times n}$ vyjadřuje vztah mezi zhlédnutími v jednotlivých skupinách a 1 Kč investovanou do jednotlivých kanálů, tj. \mathcal{R}_{ij} udává počet zhlédnutí v i -té skupině při investování 1 Kč do j -tého reklamního kanálu (tyto hodnoty jsou samozřejmě odhadnuty zcela kvalifikovaně a jsou vždy kladné). Pak j -tý sloupec matice \mathcal{R} udává efektivitu nebo dosah j -tého kanálu za 1 Kč, zatímco i -tý řádek ukazuje, jak jsou které kanály exponované v i -té skupině.

Celkový počet zhlédnutí v každé demografické skupině vektor $v = Rs \in \mathbb{R}^m$, kde $s \in \mathbb{R}^n$ popisuje investice do jednotlivých kanálů. Cílem pak je nalezení právě vektoru s , aby platilo $v = Rs \approx v^{\text{pož}}$, k čemuž využijeme metodu nejmenších čtverců, tj. chceme najít s , pro které

$$\|Rs - v^{\text{pož}}\|^2 \rightarrow \min.$$

V takovém případě ale nemůžeme a priori zaručit nezápornost složek vektoru s a také splnění našeho rozpočtového omezení. Toho docílíme požadavkem

$$s_1 + \dots + s_n = B,$$

⁴ Chceme vlastně optimálně rozdělit naše dostupné zdroje (finance) mezi jednotlivé kanály.

kde B určuje celkový dostupný rozpočet⁴, tj. celkem dostáváme úlohu

$$\begin{aligned} \|Rs - v^{\text{pož}}\|^2 &\rightarrow \min \\ s_1 + \dots + s_n &= B \\ s_1 \geq 0 \quad &\& \quad \dots \quad &\& \quad s_n \geq 0. \end{aligned}$$

Úloha může mít i výrazně fyzikální charakter.

Příklad 4.1.2

Uvažme 10-rozměrný vektor f , který popisuje posloupnost sil aplikovaných vždy po dobu 1 sekundy na hmotný bod jednotkové hmotnosti na přímce bez tření. Na počátku máme nulovou rychlost a pozici. Podle Newtonových zákonů⁵ bude finální rychlost a pozice

$$\begin{aligned} v^F &= f_1 + \dots + f_{10} \\ p^F &= \frac{19}{2} f_1 + \frac{17}{2} f_2 + \dots + \frac{1}{2} f_{10}. \end{aligned}$$

Naším úkolem je najít takovou posloupnost, aby $v^F = 0$ a $p^F = 1$, tj. posloupnost sil, které zanechají hmotný bod nehybný v pozici 1 m vpravo. Takových sil samozřejmě existuje mnoho, např.

$$f^{bb} = \{1, -1, 0, \dots, 0\},$$

kteřá akceleruje hmotný bod na průměrnou rychlost 0,5 m/s po 1 sekundě a potom jej během druhé sekundy zpomalí na nulovou rychlost, čímž dosáhneme právě pozice 1 m vpravo. Dalším aplikováním nulových sil dosáhneme požadovaného cíle. Horní index „bb“ odkazuje na tzv. *bang-bang*, což znamená, že nejdříve aplikujeme velkou sílu, abychom hmotný bod rozpochybovali (první „bang“), a poté jinou velkou silou (druhé „bang“), abychom jej zpomalili na nulovou rychlost, viz obrázky XXX⁶.

My bychom ale rádi zjistili, jaká „nejmenší“ posloupnost sil povede k $v^F = 0$ a $p^F = 1$, kde „nejmenší“ je určeno součtem čtverců jednotlivých aplikovaných sil, tj.

$$\|f\|^2 = f_1^2 + \dots + f_{10}^2.$$

Toto je vlastně *podmíněný problém nejmenší normy*⁷

$$\begin{aligned} \|f\|^2 &\rightarrow \min \\ \begin{pmatrix} 1 & 1 & \dots & 1 \\ 19/2 & 17/2 & \dots & 1/2 \end{pmatrix} f &= \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{aligned}$$

s hledaným vektorem $f = (f_1, f_2, \dots, f_{10})^\top$. Řešením je vektor/posloupnost sil

$$f^* = (9/165, 7/165, 5/165, 3/165, 1/165, -1/165, -3/165, -5/165, -7/165, -9/165)^\top,$$

viz obrázek XXX⁸. Hodnota $\|f^*\|^2 = 2/165$, což je ve výrazném kontrastu se čtvercem normy „bang-bang“ posloupnosti $\|f^{bb}\|^2 = 2$, která je 165-krát větší.

⁵ Platí $F = ma$, takže vzhledem k jednotkové hmotnosti máme rovnost mezi zrychlením a působící silou $F = a$. Protože $s(t) = v_0 t + \frac{1}{2} at^2$ a $v(t) = v_0 + at$, kvůli nulové počáteční rychlosti dostaneme aplikováním síly f_1 v čase $t = 0$ po jedné sekundě rychlost $v(1) = a = f_1$. Aplikováním síly f_2 v čase $t = 1$, kdy už máme rychlost f_1 , bude rychlost $v(2) = v_1 + af_2 = f_1 + f_2$ atd. Současně po první sekundě urazíme dráhu $s(1) = f_1/2$. Po další sekundě to už bude $s(2) = f_1/2 + f_1 + f_2/2 = 3f_1/2 + f_2/2$ a o další sekundu později $s(3) = f_1/2 + f_1 + f_2/2 + (f_1 + f_2) + f_3/2 = 5f_1/2 + 3f_2/2 + f_3/2$ atd.

⁶ OBRAZEK

⁷ To je speciální případ metody nejmenších čtverců, kde máme $Y = 0$ a $\beta_0 = 0$, v kterémžto případě můžeme vzít $X = I$.

⁸ OBRAZEK

Ještě si můžeme uvést jednu praktickou úlohu stejného charakteru.

Příklad 4.1.3

Masarykova univerzita by ráda odhadla zaměstnanost svých absolventů. Pro jednoduchost se omezme pouze na Přírodovědeckou fakultu a připusťme, že každý absolvent má zaměstnání pouze ve veřejném sektoru, průmyslu nebo v akademické sféře. Ignorujeme nezaměstnanost, která beztak u absolventů PřF jistě ani není. Označme symbolem a_j počet absolventů v j -tém roce pro $j = 1, \dots, n$ a v_j, p_j, z_j počty absolventů v j -tém roce, kteří nastoupí do veřejné sféry, průmyslu nebo na akademickou pozici, tj. $a_j = v_j + p_j + z_j$. Jeden z uvažovaných modelů předpokládá, že jistý zlomek populace studentů se vždy připojí k zaměstnanecké kategorii v každém roce. Označme tyto zlomky jako λ_1, λ_2 a λ_3 . Potom odhadované hodnoty nástupů do jednotlivých kategorií zaměstnání v j -tém roce jsou dány jako

$$\widehat{v}_j = \lambda_1 a_j \quad \& \quad \widehat{p}_j = \lambda_2 a_j \quad \& \quad \widehat{z}_j = \lambda_3 a_j.$$

Smysluplným měřítkem pro platnost modelu pak může být rozdíl mezi aktuálními hodnotami v_j, p_j, z_j a předpovězenými hodnotami $\widehat{v}_j, \widehat{p}_j, \widehat{z}_j$ jako v metodě nejmenších čtverců, tj.

$$\sum_{j=1}^n [(v_j - \widehat{v}_j)^2 + (p_j - \widehat{p}_j)^2 + (z_j - \widehat{z}_j)^2] \rightarrow \min$$

za předpokladu, že všichni absolventi jsou zaměstnání v jedné ze tří uvažovaných kategorií. Tím se dostáváme k úloze kvadratického programování v proměnných $\lambda_1, \lambda_2, \lambda_3$ mající podobu

$$\sum_{j=1}^n [(v_j - \lambda_1 a_j)^2 + (p_j - \lambda_2 a_j)^2 + (z_j - \lambda_3 a_j)^2] \rightarrow \min$$

$$\lambda_1 + \lambda_2 + \lambda_3 = 3$$

$$\lambda_1 \geq 0 \quad \& \quad \lambda_2 \geq 0 \quad \& \quad \lambda_3 \geq 0.$$

(ii) Support Vector Machines (SVM)

Jedná se o metodu tzv. *strojového učení*, což je podoblast *umělé inteligence*. Je dána m -tice bodů v \mathbb{R}^n , z nichž každý patří do jedné z tříd označených $+1$ a -1 , tj. je dána množina dvojic $\{x_i, y_i\} \in \mathbb{R}^n \times \{\pm 1\}$ pro $i = 1, \dots, m$. Naším cílem je nalézt nadrovinu v \mathbb{R}^n , která odděluje body z obou tříd, tj. hledáme $a \in \mathbb{R}^n$ a $b \in \mathbb{R}$ takové, že

$$\langle a, x_i \rangle - b < 0, \quad \text{je-li odpovídající } y_i = -1,$$

$$\langle a, x_i \rangle - b > 0, \quad \text{je-li odpovídající } y_i = +1.$$

Jelikož se tyto nerovnosti nezmění po vynásobení dvojice (a, b) libovolným kladným číslem, můžeme ekvivalentně říci, že hledáme a, b splňující

$$\langle a, x_i \rangle - b \leq -1, \quad \text{je-li odpovídající } y_i = -1,$$

$$\langle a, x_i \rangle - b \geq 1, \quad \text{je-li odpovídající } y_i = +1$$

neboli

$$y_i (\langle a, x_i \rangle - b) \geq 1 \quad \text{pro všechna } i = 1, \dots, m.$$

° OBRAZEK

Body jsou odděleny pásem $\{x \in \mathbb{R}^n \mid -1 \geq \langle a, x \rangle - b \geq 1\}$, tj. obrázek XXX⁹

V úloze SVM pak chceme nalézt nejen oddělující nadrovinu, ale i maximalizovat šířku tohoto

¹⁰ Portfolio Selection Problem

¹¹ Toto je velmi úzce spojeno především s prací Harryho Markowitz.

¹² Harry Max Markowitz (* 24. srpna 1927) je americký ekonom. V roce 1989 získal *John von Neumann Theory Prize* za jeho příspěvky k rozvoji tří oblastí: teorie portfolia, metod pro řídké matice v lineárním programování a simulačního programovacího jazyku SIMSCRIPT, viz [55, 56, 58–60]. O rok později obdržel společně Williamem Sharpem a Mertonem Millerem *Cenu Švédské národní banky za rozvoj ekonomické vědy na památku Alfreda Nobela* za jejich „průkopnickou práci v oblasti finanční ekonomie“. V současnosti je profesorem na Kalifornské univerzitě v San Diegu. Je znám především díky jeho práci věnované moderní teorii portfolia. Tu započal články ze šedesátých let, kdy pracoval ve společnosti RAND. Zde se setkal s Dantzigem a s jeho pomocí pokračoval ve studiu optimalizačních metod a dalším rozvoji algoritmu určení kritické přímky pro nalezení portfolia s optimální hodnotou rozptylu, přičemž využíval toho, co je dnes známo jako *Markowitzova hranice*. V roce 1954 obdržel doktorát na Univerzitě v Chicagu s disertační prací věnovanou právě teorii portfolia. Toto téma bylo tak nové, že během obhajoby práce se s ním Milton Friedman dohadoval o tom, zda jeho příspěvek je vůbec o ekonomii. Během let 1955–1956 pobýval na *Cowlesově nadaci (Cowles Foundation)* na Yaleově univerzitě. V roce 1956 publikoval článek ohledně kritické přímky a čas strávený v nadaci využil k napsání knihy o teorii portfolia, která vyšla v roce 1959. Během 50. let dospěl Markowitz stejně jako mnoho ostatních k názoru, že mnohé praktické problémy se vymykají analytickému řešení a že jsou potřebné simulační techniky. Ve společnosti RAND se proto účastnil vytváření velkých logistických modelů simulace. Počátkem 60. let vytvořil programovací jazyk, který byl později pojmenován SIMSCRIPT. Místo toho, aby popisoval kroky, které musí počítač učinit, aby uskutečnil simulaci, tento jazyk umožnil programátorovi popsat určitým stylizovaným způsobem systém, který měl být simulován, což omezilo čas potřebný na programování.

pásu, která je rovna $2/\|a\|$. Ovšem to je stejné jako

$$\|a\|^2 = \langle a, a \rangle \rightarrow \min$$

za podmínek

$$y_i(\langle a, x_i \rangle - b) \geq 1 \quad \text{pro všechna } i = 1, \dots, m,$$

což je opět úloha kvadratického programování.

(iii) Tvorba portfolia^{10,11,12}

Asi nejdůležitějším příkladem úloh kvadratického programování je tvorba portfolia. Ten získáme zobecněním úlohy lineárního programování, viz Příklad 2.2.3, jestliže jednotlivé koeficienty c_1, \dots, c_n budeme považovat spíše za náhodné veličiny (původně udávaly hodnotu nebo výnos z držení i -tého aktiva). Mají-li tyto náhodné veličiny střední hodnotu

$$d = (d_1, \dots, d_n)^T$$

a kovarianční matici

$$C = \begin{pmatrix} c_{11} & \dots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{n1} & \dots & c_{nn} \end{pmatrix},$$

pak v takovém případě nemá smysl maximalizovat $\langle c, x \rangle$, neboť to je náhodná veličina. Kdybychom při daných lineárních omezeních maximalizovali střední hodnotu $\langle d, x \rangle$, bylo by výsledné rozhodnutí x^* značně nespolehlivé, neboť by bylo odvozeno bez ohledu na variabilitu veličin c_1, \dots, c_n . Proto je mnohem výhodnější snažit se omezit rozptyl $\langle Cx, x \rangle$ náhodné veličiny $\langle c, x \rangle$ a maximalizovat funkci

$$f(x) := a \langle d, x \rangle - b \langle Cx, x \rangle,$$

kde $a, b \geq 0$ jsou váhy, které vyjadřují důležitost, jaká se při rozhodování přikládá vyšší účelové funkce (koeficient a) a spolehlivosti rozhodnutí (koeficient b). Dokonce se stačí omezit pouze na váhy splňující $a + b = 1$.

Příklad 4.1.4

Investor má k dispozici 1 milion korun, který může využít pro nákup 3 druhů akcií. Označme S_i náhodnou veličinu reprezentující roční výnos z 1 Kč investované do i -té akcie, tj. je-li $S_i = 0, 12$, pak 1 Kč investovaná do této akcie na začátku roku by měla na konci téhož roku přinést nejhůře 1,12 Kč. Máme následující informace středních hodnot, rozptyl a kovariance náhodných veličin

$$E(S_1) = 0,14 \quad \& \quad E(S_2) = 0,11 \quad \& \quad E(S_3) = 0,1$$

$$\text{Var}(S_1) = 0,2 \quad \& \quad \text{Var}(S_2) = 0,08 \quad \& \quad \text{Var}(S_3) = 0,18$$

$$\text{Cov}(S_1, S_2) = 0,05 \quad \& \quad \text{Cov}(S_1, S_3) = 0,02 \quad \& \quad \text{Cov}(S_2, S_3) = 0,03.$$

Sestavte úlohu kvadratického programování, jejímž řešením nalezneme skladbu portfolia, které dosáhne alespoň 12% ročního zhodnocení a bude mít minimální rozptyl ročního zisku.

Řešení. Označme jako x_i množství peněz [v tisících Kč] investované do i -té akci pro $i =$

1, 2, 3. Roční výnos takového portfolia pak je

$$\frac{x_1 S_1 + x_2 S_2 + x_3 S_3}{1000}$$

a očekávaný roční výnos je

$$\frac{x_1 E(S_1) + x_2 E(S_2) + x_3 E(S_3)}{1000},$$

neboť $E(X+Y) = E(X) + E(Y)$ a $E(aX) = aE(X)$. K tomu, aby očekávaný výnos byl alespoň 12 %, musíme požadovat

$$0,14x_1 + 0,11x_2 + 0,1x_3 \geq 1000 \times 0,12 = 120.$$

Současně musí také platit $x_1 + x_2 + x_3 = 1000$ a samozřejmě¹³ $x_1 \geq 0$, $x_2 \geq 0$, $x_3 \geq 0$. Naším cílem je za těchto podmínek minimalizovat rozptyl portfolia, tj.

$$\begin{aligned} \text{Var}(x_1 S_1 + x_2 S_2 + x_3 S_3) &= \text{Var}(x_1 S_1) + \text{Var}(x_2 S_2) + \text{Var}(x_3 S_3) + \\ &+ 2\text{Cov}(x_1 S_1, x_2 S_2) + 2\text{Cov}(x_1 S_1, x_3 S_3) + 2\text{Cov}(x_2 S_2, x_3 S_3) = \\ &= 0,2x_1^2 + 0,08x_2^2 + 0,18x_3^2 + 0,1x_1x_2 + 0,04x_1x_3 + 0,06x_2x_3, \end{aligned}$$

což je úloha (QP) s $d = 0$.

¹³ Nepovolujeme žádný krátký prodej (short sale), tj. spekulace na pokles ceny.

4.2 Když stačí tužka a pravítko...

Stejně jako úlohy lineárního programování i úlohy (QP) lze v případě $n = 2$ (a s nemalým úsilím i pro $n = 3$) řešit pomocí náčrtku přípustné množiny a vrstevnic účelové funkce. Zatímco zakreslení přípustné množiny je stejné jako v případě úloh lineárního programování, s vrstevnicemi už to je trochu složitější. Jak mohou vrstevnice vypadat? Tentokrát lineární část funkce f určuje pouze jejich posunutí v \mathbb{R}^2 a hlavní roli hraje kvadratická část, takže to bude některá ze známých kuželoseček, jejichž rovnice v kanonickém tvaru jsou

elipsy: $\frac{x_1^2}{a^2} + \frac{x_2^2}{b^2} = 1 \quad a, b > 0$

hyperboly: $\frac{x_1^2}{a^2} - \frac{x_2^2}{b^2} = 1 \quad a, b > 0$

paraboly: $x_1^2 + 2px_2 = 0 \quad p \neq 0$

imaginární rovnoběžky a bod $[0, 0]$: $\frac{x_1^2}{a^2} + \frac{x_2^2}{b^2} = 0 \quad a, b > 0$

dvojice různoběžek: $\frac{x_1^2}{a^2} - \frac{x_2^2}{b^2} = 0 \quad a, b > 0$

dvojice rovnoběžek: $x_1^2 - \alpha^2 = 0 \quad \alpha > 0$

dvojnásobná přímka: $x_1^2 = 0$

My hledáme vrstevnice na různých úrovních, tj. $f(x) = \alpha$ pro $\lambda \in \mathbb{R}$, což znamená, že v závislosti na hodnotě matice C může nastat pouze některá z následujících situací:

- (i) pro některé hodnoty $\alpha \in \mathbb{R}$ to jsou elipsy, které se „zhroutlí“ na bod, a pak už jen prázdné množiny; toto je možné tehdy a jen tehdy, když $C > 0$ (nebo $C < 0$, což ale neuvažujeme);
- (ii) pro všechna $\alpha \in \mathbb{R}$ to jsou paraboly;
- (iii) pro některá $\alpha \in \mathbb{R}$ to jsou dvojice rovnoběžek, které se „zhroutlí“ na dvojnásobnou přímku, a pak už jen prázdné množiny.

Vzhledem k požadavku $C \geq 0$ není situace

- (iv) pro některé hodnoty $\alpha \in \mathbb{R}$ se jedná o hyperboly, které se „zhroutlí“ na bod, a pak už jen prázdné množiny;

možná, neboť pro hyperboly by matice C musela být indefinitní. To ale neznamená, že nemůžeme uvažovat i takovou úlohu kvadratického programování, kde C je indefinitní nebo negativně (semi-)definitní. „Jenom“ funkce f nebude konvexní, takže nebudeme mít už vůbec zaručenu existenci řešení a nebude možné použít metody, které si za chvíli vybudujeme. Navíc z předchozí části víme, že takový problém bude NP-úplný.

Pokud v účelové funkci je smíšený člen $x_1 x_2$ a nejedná se úplný čtverec, je analýza konkrétního tvaru vrstevnic výrazně komplikovanější než v případě úloh lineárního programování. Máme k tomu víceméně 3 možnosti:

- (a) diagonalizujeme kvadratickou část účelové funkce a dopočteme vše potřebné;
- (b) předchozí postup je početně velmi náročný; mnohem jednodušší je pracovat s celou účelovou funkcí jako kvadratickou funkcí v proměnných $(x_1, x_2, 1)^T$, což nás přivede k matici

$$Q := \begin{pmatrix} \tilde{Q} & \frac{1}{2}d \\ \frac{1}{2}d^T & \text{konst.} - \alpha \end{pmatrix} \in \mathbb{R}^{(n+1) \times (n+1)},$$

kde $\tilde{Q} := \frac{1}{2}C$ je skutečnou maticí kvadratické formy a „konst.“ značí případný absolutní člen ve funkci f . Pak platí následující klasifikace

- $\text{rank}(Q) = 3$ & $\text{rank}(\tilde{Q}) = 2$: elipsy (hyperboly) s kanonickou rovnicí

$$\lambda_1 y_1^2 + \lambda_2 y_2^2 + \frac{\det Q}{\det \tilde{Q}} = 0;$$

- $\text{rank}(Q) = 3$ & $\text{rank}(\tilde{Q}) = 1$: paraboly s kanonickou rovnicí

$$\lambda_1 y_1^2 \pm 2 \sqrt{-\frac{\det Q}{\lambda_1}} y_2 = 0;$$

- $\text{rank}(Q) = 2$ & $\text{rank}(\tilde{Q}) = 2$: různoběžky s kanonickou rovnicí

$$\lambda_1 y_1^2 + \lambda_2 y_2^2 = 0;$$

- $\text{rank}(Q) = 2$ & $\text{rank}(\tilde{Q}) = 1$: rovnoběžky s kanonickou rovnicí

$$\lambda_1 y_1^2 + \lambda_2 y_2^2 = 0;$$

- $\text{rank}(Q) = 1$ & $\text{rank}(\tilde{Q}) = 1$: dvojnásobná přímka s kanonickou rovnicí

$$\lambda_1 y_1^2 = 0.$$

(c) úprava kvadratické části tak, abychom odstranili smíšený člen (vhodnou transformací do nových souřadnic, které nemusí být ortogonální – obvykle totiž nepotřebujeme znát zcela přesně tvar vrstevnic v původních souřadnicích).

Na rozdíl od úloh lineárního programování zde již neplatí to, že hledaný extrém se realizuje v nějakém vrcholu množiny X či na nějaké jeho hraně. Toto si ilustrujeme v několika následujících jednoduchých příkladech.

Příklad 4.2.1

Uvažme

$$x_1^2 + x_2^2 - 4x_1 - 4x_2 + 1 \rightarrow \min$$

na množinách

$$X_1 := \{x \in \mathbb{R}^2 \mid x_1 \in [0, 4] \text{ \& } x_2 \in [0, 2]\},$$

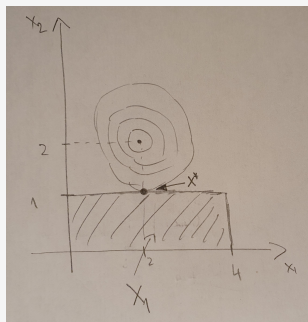
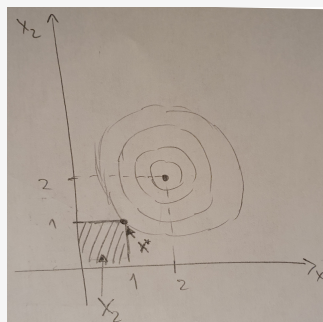
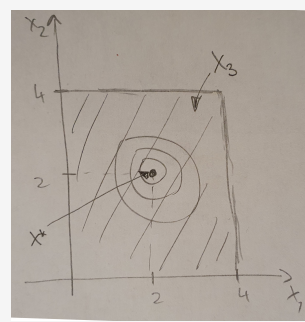
$$X_2 := \{x \in \mathbb{R}^2 \mid x_1 \in [0, 1] \text{ \& } x_2 \in [0, 1]\},$$

$$X_3 := \{x \in \mathbb{R}^2 \mid x_1 \in [0, 4] \text{ \& } x_2 \in [0, 4]\}.$$

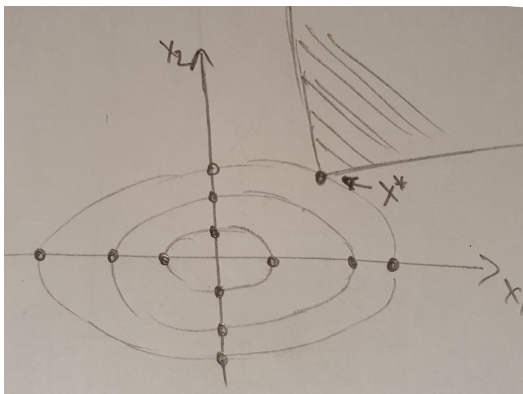
Vrstevnice účelové funkce jsou určeny rovností

$$x_1^2 + x_2^2 - 4x_1 - 4x_2 + 1 = \alpha \quad \text{neboli} \quad (x_1 - 2)^2 + (x_2 - 2)^2 = \alpha + 7,$$

což pro $\alpha < -7$ jsou prázdné množiny, pro $\alpha = -7$ to je bod $[2, 2]$ a pro $\alpha > -7$ se jedná o kružnici se středem v bodě $[2, 2]$ a poloměrem $\sqrt{\alpha + 7}$, které se pro $\alpha \rightarrow \infty$ rozpínají. Na množině X_1 je řešením bod $x_1^* = [2, 1]$ s hodnotou $f_1^* = -6$, na množině X_2 to je bod $x_2^* = [1, 1]$ s hodnotou $f_1^* = -5$ a konečně na množině X_3 to je bod $x_3^* = [2, 2]$ s hodnotou $f_1^* = -7$, viz Obrázky 4.1.a–4.1.c. V prvním případě jde o bod ležící na hraně množiny X_1 (ovšem už to není celá hrana), ve druhém případě to je vrchol množiny X_2 a ve třetím případě jde o vnitřní bod přípustné množiny X_3 .

Obrázek 4.1.a: Vrstevnice a množina X_1 .Obrázek 4.1.b: Vrstevnice a množina X_2 .Obrázek 4.1.c: Vrstevnice a množina X_3 .

Už jsme se zmínili, že vrstevnice mohou být i elipsy (což vlastně zahrnuje i kružnice jako velmi speciální případ). Jednou z jejich velmi důležitých charakteristik jsou poloosy, které mimo jiné určují vrcholy elipsy (průnik elipsy a poloos). Nicméně je asi celkem zřejmé, že neexistuje žádná souvislost mezi těmito vrcholy a řešením příslušné úlohy (QP), viz Obrázek 4.2.



Obrázek 4.2: Úloha (QP) a vrcholy elipsy.

Ve všech předchozích případech bylo řešení vždy jediné. Mohlo by jich být nekonečně mnoho? Jsou-li vrstevnice elipsy (kružnice), tak to možné není. Jelikož se to stane právě tehdy, když matice C je pozitivně definitní, je taková účelová funkce *ostře konvexní*, a tedy nutně existuje nejvýše jediné globální minimum na množině X . Ovšem v případě singulární matice $C \geq 0$ už řešení *může* být nekonečně mnoho, jak ukazuje následující příklad¹⁴.

¹⁴ Obecně lze ukázat, že řešení bude nejvýše jediné kdykoli soustava

$$Cz = 0 \quad \& \quad d^T z = 0$$

nemá nenulové řešení.

Příklad 4.2.2

Uvažme

$$x_1^2 + 2x_1x_2 + x_2^2 \rightarrow \min$$

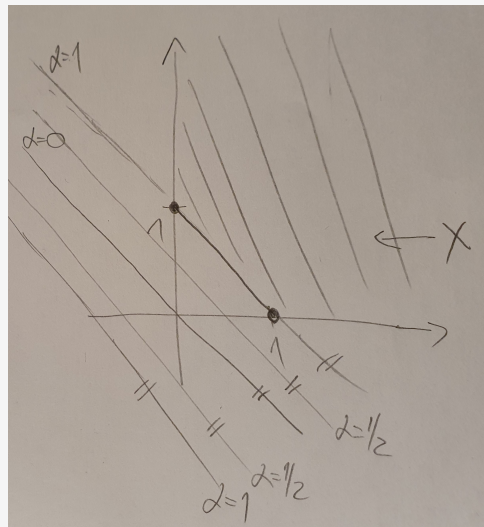
na množině

$$X := \{x \in \mathbb{R}^2 \mid x_1 + x_2 \geq 1 \ \& \ x_1 \geq 0 \ \& \ x_2 \geq 0\}.$$

Účelová funkce odpovídá matici $C = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$ splňující $C \geq 0$ a $\det C = 0$. Vrstevnice jsou dány rovnicí

$$x_1^2 + 2x_1x_2 + x_2^2 = \alpha \quad \text{neboli} \quad (x_1 + x_2)^2 = \alpha,$$

takže pro $\alpha < 0$ to jsou prázdné množiny, pro $\alpha = 0$ máme přímkou $x_2 = -x_1$ a pro $\alpha > 0$ máme dvojici rovnoběžných přímek $x_2 = \pm\sqrt{\alpha} - x_1$. Řešením pak je celá úsečka spojující body $[1, 0]$ a $[0, 1]$, viz Obrázek 4.3.



Obrázek 4.3: Úloha (QP) a $C \geq 0$ s nekonečně mnoha řešeními.

Ve zbývajících třech příkladech se podíváme na složitější situace, ve kterých již kvadratická část bude obsahovat i smíšený člen a zároveň se nebude jednat o úplný čtverec.

Příklad 4.2.3

Grafické řešení – řešený příklad #1 (elipsy):    <https://goo.gl/5i9Wx6>

Příklad 4.2.4

Grafické řešení – řešený příklad #2 (paraboly):    <https://goo.gl/DdVbau>

Příklad 4.2.5

Grafické řešení – řešený příklad #3 (přímky):    <https://goo.gl/zHwFEX>

¹⁵ Toto je jedno z velmi mála míst dosavadního výkladu, kde by se hodila znalost některých pojmů z konvexní analýzy v Kapitole ???. Nicméně pro tuto chvíli stačí s jistou mírou představivosti alespoň to, co jsme si připomněli ve Větě 1.2.6. Ačkoli tam žádné nerovnosti ani podmínky nezápornosti nebyly.

4.3 Teoretické základy

Napíšeme-li vektory a_1, \dots, a_m z (QP) jako řádky matice A a $b := (b_1, \dots, b_m)^\top$, pak lze jednotlivá omezení získat pomocí součinu Ax a vektoru b , takže k této úloze můžeme přiřadit (regulární) Lagrangeovu funkci¹⁵

$$L(x, y) := \frac{1}{2} \langle Cx, x \rangle + \langle d, x \rangle + \langle y, Ax - b \rangle = \frac{1}{2} \langle Cx, x \rangle + \langle A^\top y + d, x \rangle - \langle y, b \rangle,$$

kde $y \in Q := \{y = (y_1, \dots, y_m)^\top \in \mathbb{R}^m \mid y_1 \geq 0 \ \& \ \dots \ \& \ y_k \geq 0\}$ jsou *Lagrangeovy multiplikátory*. Funkce $L(x, y)$ je jistě diferencovatelná a není těžké se přesvědčit o tom, že

$$\text{grad}_x L(x, y) = Cx + A^\top y + d.$$

Proto díky požadavku $C \geq 0$ vyplývá z Karushových–Kuhnových–Tuckerových podmínek v diferenciálním tvaru následující tvrzení, viz Větu XXX¹⁶.

¹⁶ ODKAZ**VĚTA 4.3.1**

Bod $x^* \in X$ je řešením úlohy (QP) právě tehdy, když existuje $y^* \in Q$ takové, že platí

$$\langle Cx^* + A^\top y^* + d, x - x^* \rangle \geq 0 \quad \text{pro všechna } x \in P, \quad (4.3.1)$$

$$\langle y, Ax^* - b \rangle = 0. \quad (4.3.2)$$

Analogicky postupu známému z lineárního programování lze libovolnou úlohu (QP) převést do libovolného do libovolného z dříve zmíněných speciálních tvarů, přičemž opět platí, že situace *in* \mathbb{R}^n a $Ax = b$ je pro nás nezajímavá. Ovšem tentokrát se tímto přechodem může změnit ta nejdůležitější charakteristika úlohy – definitnost matice kvadratické formy v účelové funkci¹⁷.

¹⁷ PODROBNOSTI

A jak v těchto speciálních případech vypadají podmínky (4.3.1)–(4.3.2)?

- (i) Standardní tvar: $P = \{x \in \mathbb{R}^n \mid x \geq 0\}$ a $X = \{x \in P \mid Ax \leq b\}$. Pak (4.3.1) je splněna právě tehdy, když $Cx^* + A^\top y^* + d \geq 0$ a zároveň

$$\langle Cx^* + A^\top y^* + d, x^* \rangle = 0.$$

Podmínka (4.3.2) zůstává beze změny. V takovém případě je řešení úlohy (QP) ekvivalentní s nalezením řešení soustavy

$$Cx + A^\top y - u = -d$$

$$Ax + v = b$$

$$\langle x, u \rangle + \langle y, v \rangle = 0$$

$$x \geq 0 \ \& \ u \geq 0 \ \& \ y \geq 0 \ \& \ v \geq 0.$$

- (ii) Kanonický tvar: $P = \{x \in \mathbb{R}^n \mid x \geq 0\}$ a $X = \{x \in P \mid Ax = b\}$. Pak podmínka (4.3.1) je opět splněna právě tehdy, když $Cx^* + A^\top y^* + d \geq 0$ a zároveň

$$\langle Cx^* + A^\top y^* + d, x^* \rangle = 0,$$

zatímco podmínka (4.3.2) je splněna vždy. V takovém případě je řešení úlohy (QP) ekvivalentní s nalezením řešení soustavy

$$Cx + A^\top y - u = -d$$

$$Ax = b$$

$$\langle x, u \rangle = 0$$

$$x \geq 0 \ \& \ u \geq 0.$$

- (iii) Základní tvar: $P = \mathbb{R}^n$ a $X = \{x \in P \mid Ax \leq b\}$. Pak (4.3.1) se redukuje na

$$Cx^* + A^\top y^* + d = 0,$$

zatímco podmínka (4.3.2) zůstává beze změny. V takovém případě je řešení úlohy (QP) ekvivalentní s nalezením řešení soustavy

$$\begin{aligned}Cx + A^\top y + d &= 0 \\Ax + v &= b \\ \langle y, v \rangle &= 0 \\ y \geq 0 \quad \& \quad v \geq 0.\end{aligned}$$

Nyní se blíže podíváme na duální úlohu v případě přípustné množiny dané v základním tvaru. Pro ni je $Q = \mathbb{R}_+^m$ a duální úloha, která má v obecném případě podobu

$$\varphi(y) \rightarrow \max \quad \& \quad y \in Q \quad \& \quad \varphi(y) := \inf_{x \in P} L(x, y),$$

bude v takovém případě mít tvar

$$\frac{1}{2} \langle Cx, x \rangle + \langle A^\top y + d, x \rangle - \langle y, b \rangle \rightarrow \max \quad \& \quad Cx + A^\top y + d = 0 \quad \& \quad y \geq 0$$

což díky druhé podmínce může zapsat jako

$$\varphi(y) = -\frac{1}{2} \langle Cx, x \rangle - \langle y, b \rangle \rightarrow \max \quad \& \quad Cx + A^\top y + d = 0 \quad \& \quad y \geq 0. \quad (4.3.3)$$

Kdyby bylo $C > 0$, lze z druhé podmínky přímo vyjádřit x a dosadit do účelové funkce, čímž bychom jej vyeliminovali z účelové funkce a měli bychom duální úlohu pouze v proměnné y . Toto ale v případě $\det C = 0$ nemusí být možné, takže duální úloha má spíše výše uvedený tvar s proměnnými x a y . Mimochodem ona druhá podmínka plyne z faktu, že hodnotu

$$\varphi(y) = \inf_{x \in \mathbb{R}^n} \left(\frac{1}{2} \langle Cx, x \rangle + \langle d, x \rangle - \langle y, Ax - b \rangle \right)$$

nalezneme přímo derivováním vzhledem k x , tj. infimum se realizuje ve stacionárním bodě určeném právě rovnicí

$$Cx + A^\top y + d = 0.$$

Podobně můžeme ukázat, že v případě úlohy v kanonickém tvaru má duální úloha podobu

$$-\frac{1}{2} \langle Cx, x \rangle - \langle y, b \rangle \rightarrow \max \quad \& \quad Cx + A^\top y + d \geq 0$$

a pro úlohu ve standardním tvaru obdržíme

$$-\frac{1}{2} \langle Cx, x \rangle - \langle y, b \rangle \rightarrow \max \quad \& \quad Cx + A^\top y + d \geq 0 \quad \& \quad y \geq 0.$$

Poznámka 4.3.2.

- (i) Je zřejmé, že duální úloha k úloze (QP) bude vždy opět úloha kvadratického programování.
- (ii) Volbou $C = 0$ dostaneme dvojice primárních a duálních úloh známé z Kapitoly 2, tj. tyto dvojice jsou speciálními případy dvojic primárních a duálních úloh pro (QP).
- (iii) Tentokrát ale uvedené dvojice duálních úloh nevykazují žádnou symetrii. Té lze docílit pomocí úlohy kvadratického programování ve tvaru

$$\frac{1}{2} \langle Cx, x \rangle + \langle d, x \rangle + \frac{1}{2} \langle Dz, z \rangle \rightarrow \min \quad \& \quad Ax + Dz \geq -b \quad \& \quad x \geq 0,$$

kde C a D jsou symetrické a pozitivně semidefinitní matice, viz [16]. Pak odpovídají duální úlohou je

$$-\frac{1}{2} \langle Cx, x \rangle - \langle y, b \rangle - \frac{1}{2} \langle Dy, y \rangle \rightarrow \min \quad \& \quad Cx - A^\top z \geq -b \quad \& \quad y \geq 0,$$

takže v porovnání primární a duální úlohy lze nalézt jistou formu symetrie. Opět speciální volbou $C = 0 = D$ dostaneme dvojici symetrických úloh lineárního programování.

Vraťme se nyní zpět k úloze (4.3.3) a připusťme, že matice C je regulární. Pak z druhé podmínky dostaneme

$$x = -C^{-1}(A^T y + d),$$

což po dosazení do účelové funkce v této duální úloze dává

$$\begin{aligned} \varphi(y) &= -\frac{1}{2} \langle A^T y + d, C^{-1}(A^T y + d) \rangle - \langle y, b \rangle = \\ &= -\frac{1}{2} \langle A^T y, C^{-1} A^T y \rangle - \frac{1}{2} \langle A^T y, C^{-1} d \rangle - \frac{1}{2} \langle d, C^{-1}(A^T y + d) \rangle - \langle y, b \rangle = \\ &= -\frac{1}{2} \langle AC^{-1} A^T y, y \rangle - \langle AC^{-1} d + b, y \rangle - \frac{1}{2} \langle C^{-1} d, d \rangle = \\ &= -\frac{1}{2} \langle Gy, y \rangle - \langle h, y \rangle + \gamma, \end{aligned}$$

kde

$$G := AC^{-1}A^T, \quad h := AC^{-1}d + b, \quad \gamma := -\frac{1}{2} \langle C^{-1}d, d \rangle.$$

Jelikož konstanta v účelové funkci nemá vliv na řešení úlohy, můžeme duální úlohu v tomto případě zapsat jako

$$-\frac{1}{2} \langle Gy, y \rangle - \langle h, y \rangle \rightarrow \max, \quad y \geq 0$$

neboli

$$\frac{1}{2} \langle Gy, y \rangle + \langle h, y \rangle \rightarrow \min, \quad y \geq 0. \quad (4.3.4)$$

Jelikož $C > 0$, pak také $C^{-1} > 0$, což implikuje $G \geq 0$ a navíc diagonální prvky matice G splňují $g_{ii} = \langle C^{-1}a_i, a_i \rangle > 0$ pro $i = 1, \dots, m$.

Ještě než z těchto úvah odvodíme první metodu pro řešení úloh kvadratického programování, potřebujeme znát vztah mezi primární a duální úlohou. To opět vyplývá z obecné teorie duality v matematickém programování a platí pro všechny dvojice, které jsme před chvílí odvodili.¹⁸

¹⁸ ODKAZ nebo DŮKAZ

VĚTA 4.3.3 SLABÁ VĚTA O DUALITĚ

Uvažme (libovolnou) dvojici primární a duální úlohy kvadratického programování. Je-li $x \in X$ a (z, y) je přípustný bod duální úlohy, pak platí

$$\frac{1}{2} \langle Cx, x \rangle + \langle d, x \rangle \geq -\frac{1}{2} \langle Cz, z \rangle - \langle y, b \rangle.$$

Odtud (opět) vyplývají následující skutečnosti.

- (i) Jsou-li obě úlohy přípustné, pak obě mají také řešení.
- (ii) Je-li v některé z těchto úloh účelová funkce neomezená (zdola/shora), pak druhá úloha je nepřípustná.

VĚTA 4.3.4 SILNÁ VĚTA O DUALITĚ

Pro (libovolnou) dvojici primární a duální úlohy kvadratického programování platí:

- (i) Je-li x^* řešením primární úlohy, pak existuje $y^* \in Q$ takové, že $[x^*, y^*]$ je řešením duální úlohy.
- (ii) Je-li $[x^*, y^*]$ řešením duální úlohy a zároveň $C > 0$, pak x^* je řešením primární úlohy.

V obou případech přitom platí rovnost

$$\min \left(\frac{1}{2} \langle Cx, x \rangle + \langle d, x \rangle \right) = \max \left(-\frac{1}{2} \langle Cz, z \rangle - \langle y, b \rangle \right).$$

Tvrzení Věty 4.3.4(ii) lze ještě více zesílit tak, že vlastně obdržíme podobné tvrzení jako v úloze lineárního programování.

VĚTA 4.3.5

Pro (libovolnou) dvojici primární a duální úlohy kvadratického programování platí:

- (i) Má-li jedna z úloh řešení, pak má řešení i druhá úloha.
- (ii) Je-li jedna z úloh nepřipustná, pak druhá úloha je buď také nepřipustná nebo účelová funkce je na přípustné množině neohraničená (zdola/shora).

Pro dvojice primární a duální úlohy kvadratického programování tedy platí stejná tvrzení jako v případě lineárního programování. Vlastně lze říci, že teorie duality lineárního programování je speciálním případem teorie duality kvadratického programování.

4.4 Hildrethova²⁵–d’Esopova²⁶ metoda

²⁴ Clifford George Hildreth (8. prosince 1917 – 15. srpna 1995) byl americký ekonom. Byl také vedoucím Ústavu ekonomie na Michiganské státní univerzitě. V roce 1973 byl prezidentem Americké statistické asociace (American Statistical Association).

²² Žádné podrobnější informace o něm nemáme. Jeho profesní dráha nejspíše šla jiným než matematickým směrem, neboť se mi pod jeho jménem podařilo dohledat pouze 3 odborné články.

První metoda, kterou si ukážeme, je určena pro úlohu kvadratického programování v základním tvaru

$$\frac{1}{2} \langle Cx, x \rangle + \langle d, x \rangle \rightarrow \min, \quad Ax \leq b \quad (4.4.1)$$

s maticí $C > 0$. V předchozí části jsme ukázali, že příslušná duální úloha je ekvivalentní s úlohou (4.3.4), která je vždy přípustná. Potom z části (i) poznámky za Větou 4.3.3 plyne, že je-li také úloha (4.4.1) přípustná, pak obě mají řešení. V takovém případě plyne z Věty 4.3.4(ii), že je-li y^* řešením úlohy (4.3.4), pak $x^* := -C^{-1}(A^T y^* + d)$ je řešením primární úlohy (4.4.1). Toto řešení je kvůli podmínce $C > 0$ jediné, tj. pro libovolné řešení úlohy (4.3.4) dostaneme totéž x^* .

Princip algoritmu je velmi jednoduchý. Začneme v nějaké bodě $y^{[0]}$ a bod $y^{[1]} = [y_1^{[1]}, \dots, y_m^{[1]}]$ získáme tak, že postupně řešíme m minimalizačních úloh s podmínkou $y_i^{[1]} \geq 0$, přičemž (obvykle) postupujeme od $i = 1$ až k $i = m$ a v každém kroku považujeme ostatní proměnné za konstanty. Podrobněji si algoritmus popíšeme za chvíli.

Ještě se ale pozastavme nad jedním problémem souvisejícím s tímto iterativním procesem. Jeho pomocí získáme body $y^{[1]}, \dots$, které jsou přípustnými pro duální úlohu (4.3.4) a aproximují řešení y^* . Nabízí se proto otázka, kdy odpovídající $x^{[1]} := -C^{-1}(A^T y^{[1]} + d)$, atd. bude přípustným bodem úlohy (4.4.1)?

LEMMA 4.4.1

Je-li $y \geq 0$ a $Gy + h \geq 0$, potom

$$x = -C^{-1}(A^T y + d) \quad (4.4.2)$$

je přípustným bodem úlohy (4.4.1).

Důkaz. Vždyť platí

$$b - Ax = b + AC^{-1}(A^T y + d) = Gy + h. \quad \blacksquare$$

Toto nám dává jednoduché kritérium přípustnosti přibližného řešení x určeného vztahem (4.4.2)

²³ To ale v mnoha praktických úlohách nemusí představovat žádné úskalí, neboť A a b mohou být dány jen s určitým stupněm přesnosti (odhad atd.).

bez toho, aniž bychom samotné x vypočítali. Je-li totiž $(Gy + h)_i = -\varepsilon_i < 0$, pak platí $\langle a_i, x \rangle = b_i + \varepsilon_i$, tj. $\varepsilon_i > 0$, $i = 1, \dots, m$, určuje „míru porušení“ i -tého omezení primární úlohy (4.4.1). Jinými slovy, jsou-li $\varepsilon_1, \dots, \varepsilon_m$ předem dané tolerance porušení jednotlivých omezení²³ úlohy (4.4.1), pak při řešení úlohy (4.3.4) stačí najít $y \geq 0$ takové, že

$$Gy + h + \varepsilon \geq 0.$$

Pak odpovídající x bude ε -přípustným bodem úlohy (4.4.1). Ještě jinak řečeno, potřebujeme, aby existovaly takové přípustné body úlohy (4.4.1), že

$$\langle a_i, x \rangle \leq b_i - \delta, \quad i = 1, \dots, m$$

pro $\delta > 0$. To ale není nic jiného než tzv. *Slaterova podmínka*²⁴, která zaručuje, že úloha (4.4.1) je tzv. *regulární* úlohou konvexního programování, a tedy i konvergenci Hildrethovy–d'Esopovy metody.

Snadno si můžeme všimnout, že volné minimum účelové funkce úlohy (4.4.1) se realizuje v bodě

$$\tilde{x} = -C^{-1}d.$$

Tedy vztah $x = C^{-1}(A^T y + d)$ určuje volné minimum \tilde{x} v případě $\tilde{y} = (0, \dots, 0)^T$. Navíc platí

$$b - A\tilde{x} = b + AC^{-1}(A^T \cdot 0 + d) = AC^{-1}d + b = h,$$

tj. složky vektoru h určují „odchylku“ bodu \tilde{x} od $Ax = b$, přičemž h_i určuje „odchylku“ pro $\langle a_i, x \rangle = b_i$. Pokud $h \geq 0$, pak $\tilde{y} = 0$ je řešením úlohy (4.3.4), neboť $G \geq 0$, takže \tilde{x} je řešením úlohy (4.4.1). V opačném případě „nastupuje“ následující algoritmus.

Krok 0 (inicializace): Zvolíme nějaké $y^{[0]} \geq 0$. Obvykle se volí $y^{[0]} = 0$, jelikož ve výsledku bývá některá z proměnných nulová (vzhledem k předchozímu komentáři). V dalším kroku je potřeba vypočítat $y^{[1]}$ atd., tj. určit *minimalizující posloupnost* $\{y^{[k]}\}_{k=0}^*$. Výpočet se zastaví ve chvíli, kdy je dosaženo požadované přesnosti mezi dvěma následujícími členy posloupnosti $\{y^{[k]}\}$, neboť platí $\rho(y^{[k]}) \leq \rho(y^{[k-1]})$, tj. posloupnost $\{\rho(y^{[k]})\}_{k=0}^*$ je nerostoucí, kde ρ je účelová funkce úlohy (4.3.4), tj.

$$\rho(y) := \frac{1}{2} \langle Gy, y \rangle + \langle h, y \rangle.$$

Mějme již vypočteny $y^{[0]}, y^{[1]}, \dots, y^{[k-1]}$. Nyní chceme vypočítat $y^{[k]} := (y_1^{[k]}, \dots, y_m^{[k]})^T$. To se provede pomocí následujícího schématu.

Krok 1 (výpočet $y_1^{[k]}$): Řešíme úlohu

$$\rho(t, y_2^{[k]}, \dots, y_m^{[k]}) \rightarrow \min, \quad t \geq 0.$$

Tato funkce ρ je kvadratickou (konvexní) funkcí jedné proměnné t . Proto minimum nastává buď ve stacionárním bodě nebo pro $t = 0$, tj. $y_1^{[k]} = \max\{0, t_1^{[k]}\}$, kde $t_1^{[k]}$ značí stacionární bod, tj. řešení úlohy

$$\frac{d}{dt} \rho(t, y_2^{[k]}, \dots, y_m^{[k]}) = 0 \quad \text{neboli} \quad g_{11}t + \sum_{j=2}^m g_{1j} y_j^{[k-1]} + h_1 = 0,$$

což dává

$$t_1^{[k]} = -\frac{1}{g_{11}} \left(\sum_{j=2}^m g_{1j} y_j^{[k-1]} + h_1 \right)$$

Krok i (výpočet $y_i^{[k]}$, $i = 2, \dots, m$): Řešíme úlohu

$$\rho(y_1^{[k]}, \dots, y_{i-1}^{[k]}, t, y_{i+1}^{[k]}, \dots, y_m^{[k]}) \rightarrow \min, \quad t \geq 0.$$

Opět dostáváme $y_i^{[k]} = \max\{0, t_i^{[k]}\}$, kde $t_i^{[k]}$ je stacionární bod, tj.

$$t_i^{[k]} = -\frac{1}{g_{ii}} \left(\sum_{j=1}^{i-1} g_{ij} y_j^{[k]} + h_i + \sum_{j=i+1}^m g_{ij} y_j^{[k-1]} \right)$$

V jednotlivých iteracích je možné zaměnit pořadí výpočtu $1, 2, \dots, m$ -té složky $y^{[k]}$. Princip metody se tím nikterak nezmění, ale může to mít vliv na konvergenci metody, která obvykle není příliš rychlá (pochopitelně to velmi závisí na počáteční volbě $y^{[0]}$). Jistou výhodou této metody je „autokorekce“ případných numerických chyb a zaokrouhlování. Jednotlivé body $y^{[k]}$ lze totiž považovat za nové výchozí body bez ohledu na „historii“ jeho vzniku $y^{[0]}, \dots, y^{[k-1]}$. Navíc při výpočtu není nutné ihned počítat všechny iterace úplně přesně. Díky zmíněné autokorekční vlastnosti stačí začít s jedním desetinným místem a další přidat až v případě, kdy se hodnota ustálí atd. Toto zjednodušení oceníme obzvláště při ručních výpočtech. Na druhou stranu jednou z výrazných nevýhod této metody je nutnost výpočtu C^{-1} , což může být početně velmi náročné – obzvláště v případě, kdy hodnota $\det C$ je blízká nule.

Realizace výpočtu: Danou převedeme do základního tvaru a určíme (4.3.4), tj. vypočteme matici G a vektor h . Výpočet je pak realizován pomocí jednoduché tabulky. Do záhlaví napíšeme jednotlivé složky matice G a vektoru h tak, že všechny členy na stejném řádku podělíme diagonálním prvkem, změníme znaménko a místo diagonálního prvku napíšeme 0. Do prvního řádku tabulky zapíšeme výchozí $y^{[0]}$ a číslo 1. Potom počítáme $y_1^{[1]}, y_2^{[1]}, \dots$ tak, že pro jejich i -tou složku potřebujeme i -tý řádek záhlaví, jehož složkami postupně vynásobíme předchozích $m - 1$ předchozích prvků (včetně těch, které jsou o řádek výše). Jednotlivé součiny sečteme a přičteme poslední prvek i -tého řádku v záhlaví. Je-li výsledek kladný zapíšeme jej jako $y_i^{[k]}$, v opačném případě zapíšeme 0. Na konci každého řádku pak ještě napíšeme 1. Následující tabulka ilustruje výpočet $y_3^{[k]}$ v případě $m = 4$.

$i = 3$	0	$-\frac{g_{12}}{g_{11}}$	$-\frac{g_{13}}{g_{11}}$	$-\frac{g_{14}}{g_{11}}$	$-\frac{h_1}{g_{11}}$
	$-\frac{g_{21}}{g_{22}}$	0	$-\frac{g_{23}}{g_{22}}$	$-\frac{g_{24}}{g_{22}}$	$-\frac{h_2}{g_{22}}$
	$-\frac{g_{31}}{g_{33}}$	$-\frac{g_{32}}{g_{33}}$	0	$-\frac{g_{34}}{g_{33}}$	$-\frac{h_3}{g_{33}}$
	$-\frac{g_{41}}{g_{44}}$	$-\frac{g_{42}}{g_{44}}$	$-\frac{g_{43}}{g_{44}}$	0	$-\frac{h_4}{g_{44}}$
	g_{44}	g_{44}	g_{44}	0	g_{44}
$y^{[0]}$	$y_1^{[0]}$	$y_2^{[0]}$	$y_3^{[0]}$	$y_4^{[0]}$	1
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
$y^{[k-1]}$	\vdots	\vdots	$y_3^{[k-1]}$	$y_4^{[k-1]}$	1
$y^{[k]}$	$y_1^{[k]}$	$y_2^{[k]}$	\odot		

Tabulka 4.1: Ilustrace jednoho kroku Hildrethovy–d’Esopovy metody.






Philip Starr „Phil“ Wolfe (11. srpna 1927 – 29. prosince 2016) byl americký matematik, který je považován za jednoho z „otců zakladatelů“ teorie konvexního programování. Na základní škole patřil mezi vynikající žáky a miloval vědu, ale na střední škole se pro něj stala překážkou na algebře a geometrii. Naštěstí si přečetl Euklidovy *Základy*, díky čemuž (jak sám Wolfe říkal) se z něj a Euklida „stali kolegové“. V roce 1943 nastoupil na Kalifornskou univerzitu v Berkeley, ale studium musel přerušit kvůli nástupu do armády krátce před koncem druhé světové války. V roce 1947 se na Berkeley vrátil a v roce 1954 zde získal doktorát, když se ve své disertační práci chtěl věnovat především teorii her. Tento zájem jej přivedl k Projektu SCOOP a k setkání s Dantzigem, který jej ponoukl k vyřešení problému s možným zacyklením simplexového algoritmu. To se Wolfe podařilo, viz [25], s využitím *lexikografického* pravidla, které se stalo součástí jeho disertační práce. Po zisku doktorátu přijal Wolfe pozici na univerzitě v Princetonu, kde se věnoval problémům kvadratického programování. Společně s Marguerite Frank(ovou) navrhli tzv. *Frankové-Wolfeho* algoritmus pro řešení úloh kvadratického programování, který lze aplikovat i na úlohy konvexního programování, viz [40]. Ve stejném čísle časopisu *Naval Research Logistics Quarterly* (vlastně hned na další straně) vyšel i Markowitzův článek [57] ohledně minimalizace kvadratické funkce s využitím kritické přímky. Před nástupem na Princeton dostal i nabídku ze společnosti RAND, kterou ale tehdy odmítl. V roce 1957 RAND svoji nabídku zdvojnásobil, a tak se Wolfe vrátil do Kalifornie, kde společně s Dantzigem a dalšími pracovali na zlepšení simplexového algoritmu. Z této doby je i jeho modifikace simplexového algoritmu pro řešení úloh kvadratického progra-

Číslo \odot pak má buď nulovou hodnotu nebo

$$y_3^{[k-1]} \cdot 0 - y_4^{[k-1]} \frac{g_{34}}{g_{33}} - 1 \cdot \frac{h_3}{g_{33}} - y_1^{[k]} \frac{g_{31}}{g_{33}} - y_2^{[k]} \frac{g_{32}}{g_{33}},$$

je-li tento výraz kladný.

Příklad 4.4.2

Hildrethova–D’Esopova metoda – řešený příklad #1:    <https://goo.gl/aPhpAJ>

4.5 Wolfeho metoda v krátkém tvaru

Jakkoli je princip předchozí metody snadný, je za něj velmi drazo zapláceno velmi silnými požadavky zahrnující $C > 0$ a Slaterovu podmínku. Je proto potřeba mít k dispozici i algoritmus pro řešení obecnějších úloh (QP), zejména dovolující i singulární matice $C \geq 0$. To nás přivádí k algoritmu Philipa Wolfeho²⁵ z roku 1959 založeném na Karushových–Kuhnových–Tuckerových podmínkách z poznámky (i) za Větou 4.3.1 a upraveném simplexovém algoritmu, viz [80]. Ačkoli podobně jako v případě lineárního programování dnes už existují efektivnější metody (ať již založené na simplexovém algoritmu nebo na zcela jiných základech), my u této metody zůstaneme kvůli jejímu přirozenému navázání na simplexový algoritmus a také její názornost (a celkem „snadný“ výpočet). Toto jsou vlastně také důvody ospravedlňující velké rozšíření a popularitu tohoto algoritmu v 60. letech. Tato metoda byly původně rozpracována pro úlohu kvadratického programování v kanonickém tvaru a její popis lze v takovém případě nalézt v řadě monografií a také ve skriptech [30, Sekce 7.5]. Jenže my se touto cestou nevydáme, a to minimálně ze dvou důvodů

- (i) potřebná modifikace simplexové metody ubírá této metodě na „čistotě“;
- (ii) většina (zajímavých) úloh v \mathbb{R}^2 , které umíme vizualizovat (a tedy řešit i geometricky), by vyžadovaly pro převod na kanonický tvar zavedení pomocných přidavných proměnných, což by nutně vedlo k použití tzv. *dlouhého tvaru* Wolfeho metody (odpovídající účelová funkce by byla vzhledem k novým proměnným singulární). Toto ale v našem pojetí Wolfeho metody potřeba nebude. Pro úplnost ještě dodejme, že bude-li původní úloha v základním tvaru, pak její řešení Wolfeho metodou bude vždy vyžadovat dlouhý tvar (bez ohledu na to zda použijeme původní algoritmus nebo ten „vylepšený“).

Naším výchozím bodem tedy je úloha kvadratického programování

$$\frac{1}{2} \langle Cx, x \rangle + \langle d, x \rangle \rightarrow \min, \quad Ax \leq b, \quad x \geq 0, \quad (4.5.1)$$

s maticí $C \geq 0$ a odpovídajícími Karushovými–Kuhnovými–Tuckerovými podmínkami

$$-Cx - A^T y + u = d \quad (4.5.2a)$$

$$Ax + \quad \quad v = b \quad (4.5.2b)$$

$$x \geq 0 \quad \& \quad y \geq 0 \quad \& \quad u \geq 0 \quad \& \quad v \geq 0 \quad (4.5.2c)$$

$$\langle x, u \rangle = 0 \quad \& \quad \langle y, v \rangle = 0 \quad (4.5.2d)$$

Připusťme nyní, že $b \geq 0$. V opačném případě bude opět nutná tzv. *první fáze* Wolfeho metody, kterou si ukážeme později. Najdeme-li řešení $[x^*, y^*, u^*, v^*]$ soustavy (4.5.2), pak odpovídající x^*

mování z [80]. V roce 1964 zavítal na pozvání svého přítele Gomoryho na půlroční stáž do společnosti IBM, do které následně nastoupil, a strávil v ní zbytek své profesní kariéry. V roce 1992 získal společně s Hoffmannem *John von Neumann Theory Prize* za „jejich příspěvky k intelektuálním základům matematického programování“ (their contributions to the intellectual foundations of mathematical programming). Na webových stránkách *INFORMS* je dokonce dostupný i záznam velmi zajímavého rozhovoru s Wolfem z května 2001, viz <https://odkaz.page.link/wolfe1>.

bude řešením úlohy (4.5.1) dle Věty 4.3.1 a následného rozboru. Podmínky 4.5.2a–4.5.2c jsou lineární, zatímco podmínka 4.5.2d je nelineární. Z poslední podmínky vyplývá, že z každého páru proměnných x_i, u_i a y_i, v_i je alespoň jedna složka nulová, a tedy nejvýše $n + m$ proměnných v $(x, u, y, v)^T \in \mathbb{R}^{2(n+m)}$ je nenulových. Avšak to je právě tolik, kolik má soustava (4.5.2a)–(4.5.2b) rovnic, tj. nejvýše tolik proměnných je nenulových, kolik má soustava rovnic. Jelikož platí, že má-li soustava lineárních rovnic nezáporné řešení, pak má i nezáporné bazické řešení, stačí se při řešení (4.5.2a)–(4.5.2d) omezit na nezáporné bazické řešení (tzv. *simplexový algoritmus*, srovnej s lineárním programováním).

Hlavní myšlenka dalšího postupu je následující (tzv. *druhá fáze* Wolfoho metody): Zavedením pomocných umělých nezáporných proměnných rozšíříme soustavu (4.5.2a)–(4.5.2c) do podoby, ve které bude snadné určit výchozí BPB splňující (4.5.2d). Potom simplexovým algoritmem anulujeme umělé proměnné, přičemž stále máme na paměti podmínku (4.5.2d). Pokud se nám takto podaří vyloučit všechny umělé proměnné z báze, pak konečná simplexová tabulka bude určovat řešení soustavy (4.5.2a)–(4.5.2d).

Nyní si to rozebereme trochu podrobněji. Soustavu (4.5.2a) rozšíříme o umělé proměnné $z \in \mathbb{R}^n$ tak, že dostaneme soustavu

$$\left. \begin{array}{l} -Cx - A^T y + u + Dz = d \\ Ax + \quad \quad v = b \\ x \geq 0 \quad \& \quad y \geq 0 \quad \& \quad u \geq 0 \quad \& \quad v \geq 0 \quad \& \quad z \geq 0, \end{array} \right\} \quad (4.5.3)$$

kde $D \in \mathbb{R}^{n \times n}$ je diagonální matice s diagonálními prvky

$$d_{ii} = \begin{cases} -1, & d_i < 0, \\ 1, & d_i \geq 0, \end{cases} \quad i = 1, \dots, n,$$

tj. $d_{ii} = \text{sgn } d_i$ pro $d_i \neq 0$. Toto lze maticově zapsat jako

$$\begin{pmatrix} -C & -A^T & I & 0 & D \\ A & 0 & 0 & I & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ u \\ v \\ z \end{pmatrix} = \begin{pmatrix} d \\ b \end{pmatrix}$$

$$n \left\{ \begin{array}{ccccc} & \overbrace{\quad}^n & \overbrace{\quad}^m & \overbrace{\quad}^n & \overbrace{\quad}^m & \overbrace{\quad}^n \\ -C & -A^T & I & 0 & D \\ A & 0 & 0 & I & 0 \end{array} \right\} \begin{pmatrix} x \\ y \\ u \\ v \\ z \end{pmatrix} = \begin{pmatrix} d \\ b \end{pmatrix}. \quad (4.5.4)$$

²⁶ Zde symbol $|d|$ značí bod $[|d_1|, \dots, |d_n|]$. „Bazickost“ uvedeného bodu plyne ihned z toho, že odpovídající část matice na levé straně (4.5.4) je $\begin{pmatrix} 0 & D \\ I & 0 \end{pmatrix}$, která má plnou hodnotu $n + m$.

Potom $[x, y, u, v, z] = [0, 0, 0, b, |d|]$ je BPB²⁶, který zároveň splňuje i (4.5.2d). Nyní za účelem anulování umělých proměnných řešíme úlohu

$$z_1 + \dots + z_n \rightarrow \min \quad \text{za podmínek (4.5.3)}. \quad (4.5.5)$$

Ovšem ne každé řešení úlohy (4.5.5) musí vyhovovat podmínce (4.5.2d). Pro je nutné doplnit simplexový algoritmus o

Wolfeho pravidlo: Je-li x_i (nebo y_i) bazickou proměnnou, pak při přechodu k další simplexové tabulce a novému BPB se nesmí u_i (nebo v_i) stát bazickou proměnnou a naopak, tj. je-li u_i (nebo v_i) v bázi, nesmí do báze vstoupit x_i (nebo y_i).

Toto pravidlo nám zaručuje, že v žádném kroku x_i a u_i (stejně jako y_i a v_i) nebudou nikdy současně bazickými proměnnými. Pokud se nám pomocí tohoto upraveného simplexového algoritmu podaří získat řešení úlohy (4.5.5) splňující $\sum_{i=1}^n z_i = 0$, potom poslední simplexová tabulka určuje řešení $[x^*, y^*, u^*, v^*]$ soustavy (4.5.3), a tedy odpovídající x^* je řešením původní úlohy (4.5.1). Toto je tzv. *krátký tvar Wolfeho metody*²⁷.

Konverguje vůbec tato metoda? Nemohlo by se stát, že nastane $\sum_{i=1}^n z_i > 0$ a kvůli Wolfeho pravidlu by nebylo možné pokračovat? Odpověď dává následující věta, která je „ukryta“ i ve Wolfeho článku [80].

Je-li $C > 0$ nebo $d = 0$, pak je úloha (4.5.5) řešitelná simplexovým algoritmem splněným o Wolfeho pravidlo, tj. po konečném počtu kroků dostaneme $\sum_{i=1}^n z_i = 0$.

VĚTA 4.5.1 WOLFEHO

²⁷ Výpočet v této fázi lze výrazně urychlit tím, že zavedeme pouze tolik pomocných proměnných z_i , kolik má záporných složek vektor d , tj. pokud $d_i \geq 0$ pro nějaké $i \in \{1, \dots, n\}$, pak místo zavedení z_i ponecháme v bázi u_i . V takovém případě bude výchozím BPB $A := [x, y, u, v, z] = [0, 0, \bar{u}, b, \bar{z}]$,

kde $\bar{u} = [\bar{u}_1, \dots, \bar{u}_n]$ a $\bar{z} = [\bar{z}_1, \dots, \bar{z}_n]$ jsou dána jako

$$\bar{u}_i = \begin{cases} 0 & d_i < 0, \quad i \in \{1, \dots, n\}, \\ d_i & d_i \geq 0, \quad i \in \{1, \dots, n\}, \end{cases}$$

$$\bar{z}_i = \begin{cases} |d_i| & d_i < 0, \quad i \in \{1, \dots, n\}, \\ 0 & d_i \geq 0, \quad i \in \{1, \dots, n\}. \end{cases}$$

Potom hodnota účelové funkce úlohy ve druhé fázi v bodě A je nejvýše rovna hodnotě této účelové funkce v „klasickém“ výchozím BPB popsaném výše. Proto stačí zavést pouze taková z_i , pro která je $\bar{z}_i \neq 0$. Kdybychom měli dokonce danou úlohu s $d \geq 0$, pak není potřeba zavádět žádné z_i a máme ihned řešení. Jaké? V takovém případě je bod $[0, 0, 0, b, d]$ řešením soustavy (4.5.3), takže $x^* = 0$ je řešením původní úlohy (4.5.1). To je zřejmé z toho, že $C \geq 0$, $d \geq 0$, $b \geq 0$ a $x \geq 0$ znamená, že nejmenší možná hodnota účelové funkce je 0, kterou dává právě bod $x^* = 0$.

Samozřejmě je možné, že tento algoritmus bude fungovat i v případě, kdy $C \geq 0$, $\det C = 0$ a $d \neq 0$, ovšem pro důkaz konvergence je potřebné $C > 0$ nebo $d = 0$. Alespoň zatím...

Příklad 4.5.2

Wolfeho metoda (krátký tvar) – řešený příklad #1: 📄 ☺ 🖨 <https://goo.gl/G2tAsa>

V případě $b \neq 0$ je situace komplikovanější a je nutné nejdříve provést *první fázi* Wolfeho metody, abychom našli výchozí BPB pro druhou fázi (to je vlastně analogie dvoufázové metody pro lineární programování). Karushovy–Kuhnovy–Tuckerovy podmínky lze schématicky zapsat jako

$$\begin{array}{c|cc|cc|c} & \overbrace{\begin{matrix} x & y \end{matrix}}^{n+m} & \overbrace{\begin{matrix} u & v \end{matrix}}^{n+m} & & \\ \text{„báze“} & & & & \\ \left\{ \begin{array}{l} u \\ v \end{array} \right. & \begin{array}{cc} -C & -A^\top \\ A & 0 \end{array} & \begin{array}{cc} I & 0 \\ 0 & I \end{array} & & \begin{array}{c} d \\ b \end{array} \end{array}$$

Tabulka 4.2: Schématická tabulka pro omezení (4.5.2a)–(4.5.2b).

Bod $[x, y, u, v] = [0, 0, d, b]$ je bazickým, neboť má $m + n$ bazických proměnných a odpovídající část matice určující omezení v (4.5.2a)–(4.5.2b) je $\begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix}$, ovšem pro $b \neq 0$ není přípustný (kdyby bylo, pak by $x^* = 0$ muselo být řešením). „Rozdělíme“ vektor b na²⁸ dvě části $\begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$, kde $b_1 \geq 0$ a

²⁸ Toto provedeme bez újmy na obecnosti – stačí pouze přehodit pořadí jednotlivých omezení v $Ax \leq b$.

$b < 0$. Analogicky rozdělíme také matici A , tj. máme

$$A_1 x \leq b_1, \quad b_1 \geq 0,$$

$$A_2 x \leq b_2, \quad b_2 < 0.$$

Stejným způsobem rozdělíme také vektory y, v a zavedeme umělé proměnné w_1, w_2, \dots , jejichž počet je dán počtem záporných složek vektoru b . Takto dostaneme novou výchozí tabulku

„báze“	x	y_1	y_2	u	v_1	v_2	w	
u	$-C$	$-A_1^\top$	$-A_2^\top$	I	0	0	0	d
v_1	A_1	0	0	0	I	0	0	$b_1 \geq 0$
w	$-A_2$	0	0	0	0	$-I$	I	$-b_2 > 0$

Tabulka 4.3: Výchozí tabulka pro první fázi Wolfeho metody.

BPB úlohy (4.5.1) pak dostaneme řešením pomocné úlohy

$$\begin{aligned} w_1 + \dots + w_r &\rightarrow \min \\ A_1 x + v_1 &= b_1 \\ -A_2 x - v_2 + w &= -b_2 \\ x \geq 0 \quad &\& \quad v_1 \geq 0 \quad &\& \quad v_2 \geq 0 \quad &\& \quad w \geq 0 \end{aligned} \quad (4.5.6)$$

s výchozím BPB $[x, v_1, v_2, w] = [0, b_1, 0, -b_2]$. Ve výchozí tabulce ponecháme i první „řádek“ reprezentující omezení (4.5.2a), ale při určování klíčového řádku se omezujeme jen a pouze na proměnné vystupující v pomocné úloze (4.5.6), tj. x, v_1, v_2, w . Jinými slovy, v první fázi žádná z proměnných y_1, \dots, y_m nevstoupí do báze a žádné u_1, \dots, u_n nevystoupí z báze. Nicméně úpravy popsané v simplexovém algoritmu provádíme s celou tabulkou²⁹. Cílem je vyloučit w z báze. Pokud se nám to nepodaří, znamená to, že původní úloha (4.5.1) je nepřípustná. Pokud se nám podaří vyloučit w z báze, pak pokračujeme druhou fází Wolfeho algoritmu tak, jak jsme ji popsali v předchozí části: zavedeme proměnné z_i podle prvního „bloku“ poslední simplexové tabulky v první fázi, tj. u_i nahradíme z_i a zavedeme matici D . Ovšem tentokrát již neplatí, že z_i zavádíme pouze pro zápornou pravou stranu – musíme totiž respektovat i Wolfeho pravidlo, tj. pokud je pravá strana kladná, ale odpovídající složka vektoru u nesmí být v bázi kvůli Wolfeho pravidlu (tj. v bázi už je x_i), musíme místo ní také zavést z_i .

Příklad 4.5.3

Wolfeho metoda (krátký tvar) – řešený příklad #2:    <https://goo.gl/WnaK1x>

²⁹ \triangle Pozor! Nelze se omezit pouze na druhé omezení a pracovat jenom s ním podobně jako v první fázi pro řešení úlohy lineárního programování. Potřebujeme najít hodnoty x, v tak, aby platilo druhé omezení a zároveň byli schopni najít výchozí BPB pro druhou fázi. Není totiž možné upravovat jenom druhé omezení a následně pokračovat druhou fází jako v předchozím případě, neboť ten byl založen na volbě výchozího BPB s $x = 0$, což asi nebude výsledkem první fáze. Nebo ano? Na začátku druhé fáze proto potřebujeme pracovat s novými omezeními, která získáme tím, že v první fázi upravujeme celou soustavu.

4.6 Wolfeho metoda v dlouhém tvaru

V případě, kdy nejsou splněny předpoklady Věty 4.5.1, tj. $\det C = 0$ a $d \neq 0$, pak může nastat situace, že při řešení úlohy výše popsaným způsobem již nebude možné kvůli Wolfeho pravidlu provést další krok simplexového algoritmu, ačkoli stále $\sum_{i=1}^n z_i > 0$. Z tohoto důvodu se takové

úlohy řeší pomocí tzv. *dlouhého tvaru Wolfeho metody*³⁰. Tato verze Wolfeho metody se skládá ze tří fází, přičemž princip prvních dvou je shodný s těmi v krátké variantě.

Základní myšlenka je v parametrizaci lineární části účelové funkce $f(x) = \frac{1}{2} \langle Cx, x \rangle + \langle d, x \rangle$ tak, že místo ní budeme pracovat s funkcí

$$\frac{1}{2} \langle Cx, x \rangle + \mu \langle d, x \rangle$$

pro $\mu \geq 0$. Pak díky Větě 4.5.1 můžeme najít řešení úlohy odpovídající $\mu = 0$ krátkým tvarem Wolfeho metody. Ovšem řešení původní úlohy odpovídá $\mu = 1$. Proto ve třetí fázi vyjdeme z řešení pro $\mu = 0$ a při stálém zachování Karushových–Kuhnových–Tuckerových podmínek začneme zvětšovat hodnotu μ , dokud nedosáhneme požadované hodnoty $\mu = 1$. Tímto způsobem lze dokonce řešit úlohu (4.5.1), kde účelová funkce $f(x) = \frac{1}{2} \langle Cx, x \rangle + v \langle d, x \rangle$ závisí na nějakém parametru v nabývajícím různých hodnot $0 \leq v_1 < v_2 < \dots < v_r$. My ale pro jednoduchost zůstaneme u případě³¹ $v = 1$.

Uvažme tedy úlohu

$$\frac{1}{2} \langle Cx, x \rangle + \mu \langle d, x \rangle \rightarrow \min, \quad Ax \leq b, \quad x \geq 0, \quad (4.6.1)$$

s odpovídajícími Karushovými–Kuhnovými–Tuckerovými podmínkami

$$-Cx - A^T y + \mu u = d \quad (4.6.2a)$$

$$Ax + \quad \quad v = b \quad (4.6.2b)$$

$$x \geq 0 \quad \& \quad y \geq 0 \quad \& \quad u \geq 0 \quad \& \quad v \geq 0 \quad \& \quad \mu \geq 0 \quad (4.6.2c)$$

$$\langle x, u \rangle = 0 \quad \& \quad \langle y, v \rangle = 0. \quad (4.6.2d)$$

Pokud $b \not\geq 0$, je nutné začít s první fází a pokračovat do druhé fáze, po jejímž skončení chceme mít vyliminovány všechny proměnné z_1, \dots, z_n z báze, tj. v případě, že již máme řešení splňující $\sum_{i=1}^n z_i = 0$, ale z důvodu degenerovanosti této pomocné úlohy zůstaly některé proměnné z_i v bázi, je nutné ve druhé fázi pokračovat tak dlouho, dokud všechna z_i nebudou vyloučena z báze. Až se nám to povede, můžeme zahájit třetí fázi s výchozím BPB určeným poslední simplexovou tabulkou ze druhé fáze. Kdyby $b \geq 0$, pak lze ihned přistoupit ke třetí fázi s výchozím BPB $[x, y, u, v, \mu] = [0, 0, 0, v, 0]$ a bázi tvořenou proměnnými u, v . Na začátku třetí fáze začneme zvětšovat hodnotu μ , přičemž musíme dbát na všechny podmínky (4.6.2), tj. ve třetí fázi řešíme úlohu

$$-\mu \rightarrow \min \quad \& \quad (4.6.2a) - (4.6.2c) \quad (4.6.3)$$

a současně dodržujeme Wolfeho pravidlo. Při tomto postupu mohou nastat 3 možnosti:

- (i) Nalezneme BPB úlohy (4.6.3) s kladným číslem v Δ -řádku (tj. není řešením), přičemž příslušný sloupec lze v souladu s Wolfeho pravidlem zavést do báze, ovšem tento sloupec s výjimkou Δ -řádku neobsahuje žádné další kladné číslo, tj. nelze provést další krok simplexového algoritmu. V takovém případě není účelová funkce úlohy (4.6.3) ohraničená zdola, tj. $\mu \rightarrow \infty$.
- (ii) Nalezneme řešení úlohy (4.6.3), tj. Δ -řádek neobsahuje kladné číslo.
- (iii) Nalezneme BPB úlohy (4.6.3) s kladným číslem v Δ -řádku (tj. není řešením), ale kvůli Wolfeho pravidlu nelze pokračovat dále.

VĚTA 4.6.1

- (a) Jestliže při řešení úlohy (4.6.3) simplexovým algoritmem doplněným o Wolfeho pravidlo nastane situace (ii) nebo (iii) popsaná výše, potom hodnota proměnné μ v poslední simplexové tabulce je nutně nulová.

³¹ Vše by fungovalo zcela analogicky s tím, že chceme získat $\mu = v_i$, tj. zvyšovali bychom hodnotu μ až k v_i .

VĚTA 4.6.1

(b) Jestliže při řešení úlohy (4.6.3) simplexovým algoritmem doplněným o Wolfeho pravidlo nastane situace (ii), potom hodnota účelové funkce úlohy (4.5.1) není zdola ohraničená, tj. tato úloha může mít řešení pouze v situaci (i) nebo (iii).

Z části (a) této věty vyplývá, že pokud při řešení úlohy (4.6.3) je v některém kroku $\mu > 0$, pak nutně nastane situace (i), tj. μ vstoupí do báze s nenulovou hodnotou a již ji nemůžeme anulovat. Pokud nastane (i) nebo (ii), pak buď můžeme určit řešení původní úlohy, nebo víme, že původní úloha je zdola neohraničená. Navíc v případě (ii) je hodnota účelové funkce v řešení zjevně nulová.

Lze také ukázat, že pro matici $C > 0$ nemůže situace (iii) nastat. Ovšem obecně nemůžeme v situaci (iii) rozhodnout, zda úloha (4.6.3) má řešení či nikoli (má-li řešení, tak ho kvůli $\mu = 0$ nemůžeme určit). Tedy v případě singulární matice C by se mohl dlouhý tvar Wolfeho metody zaseknout. Vskutku? Naštěstí tomu lze zabránit tzv. *Shettyho pravidlem*³², které vylučuje situaci (iii). Ve výchozí tabulce pro třetí fázi je z každé dvojice x_i, u_i a y_i, v_i právě jedna proměnná v bázi a druhá mimo bázi³³. Proto zavedeme v prvním kroku této fáze proměnnou μ (jinou možnost ani nemáme). Označíme-li si dvojice x_i, u_i a y_i, v_i pomocí

$$\tilde{x}_i := \begin{cases} x_i, & i = 1, \dots, n, \\ v_{i-n}, & i = n+1, \dots, 2n, \end{cases} \quad \tilde{u}_i := \begin{cases} u_i, & i = 1, \dots, n, \\ y_{i-n}, & i = n+1, \dots, 2n, \end{cases}$$

pak na začátku třetí fáze je v bázi právě jedna proměnná z dvojice \tilde{x}_i, \tilde{u}_i a v prvním kroku některá z nich bázi. Ovšem kvůli Wolfeho pravidlu existují pouze dvě proměnné, které mohou v dalším kroku vstoupit do báze: buď bude vrácena zpět proměnná vystoupivši z báze nebo tam vstoupí druhá proměnná z dvojice \tilde{x}_i, \tilde{u}_i , což platí i pro všechny další kroky třetí fáze. Tyto dvě proměnné tvoří tzv. *nebazickou dvojici*. Označíme-li odpovídající hodnotu v Δ -řádku nové simplexové tabulky jako Δ a $\hat{\Delta}$, pak pro proměnnou vystupující z báze je nutně $\Delta \leq 0$, takže čísla Δ a $\hat{\Delta}$ nemohou být kladná zároveň, tj. nemá smysl tam ihned vracet zpět proměnnou vyloučenou z báze. V každém kroku třetí fáze tak lze jednoznačně určit proměnnou, která má vstoupit do báze

Shettyho pravidlo: opustilo-li bázi \tilde{x}_i (nebo \tilde{u}_i), pak v dalším kroku do báze musí vstoupit \tilde{u}_i (nebo \tilde{x}_i) bez ohledu na to, zda odpovídající hodnota v Δ -řádku je kladná či nulová.

Protože lze také ukázat, že čísla Δ a $\hat{\Delta}$ nemohou být ani současně záporné, je jasné, že tento proces nutně skončí buď situací (i) nebo (ii).

Nyní se podíváme na situaci (i), kdy $\mu \rightarrow \infty$, tj. když účelová funkce úlohy (4.6.3) není zdola ohraničená, a tedy úloha (4.5.1) má řešení. V takovém případě dostaneme při řešení (4.6.3) upraveným simplexovým algoritmem konečnou posloupnost BPB soustavy (4.6.2a)–(4.6.2c), tj.

$$X^{[k]} := [x^{[k]}, y^{[k]}, u^{[k]}, v^{[k]}, \mu^{[k]}], \quad k = 0, 1, \dots, r$$

a nakonec polopřímku

$$X^{[r]} + \tau \bar{X}, \quad \tau \geq 0, \quad (4.6.4)$$

na které nabývá μ libovolně velké hodnoty, tj. $\mu \rightarrow \infty$. Směr $\bar{X} = (\bar{x}, \bar{y}, \bar{u}, \bar{v}, \bar{\mu})^\top$ této polopřímky je určen tím sloupcem poslední simplexové tabulky, který má v Δ -řádku kladné číslo, tzv. *kritický sloupec*. Takový sloupec je nutně jediný v důsledku Shettyho pravidla a ostatní prvky tohoto

³² Není to pravidlo v tom klasickém smyslu – simplexový algoritmus vám totiž nic moc jiného nedovolí, než podle něj postupovat.

³³ Vyloučili jsme všechna z_i z báze a kvůli Wolfeho pravidlu tam tyto proměnné nemohou být současně, jenže v bázi musí být právě $n+m$ prvků, takže tak z každé dvojice musí být právě jeden.

sloupce jsou nekladné, protože jsme v situaci (i). Tento směr proto dostaneme tak, že proměnné odpovídající tomuto sloupci (tzv. *kritická proměnná*, kterou bychom v posledním kroku chtěli zavést do báze) přiřadíme hodnotu 1, ostatním nebazickým proměnným kritického sloupce přiřadíme hodnotu 0 a bazickým proměnným přiřadíme hodnoty odpovídající číslům z kritického sloupce vynásobeného -1 , tj. \bar{X} obsahuje pouze nezáporná čísla.

Wolfeho pravidlo nám zaručuje nejen nulovost skalárních součinů $\langle x, u \rangle$ a $\langle y, v \rangle$ v každém kroku, ale také platí

$$\begin{aligned}\langle x^{[k]}, u^{[k+1]} \rangle &= 0 = \langle x^{[k+1]}, u^{[k]} \rangle, & k = 0, 1, \dots, r-1, \\ \langle y^{[k]}, v^{[k+1]} \rangle &= 0 = \langle y^{[k+1]}, v^{[k]} \rangle, & k = 0, 1, \dots, r-1, \\ \langle x^{[r]}, \bar{u} \rangle &= 0 = \langle \bar{x}, u^{[r]} \rangle, \\ \langle y^{[r]}, \bar{v} \rangle &= 0 = \langle \bar{y}, v^{[r]} \rangle,\end{aligned}$$

tedy podmínky ortogonálnosti jsou splněny i pro libovolnou konvexní kombinaci dvou po sobě jdoucích BPB a všechny body polopřímky (4.6.4), o čemž se lze snadno přesvědčit vzhledem k linearitě skalárního součinu, tj. např.

$$\begin{aligned}\langle \alpha x^{[k]} + \beta x^{[k+1]}, \alpha u^{[k]} + \beta u^{[k+1]} \rangle &= \\ &= \alpha^2 \langle x^{[k]}, u^{[k]} \rangle + \beta^2 \langle x^{[k+1]}, u^{[k+1]} \rangle + \alpha\beta(\langle x^{[k]}, u^{[k+1]} \rangle + \langle x^{[k+1]}, u^{[k]} \rangle) = 0,\end{aligned}$$

kde $\alpha + \beta = 1$, $\alpha \geq 0$, $\beta \geq 0$ a $k = 0, 1, \dots, r-1$. Stejný výpočet můžeme zopakovat i pro dvojici y, v a konečně

$$\langle x^{[r]} + \tau \bar{x}, u^{[r]} + \tau \bar{u} \rangle = \langle x^{[r]}, u^{[r]} \rangle + \tau^2 \langle \bar{x}, \bar{u} \rangle + \tau(\langle x^{[r]}, \bar{u} \rangle + \langle \bar{x}, u^{[r]} \rangle) = 0.$$

Pro nalezení řešení úlohy (4.5.1) pak musíme rozlišit 3 možné případy:

- (i) Je-li $\mu^{[k]} = 1$ pro nějaké $0 \leq k \leq r$, pak odpovídající $X^{[k]}$ je řešením soustavy (4.6.2) s $\mu = 1$, a tedy složka $x[k]$ je řešením úlohy (4.5.1), tj. $x^* = x^{[k]}$.
- (ii) Je-li $\mu^{[k]} < 1 < \mu^{[k+1]}$ pro nějaké $0 \leq k \leq r-1$, pak konvexní kombinace

$$X^* := \lambda^* X^{[k]} + (1 - \lambda^*) X^{[k+1]}, \quad \text{kde } 0 < \lambda^* := \frac{\mu^{[k+1]} - 1}{\mu^{[k+1]} - \mu^{[k]}} < 1,$$

tvoří řešení soustavy (4.6.2) s $\mu = 1$, takže $x^* = \lambda^* x^{[k]} + (1 - \lambda^*) x^{[k+1]}$ řeší úlohu (4.5.1).

- (iii) Je-li $\mu^{[r]} < 1$, pak

$$X^* := X^{[r]} + \tau^* \bar{X}, \quad \text{kde } \tau^* := \frac{1 - \mu^{[r]}}{\bar{\mu}},$$

je řešením soustavy (4.6.2) s $\mu = 1$, a tedy $x^* = x^{[r]} + \tau^* \bar{x}$ je řešením úlohy (4.5.1).

Příklad 4.6.2

Vyřešme úlohu

$$\begin{aligned}x_1^2 + x_2^2 + 2x_1x_2 + x_1 - 2x_2 &\rightarrow \min \\ x_1 + x_2 &\leq 5, \\ 2x_1 + x_2 &\geq 4, \\ x_1 \geq 0 \quad \& \quad x_2 \geq 0.\end{aligned}$$

Řešení. Přepíšeme-li druhé omezení do tvaru $-2x_1 - x_2 \leq -4$, pak máme

$$C = \begin{pmatrix} 2 & 2 \\ 2 & 2 \end{pmatrix}, \quad d = \begin{pmatrix} 1 \\ -2 \end{pmatrix}, \quad A = \begin{pmatrix} 1 & 1 \\ -2 & -1 \end{pmatrix}, \quad b = \begin{pmatrix} 5 \\ -4 \end{pmatrix}.$$

Jelikož $b \not\geq 0$ budeme potřebovat první fázi Wolfeho metody. Současně máme $C \geq 0$ a $\det C = 0$, takže je potřeba použít dlouhý tvar Wolfeho metody.

Fáze 1

Máme pouze $b_2 < 0$, takže v této fázi přidáme pouze proměnnou w_2 , tj. $w = (0, w_2)^T$, a řešíme úlohu

$$\begin{aligned} w_2 &\rightarrow \min \\ x_1 + x_2 + v_1 &= 5 \\ 2x_1 + x_2 - v_2 + w_2 &= 4 \\ x_1 \geq 0 \quad &\& \quad x_2 \geq 0 \quad &\& \quad v_1 \geq 0 \quad &\& \quad v_2 \geq 0 \quad &\& \quad w_2 \geq 0. \end{aligned}$$

c_B	B	x_1	x_2	\dot{y}_1	\dot{y}_2	u_1	u_2	v_1	v_2	$\dot{\mu}$	w_2	x_B	x_B/λ
0	u_1	-2	-2	-1	2	1	0	0	0	-1	0	0	\times
0	u_2	-2	-2	-1	1	0	1	0	0	2	0	0	\times
0	v_1	1	1	0	0	0	0	1	0	0	0	5	5
1	w_2	②	1	0	0	0	0	0	-1	0	1	4	$2 \rightarrow$
	c_k	0	0	0	0	0	0	0	0	0	1		
	z_k	2	1	0	0	0	0	0	-1	0	1		
	Δ	$2 \uparrow$	1	0	0	0	0	0	-1	0	0	4	
				\bullet	\bullet					\bullet			
	u_1	0	-1	-1	2	1	0	0	-1	-1	\times	4	
	u_2	0	-1	-1	1	0	1	0	-1	2	\times	4	
	v_1	0	1/2	0	0	0	0	1	1/2	0	\times	3	
	x_1	1	1/2	0	0	0	0	0	-1/2	0	\times	2	
	Δ	0	0	0	0	0	0	0	0	0	0	0	

KONEC první fáze. Nalezli jsme řešení s $w_2 = 0$, takže úloha je přípustná a můžeme sestavit výchozí BPB pro druhou fázi.

Fáze 2

Nyní zavedeme proměnné z_1, z_2 a zvolíme $D = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$. Pak chceme najít řešení úlohy

s omezeními z poslední simplexové tabulky z první fáze, tj.

$$\begin{aligned}
 & z_1 + z_2 \rightarrow \min, \\
 & -x_2 - y_1 + 2y_2 + u_1 - v_2 - \mu + z_1 = 4, \\
 & -x_2 - y_1 + y_2 + u_2 - v_2 + 2\mu + z_2 = 4, \\
 & x_2/2 + v_1 + v_2/2 = 3, \\
 & x_1 + x_2/2 - v_2/2 = 2, \\
 & x_1 \geq 0 \ \& \ x_2 \geq 0 \ \& \ y_1 \geq 0 \ \& \ y_2 \geq 0 \ \& \ u_1 \geq 0 \\
 & u_2 \geq 0 \ \& \ v_1 \geq 0 \ \& \ v_2 \geq 0 \ \& \ \mu \geq 0 \ \& \ z_1 \geq 0. \ \& \ z_2 \geq 0.
 \end{aligned}
 \tag{4.6.5}$$

Za výchozí BPB pak vezmeme $x_1 = 2, x_2 = y_1 = y_2 = u_1 = u_2 = 0, v_1 = 3, v_2 = 0, \mu = 0, z_1 = 4, z_2 = 4$, tj. v bázi je x_1, v_1, z_1, z_2 . V takovém případě máme výchozí simplexovou tabulku

c_B	B	x_1	x_2	y_1	y_2	u_1	u_2	v_1	v_2	μ	z_1	z_2	x_B	x_B/λ
1	z_1	0	-1	-1	2	1	0	0	-1	-1	1	0	4	
1	z_2	0	-1	-1	1	0	1	0	-1	2	0	1	4	
0	v_1	0	1/2	0	0	0	0	1	1/2	0	0	0	3	
0	x_1	1	1/2	0	0	0	0	0	-1/2	0	0	0	2	
	c_k	0	0	0	0	0	0	0	0	0	1	1		
	z_k	0	-2	-2	3	1	1	0	-2	1	1	1		
	Δ	0	-2	-2	3	1	1	0	-2	1	0	0	8	

Cílem této fáze je vyeliminovat proměnné z_1 a z_2 z báze, což bude vyžadovat nejméně 2 kroky. Ovšem v souladu s úvodním poznámkou to můžeme udělat s pomocí menšího počtu proměnných. V poslední simplexové tabulce v první fázi máme na pravé straně (ve sloupci označeném x_B) kladná čísla, takže za výchozí BPB pro úlohu ve druhé fázi bychom mohli vzít $x_1 = 2, x_2 = y_1 = y_2 = 0, u_1 = 4, u_2 = 4, v_1 = 3, v_2 = 0, \mu = 0, z_1 = 0, z_2 = 0$, tj. nepotřebujeme z_1, z_2 . Mohli bychom tedy tuto fázi přeskočit a pokračovat rovnou třetí fázi. Jenže tento BPB je v rozporu s Wolfeho pravidlem – jelikož x_1 nám vyšlo na konci první fáze v bázi, nesmíme u_1 zavést do báze. Proto místo něj zavedeme proměnnou z_1 . Současně u_2 v bázi nechat můžeme, takže nepotřebujeme z_2 . Proto stačí ve druhé fázi řešit úlohu

$$\begin{aligned}
 & z_1 \rightarrow \min, \\
 & -x_2 - y_1 + 2y_2 + u_1 - v_2 - \mu + z_1 = 4, \\
 & -x_2 - y_1 + y_2 + u_2 - v_2 + 2\mu = 4, \\
 & x_2/2 + v_1 + v_2/2 = 3, \\
 & x_1 + x_2/2 - v_2/2 = 2, \\
 & x_1 \geq 0 \ \& \ x_2 \geq 0 \ \& \ y_1 \geq 0 \ \& \ y_2 \geq 0 \ \& \ u_1 \geq 0 \\
 & u_2 \geq 0 \ \& \ v_1 \geq 0 \ \& \ v_2 \geq 0 \ \& \ \mu \geq 0 \ \& \ z_1 \geq 0,
 \end{aligned}$$

s výchozím BPB ve tvaru $x_1 = 2, x_2 = y_1 = y_2 = u_1 = 0, u_2 = 4, v_1 = 3, v_2 = 0, \mu = 0, z_1 = 4$, tj. v bázi je x_1, u_1, v_1, z_2 .

c_B	B	x_1	\dot{x}_2	\dot{y}_1	y_2	\dot{u}_1	u_2	v_1	v_2	$\dot{\mu}$	z_1	x_B	x_B/λ
1	z_1	0	-1	-1	②	1	0	0	-1	-1	1	4	2 →
0	u_2	0	-1	-1	1	0	1	0	-1	2	0	4	4
0	v_1	0	1/2	0	0	0	0	1	1/2	0	0	3	×
0	x_1	1	1/2	0	0	0	0	0	-1/2	0	0	2	×
	c_k	0	0	0	0	0	0	0	0	0	1		
	z_k	0	-1	-1	2	1	0	0	-1	-1	1		
	Δ	0	-1	-1	2 ↑	1	0	0	-1	-1	0	4	
			•	•		•		•	•				
	y_2	0	-1/2	-1/2	1	1/2	0	0	-1/2	-1/2	×	2	
	u_2	0	-1/2	-1/2	0	-1/2	1	0	-1/2	5/2	×	2	
	v_1	0	1/2	0	0	0	0	1	1/2	0	×	3	
	x_1	1	1/2	0	0	0	0	0	-1/2	0	×	2	
	Δ	0	0	0	0	0	0	0	0	0	×	0	

KONEC druhé fáze. Úplně stejnou finální tabulku bychom dostali i při řešení úlohy (4.6.5), ale potřebovali bychom k tomu 2 kroky. Nyní tuto finální tabulku využijeme pro sestavení úlohy ve třetí fázi.

Fáze 3

Ve třetí fázi řešíme úlohu

$$\begin{aligned}
 & -\mu \rightarrow \min, \\
 & -x_2/2 - y_1/2 + y_2 + u_1/2 - v_2/2 - \mu/2 = 2, \\
 & -x_2/2 - y_1/2 - u_1/2 + u_2 - v_2/2 + 5\mu/2 = 2, \\
 & x_2/2 + v_1 + v_2/2 = 3, \\
 & x_1 + x_2/2 + -v_2/2 = 2, \\
 & x_1 \geq 0 \quad \& \quad x_2 \geq 0 \quad \& \quad y_1 \geq 0 \quad \& \quad y_2 \geq 0 \\
 & u_1 \geq 0 \quad \& \quad u_2 \geq 0 \quad \& \quad v_1 \geq 0 \quad \& \quad v_2 \geq 0 \quad \& \quad \mu \geq 0.
 \end{aligned}$$

Třetí fázi zahájíme zavedením proměnné μ do báze (jinou možnost ani nemáme).

$$\mu^{[0]} = 0$$

Shetty:

 u_2 \updownarrow

B

 \updownarrow x_2

$$\mu^{[1]} = 4/5$$

$$\mu^{[2]} = 8/5$$

c_B	B	x_1	\dot{x}_2	\dot{y}_1	y_2	\dot{u}_1	u_2	v_1	\dot{v}_2	μ	x_B	x_B/λ
0	y_2	0	-1/2	-1/2	1	1/2	0	0	-1/2	-1/2	2	×
0	u_2	0	-1/2	-1/2	0	-1/2	1	0	-1/2	5/2	2	4/5 →
0	v_1	0	1/2	0	0	0	0	1	1/2	0	3	×
0	x_1	1	1/2	0	0	0	0	0	-1/2	0	2	×
	c_k	0	0	0	0	0	0	0	0	-1		
	z_k	0	0	0	0	0	0	0	0	0		
	Δ	0	0	0	0	0	0	0	0	1 ↑	0	
				•		•			•			
	y_2	0	-3/5	-3/5	1	2/5	1/5	0	-3/5	0	12/5	×
	μ	0	-1/5	-1/5	0	-1/5	2/5	0	-1/5	1	4/5	×
	v_1	0	1/2	0	0	0	0	1	1/2	0	3	6
	x_1	1	1/2	0	0	0	0	0	-1/2	0	2	4 →
	Δ	0	1/5 ↑	1/5	0	1/5	-2/5	0	1/5	0	-4/5	
				•		•			•			
	y_2	6/5	0	-3/5	1	2/5	1/5	0	-6/5	0	24/5	
	μ	2/5	0	-1/5	0	-1/5	2/5	0	-2/5	1	8/5	
	v_1	-1	0	0	0	0	0	1	1	0	1	
	x_2	2	1	0	0	0	0	0	-1	0	4	
	Δ	-2/5	0	1/5	0	1/5	-2/5	0	2/5	0	-8/5	

KONEC třetí fáze, neboť $\mu^{[1]} < 1 < \mu^{[2]}$. Jelikož platí

$$x^{[1]} = [2, 0], \quad x^{[2]} = [0, 4] \quad \text{a} \quad \lambda^* = \frac{\mu^{[2]} - 1}{\mu^{[2]} - \mu^{[1]}} = \frac{8/5 - 1}{8/5 - 4/5} = \frac{3}{4},$$

je hledaným optimálním řešením dané úlohy

$$x^* = \lambda^* x^{[1]} + (1 - \lambda^*) x^{[2]} = [3/2, 1].$$

Příklad 4.6.3

Vyřešme úlohu

$$x_1^2 + x_2^2 + 2x_1x_2 + 20x_1 - 40x_2 \rightarrow \min$$

$$x_1 + x_2 \leq 5,$$

$$2x_1 + x_2 \geq 4,$$

$$x_1 \geq 0 \quad \& \quad x_2 \geq 0.$$

Řešení. Daná úloha se od předchozího příkladu liší pouze tím, že tentokrát máme $d = (20, -40)^\top$, což je 20-násobek vektoru d v předchozím příkladu. Proto první a druhá fáze budou zcela totožné s tím, že sloupec pro μ musíme vynásobit 20. Můžeme tedy ihned pokračovat třetí fází.

Fáze 3

Ve třetí fázi řešíme úlohu

$$-\mu \rightarrow \min,$$

$$-x_2/2 - y_1/2 + y_2 + u_1/2 - v_2/2 - 10\mu = 2,$$

$$-x_2/2 - y_1/2 - u_1/2 + u_2 - v_2/2 + 50\mu = 2,$$

$$x_2/2 + v_1 + v_2/2 = 3,$$

$$x_1 + x_2/2 + -v_2/2 = 2,$$

$$x_1 \geq 0 \quad \& \quad x_2 \geq 0 \quad \& \quad y_1 \geq 0 \quad \& \quad y_2 \geq 0$$

$$u_1 \geq 0 \quad \& \quad u_2 \geq 0 \quad \& \quad v_1 \geq 0 \quad \& \quad v_2 \geq 0 \quad \& \quad \mu \geq 0.$$

Třetí fázi opět zahájíme zavedením μ do báze (jinou možnost ani nemáme).

c_B	B	x_1	\dot{x}_2	\dot{y}_1	y_2	\dot{u}_1	u_2	v_1	\dot{v}_2	μ	x_B	x_B/λ	
$\mu^{[0]} = 0$	0	y_2	0	-1/2	-1/2	1	1/2	0	0	-1/2	-10	2	×
	0	u_2	0	-1/2	-1/2	0	-1/2	1	0	-1/2	50	2	2/50 →
	0	v_1	0	1/2	0	0	0	0	1	1/2	0	3	×
	0	x_1	1	1/2	0	0	0	0	0	-1/2	0	2	×
Shetty:		c_k	0	0	0	0	0	0	0	0	-1		
		z_k	0	0	0	0	0	0	0	0	0		
		Δ	0	0	0	0	0	0	0	0	1 ↑	0	
					•		•			•			
		y_2	0	-3/5	-3/5	1	2/5	1/5	0	-3/5	0	12/5	×
		μ	0	-1/100	-1/100	0	-1/100	1/50	0	-1/100	1	2/50	×
$\mu^{[1]} = 2/50$		v_1	0	1/2	0	0	0	0	1	1/2	0	3	6
Shetty:		x_1	1	1/2	0	0	0	0	0	-1/2	0	2	4 →
		Δ	0	1/100 ↑	1/100	0	1/100	-1/50	0	1/100	0	-2/50	
					•		•			•			
		y_2	6/5	0	-3/5	1	2/5	1/5	0	-6/5	0	24/5	12 →
		μ	1/50	0	-1/100	0	-1/100	1/50	0	-1/50	1	4/50	×
$\mu^{[2]} = 4/50$		v_1	-1	0	0	0	0	0	1	1	0	1	×
Shetty:		x_2	2	1	0	0	0	0	0	-1	0	4	×
		Δ	-1/50	0	1/100	0	1/100 ↑	-1/50	0	1/50	0	-4/50	
					•		•			•			
		u_1	3	0	-3/2	5/2	1	1/2	0	-3	0	12	×
		μ	1/20	0	-1/40	1/40	0	1/40	0	-1/20	1	1/5	×
$\mu^{[3]} = 1/5$		v_1	-1	0	0	0	0	0	1	1	0	1	1 →
Shetty:		x_2	2	1	0	0	0	0	0	-1	0	4	×
		Δ	-1/20	0	1/40	-1/40	0	-41/200	0	1/20 ↑	0	-1/5	
					•		•			•			
		u_1	0	0	-3/2	5/2	1	1/2	3	0	0	15	×
		μ	0	0	-1/40	1/40	0	1/40	1/20	0	1	1/4	×
$\mu^{[4]} = 1/4$		v_2	-1	0	0	0	0	0	1	1	0	1	×
		x_2	1	1	0	0	0	0	1	0	0	5	×
		Δ	0	0	1/40 ↑	-1/40	0	-41/200	-1/20	0	0	-1/4	

KONEC třetí fáze. Podle Shettyho pravidla má nyní vstoupit do báze proměnná y_1 . To je v souladu s Wolfeho pravidlem, ovšem sloupec pro y_1 neobsahuje kromě Δ -řádku žádné kladné číslo, takže není možné určit, kterou bychom z báze vyloučili. Současně máme $x^{[4]} = [0, 5]$, $\mu^{[4]} = 1/4$ a z poslední tabulky určíme

$$\bar{X} = (0, 0, 1, 0, 3/2, 0, 0, 0, 1/40)^T,$$

tj. $\bar{x} = [0, 0]$ a $\bar{\mu} = 1/40$. Pak $\tau^* = \frac{1-\mu^{[4]}}{\bar{\mu}} = 30$, a tedy hledaným optimálním řešením dané

úlohy je

$$x^* = x^{[4]} + \tau^* \bar{x} = [0, 5].$$

KAPITOLA 5

Dynamické programování

¹ Optimální strategie má tu vlastnost, že ať je počáteční stav a počáteční rozhodnutí jakékoli, zbývající rozhodnutí musí tvořit optimální strategii vzhledem ke stavu vyplývajícího z prvního rozhodnutí.

² Mým prvním úkolem v dynamickém programování bylo vybudování rigorózních základů. Zjistil jsem, že pořád dokola využívám stále tu stejnou metodu k odvození funkcionální rovnice. Rozhodl jsem se tuto metodu nazvat „Principem optimality“. Oliver Gross mi jednoho dne řekl: „Ten princip není rigorózní.“ Odpověděl jsem mu: „Samozřejmě, že ne. Není dokonce ani přesný.“ Správný princip by měl nasměrovat intuici.



Richard Ernest Bellman (26. srpna 1920 – 19. března 1984) byl americký matematik, v roce 1953 položil základy dynamického programování. V roce 1975 získal *John von Neumann Theory Prize* za „jeho více než dvacet let trvající vůdčí roli v oblasti, kterou sám vytvořil: dynamické programování“.

5.1 Motivace	130
5.2 Konečněkrokový deterministický proces	132
5.3 Stochastický proces	138
5.4 Nekonečněkrokový deterministický proces	139

AN OPTIMAL POLICY HAS THE PROPERTY THAT WHATEVER THE INITIAL STATE AND THE INITIAL DECISION ARE, THE REMAINING DECISIONS MUST CONSTITUTE AN OPTIMAL POLICY WITH REGARD TO THE STATE RESULTING FROM THE FIRST DECISION.¹

MY FIRST TASK IN DYNAMIC PROGRAMMING WAS TO PUT IT ON A RIGOROUS BASIS. I FOUND THAT I WAS USING THE SAME TECHNIQUE OVER AND OVER TO DERIVE A FUNCTIONAL EQUATION. I DECIDE TO CALL THIS TECHNIQUE, “THE PRINCIPLE OF OPTIMALITY.” OLIVER GROSS SAID ONE DAY, “THE PRINCIPLE IS NOT RIGOROUS.” I REPLIED, “OF COURSE NOT, IT IS NOT EVEN PRECISE.” A GOOD PRINCIPLE SHOULD GUIDE THE INTUITION.²

RICHARD BELLMAN³

Viz „bibli“ každého dynamického programovače [10, str. 83] a také Bellmanovu autobiografii [11, str. 174].

5.1 Motivace

V úvodní části Kapitoly 2 jsme hovořili o tom, že Dantzig postrádal v původním Leontiefově modelu *dynamiku* umožňující změny v průběhu času. O dynamice jsme se zmínili i v souvislosti s von Neumannovou prací z roku 1932. Hledání optimální strategie pro procesy, které se vyvíjejí v čase, jsme si také ukázali v několika příkladech, viz ⁴. Ovšem ne vždy je řešení takových problémů pomocí lineárního programování možné či efektivní. My si v této kapitole ukážeme jiný možný přístup k takovýmto úlohám, a to s využitím tzv. *dynamické programování*, jehož základy položil Bellman na počátku 50. let a v průběhu následujících desetiletí jej (společně s mnoha dalšími) úspěšně rozvíjel. Hlavním nástrojem této oblasti je tzv. *Bellmanův princip optimality*.

⁴ ODKAZY

Příklad 5.1.1

Uvažme Sjednocené pískové závody Brno, které prodávají písek do celého světa (tj. mezi Brnem, Rajhradem a Kuřimí). Tento písek je možné těžít buď v Černovicích nebo v Hrušovanech u Brna, přičemž kvalita písku je zcela totožná. Každá koruna investovaná do pískovny v Černovicích na začátku roku přinese na konci roku a v každém dalším roce 2 tuny písku a 0,5 Kč zisku. Podobně každá koruna investovaná v Hrušovanech přinese 1 tunu písku a 0,9 Kč. Naším úkolem je určení rozdělení finančních prostředků, které jsou vždy na počátku roku k dispozici, mezi oba závody tak, aby celková produkce v období následujících pěti let byla maximální. Náš počáteční kapitál činí y Kč.

Řešení. Označme symbolem $x_n^{\check{c}}$ celkové investiční prostředky vynaložené v Černovicích do konce n -tého roku a podobně investice v Hrušovanech označíme jako x_n^{h} . Dále označme volné investiční prostředky na konci $(n-1)$ -ho (tj. na začátku n -tého) roku jako v_n a poměrnou část volných prostředků reinvestovaných v Černovicích na začátku n -tého roku jako $0 \leq r_n \leq 1$. Pak pro investice platí

$$x_n^{\check{c}} = x_{n-1}^{\check{c}} + r_n v_n \quad \text{a} \quad x_n^{\text{h}} = x_{n-1}^{\text{h}} + (1 - r_n) v_n,$$

$$v_n = 0,5 x_{n-1}^{\check{c}} + 0,9 x_{n-1}^{\text{h}},$$

pro $n = 1, 2, \dots, 5$, přičemž bereme $x_0^{\check{c}} = 0 = x_0^{\text{h}}$ a $v_1 = y$. Velikost produkce na konci n -tého roku je $2x_n^{\check{c}} + x_n^{\text{h}}$ tun písku. Cílem proto je stanovení r_1, \dots, r_n tak, že

$$\sum_{n=1}^5 (2x_n^{\check{c}} + x_n^{\text{h}}) \rightarrow \max \quad \text{neboli} \quad \sum_{n=1}^5 [2(x_{n-1}^{\check{c}} + r_n v_n) + x_{n-1}^{\text{h}} + (1 - r_n) v_n] \rightarrow \max.$$

To asi nebude úloha lineárního, celočíselného ani kvadratického programování. . .

Příklad 5.1.2

Uvažme násobení matic A_1, \dots, A_n vhodných rozměrů. Velmi dobře víme, že tato operace není komutativní, ale pouze asociativní. Ovšem pro výpočet tohoto součinu je velmi důležité pořadí jednotlivých součinů a naším cílem je minimalizace celkového počtu potřebných součinů, přičemž neodlišujeme násobení nulou, jedničkou nebo jakýmkoli jiným číslem. Např. pro trojici matic

$$A_1 \in \mathbb{R}^{5 \times 4} \quad \& \quad A_2 \in \mathbb{R}^{4 \times 6} \quad \& \quad A_3 \in \mathbb{R}^{6 \times 2}$$

dostaneme

při uzávorkování $(A_1 \cdot A_2) \cdot A_3$ celkem $(5 \cdot 6 \cdot 4) + (5 \cdot 2 \cdot 6) = 180$ potřebných součinů, zatímco pro $A_1 \cdot (A_2 \cdot A_3)$ to je jenom $(5 \cdot 2 \cdot 4) + (4 \cdot 2 \cdot 6) = 88$ potřebných součinů.

Navíc, vzhledem k rovnosti mezi počty řádků a sloupců sousedních matic, není nutné si pamatovat kompletní rozměry všech matic. Stačí pracovat pouze s posloupností p_1, p_2, \dots, p_{n+1} určující počty řádků matic A_1, \dots, A_n a počet sloupců matice A_n . Jak ale pomocí této posloupnosti najít optimální pořadí jednotlivých součinů?

Tyto dva motivační příklady jsou ukázkou tzv. *víceetapových/víceetapových rozhodovacích procesů*, tj. procesů, při nichž dochází k rozhodnutí v každém kroku. My pak hledáme jednotlivá dílčí rozhodnutí, která nám společně vytvoří *optimální strategii*, která přinese maximální užitek/zisk apod. Tyto procesy obvykle zahrnují změny v čase – dynamičnost, a proto oblast věnovaná těmto úlohám se nazývá *dynamické programování*.

Základním jednotkou každého procesu je *etapa* či *krok* (v předchozích příkladech to byl 1 rok a 1 násobení). Každý proces se pak rozdělí do několika dílčích etap, tj.



Obrázek 5.1: Etapy rozhodovacího procesu.

Dynamické programy můžeme klasifikovat do několika tříd.

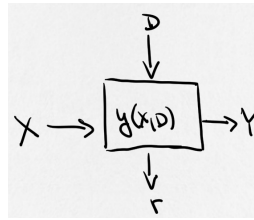
- (i) *Diskrétní*: rozhodovací proměnná může nabývat pouze diskrétních (tj. spočetně mnoha) hodnot.
Spojité: rozhodovací proměnná může nabývat libovolných hodnot z nějakého intervalu (tj. nespočetně mnoha).
- (ii) *Konečněkrokový* (tj. konečný horizont) vs. *nekonečněkrokový* (tj. nekonečný horizont).
- (iii) *Deterministický*: budoucnost je určitá, tj. výsledek každého rozhodnutí je a priori znám.
Stochastický: budoucnost je neurčitá, tj. existují i jisté vnější vlivy, které mohou ovlivnit výsledek učiněného rozhodnutí (známe alespoň míru pravděpodobnosti jednotlivých možných výsledků každého rozhodnutí).
- (iv) *Jednoparametrický proces* vs. *víceparametrický proces* (např. rozdělení dvou na sobě nezávislých zdrojů velikosti X a Y do n ekonomických aktivit).

⁵ Princip *divide et impera* (rozděl a panuj).

Hlavní myšlenkou řešení těchto problémů je tzv. *dekompozice*⁵, kdy se původní komplexní problém rozdělí na řadu (z jistého pohledu) jednodušších úloh. Výsledek pak získáme kombinací/složením řešení těchto jednodušších úloh, přičemž

- (i) *rozhodovací situace* nastává v případě, kdy existuje více než jedno přípustné řešení;
- (ii) *cílem* rozhodování (nebo optimalizačního problému) je určení jednoho rozhodnutí (řešení, politiky, strategie), které dává optimální výsledek (tj. žádná jiná strategie nedává lepší výsledek).

Nyní se podíváme blíže na jednu etapu rozhodovacího procesu:



Obrázek 5.2: Podrobný vhléd do jedné etapy rozhodovacího procesu.

- (i) *vstupní veličina (vektor) X* představuje popis výchozího (počátečního) stavu systému a obsahuje všechny relevantní informace;
- (ii) *výstupní veličina (vektor) Y* popisuje systém v konečné úrovni a obsahuje všechny informace o výstupu;
- (iii) *rozhodovací proměnná D* (též *vektor řízení*) charakterizuje operace probíhající v průběhu jednotlivých etap;
- (iv) *účelová funkce r* je jednorozměrná funkce vstupu, rozhodování a výstupu, tj.

$$r = r(X, D, Y);$$

- (v) *transformační funkce y* vyjadřuje každou komponentu výstupní proměnné jako funkci vstupu a rozhodnutí, tj.

$$Y = y(X, D),$$

a ve stochastickém programování bude záviset i na náhodné veličině, tj.

$$Y = y(X, D, Q).$$

⁶ ⚠ Pozor na značení: funkce r vyžaduje 3 proměnné, ovšem zápis $r(X, D)$ znamená, že k tomu ve skutečnosti postačuje znát pouze 2 proměnné X a D , tj. vstupní a rozhodovací proměnná.

S využitím transformační funkce můžeme vyeliminovat Y z účelové funkce, čímž dostaneme⁶

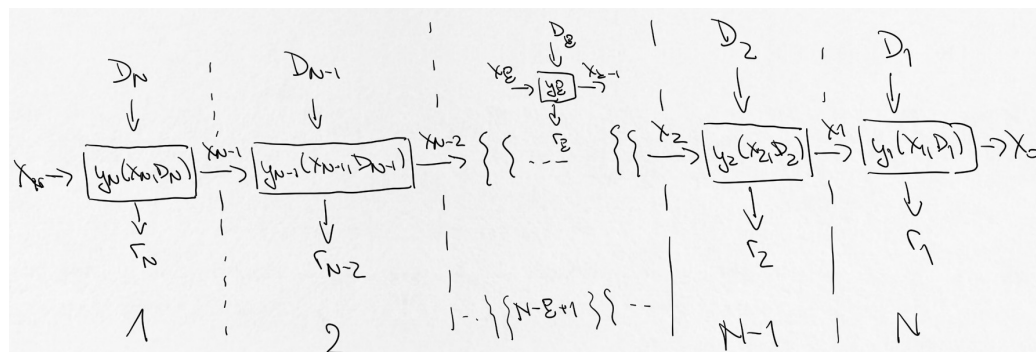
$$r = r(X, D, y(X, D)) = r(X, D).$$

Jednokrokový optimalizační problém pak spočívá v nalezení max / min účelové funkce jakožto funkce vstupní veličiny. Označíme-li tuto optimální (maximální) hodnotu jako f^* a odpovídající optimální rozhodnutí jako $D^* = D(X)$, pak platí

$$f^* = r(X, D^*) = f(X, D(X)) = \max_D r(X, D) \geq r(X, D).$$

5.2 Konečněkrokový deterministický proces

V další části se blíže podíváme na N -krokový rozhodovací proces, jehož schéma lze znázornit následovně



Obrázek 5.3: Podrobný rozbor všech etap víceetapového procesu.

⁷ V tuto chvíli to možná vypadá jako další zbytečná komplikace, ovšem význam této změny oceňujeme za chvíli. Snad...

Jednotlivé etapy jsou ve schématu úmyslně indexovány od N po 0 , tj. X_N je výchozí stav a X_0 je cílový stav⁷. Jelikož výstupní veličina k -tého kroku je totožná se vstupní veličinou $(k-1)$ -ho kroku, platí

$$x_{k-1} = y_k(X_k, D_k) \quad \text{pro } k = N, N-1, \dots, 1 \quad (5.2.1)$$

a výnos v k -tém kroku je

$$r_k = r_k(X_k, D_k).$$

⁸ Stále mějte na paměti poznámku ohledně značení a počtu proměnných.

Pak ale z (5.2.1) vyplývá, že X_k závisí na rozhodnutích, která předcházejí k -tému kroku, tj. na $D_N, D_{N-1}, \dots, D_{k+1}$ a X_N neboli⁸

$$\begin{aligned} X_k &= y_{k+1}(X_{k+1}, D_{k+1}) = y_{k+1}(y_{k+2}(X_{k+2}, D_{k+2}), D_{k+1}) = y_{k+1}(X_{k+2}, D_{k+2}, D_{k+1}) = \\ &= y_{k+1}(y_{k+3}(X_{k+3}, D_{k+3}), D_{k+2}, D_{k+1}) = y_{k+1}(X_{k+3}, D_{k+3}, D_{k+2}, D_{k+1}) = \\ &= \dots = y_{k+1}(X_N, D_N, \dots, D_{k+1}). \end{aligned}$$

Dosazením do účelové funkce pak získáme

$$r_k = r_k(X_k, D_k) = r_k(y_{k+1}(X_N, D_N, \dots, D_{k+1}), D_k) = r_k(X_N, D_N, \dots, D_k), \quad (5.2.2)$$

tj. výnos s k -tým krokem je ovlivněn pouze rozhodnutími v N -tém, $(N-1)$ -ním, ..., k -tém kroku. Nemají na něj tedy (pochopitelně) žádný vliv rozhodnutí z budoucnosti. Cílem dynamického programování pak může být nalezení strategie takové, že např.

$$\begin{aligned} \sum_{k=1}^N g(r_k(X_k, D_k)) \rightarrow \text{opt} \quad \text{nebo} \quad g\left(\sum_{k=1}^N r_k(X_k, D_k)\right) \rightarrow \text{opt}, \\ g(X_0) \rightarrow \text{opt} \quad \text{nebo} \quad r_k(X_k, D_k) \rightarrow \text{opt} \quad (\text{pro } k = N, \dots, 1). \end{aligned}$$

⁹ Toto závisí na interpretaci funkce r .

Nechť proto celkový „výnos“⁹ R_N celého procesu je nějakou funkcí jednotlivých hodnot účelových funkcí, tj.

$$R_N = R_N(X_N, \dots, X_1, D_N, \dots, D_1) = g[r_N(X_N, D_N), r_{N-1}(X_{N-1}, D_{N-1}), \dots, r_1(X_1, D_1)]. \quad (5.2.3)$$

Nyní ovšem můžeme z (5.2.3) vyeliminovat X_{N-1}, \dots, X_1 pomocí (5.2.2), čímž získáme

$$R_N = R_N(X_N, D_N, \dots, D_1) = g[r_N(X_N, D_N), r_{N-1}(X_N, D_N, D_{N-1}), \dots, r_1(X_N, D_N, \dots, D_1)]$$

neboli optimalizační problém, ve kterém chceme optimalizovat celkový výnos pomocí dílčích rozhodnutí D_N, \dots, D_1 při počátečním stavu X_N . Označíme-li dále jako $f_N(X_N)$ optimální řešení

N -krokového procesu s výchozí hodnotou X_N a $D_k^* = D_k(X_k)$, $X_{k-1}^* = y_k(X_k, D_k^*)$ optimální rozhodnutí a odpovídající výstupní hodnotu, potom

$$\begin{aligned} f_N(X_N) &= g[r_N(X_N, D_N^*), r_{N-1}(X_{N-1}^*, D_{N-1}^*), \dots, r_1(X_1^*, D_1^*)] = \\ &= \text{opt}_{D_N, \dots, D_1} g[r_N(X_N, D_N), r_{N-1}(X_{N-1}, D_{N-1}), \dots, r_1(X_1, D_1)] \end{aligned}$$

za podmínky

$$X_{k-1} = y_k(X_k, D_k) \quad \text{pro } k = N, \dots, 1,$$

a tudíž také

$$f_N(X_N) = \text{opt}_{D_N, \dots, D_1} g[r_N(X_N, D_N), r_{N-1}(X_N, D_N, D_{N-1}), \dots, r_1(X_N, D_N, \dots, D_1)].$$

Hlavním nástrojem řešení úloh dynamického programování je *Bellmanův princip optimality* zmíněný na začátku této kapitoly. Zde je ještě několik dalších alternativních formulací:

Je-li ke každému subproblému nalezeno optimální řešení, pak existuje i optimální řešení původního problému (a to je z nich složeno).

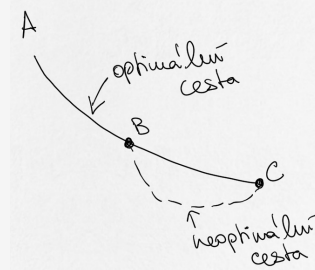
Každá podstrategie optimální strategie je opět optimální.

Je-li posloupnost $\{D_N, \dots, D_k\}$ optimální strategií $(N - k)$ -krokového rozhodovacího procesu s počátečním stavem X_N , kde $D_i = D_i(X_i)$ závisí pouze na aktuálním stavu pro $i = N, \dots, k$, pak posloupnost $\{D_N, \dots, D_{k+1}\}$ tvoří optimální strategii $(N - k - 1)$ -krokového rozhodovacího procesu.

Celou „filozofii“ Bellmanova principu je možné označit za používání *zdravého rozumu* a ilustrovat na velmi jednoduchém příkladu.

Příklad 5.2.1

Mějme za úkol najít optimální cestu z bodu A do C .



Obrázek 5.4: Optimální cesta a Bellmanův princip optimality.

Připusťme, že řešení vede přes bod B . Pak ale cesta z B do C , která je součástí nalezeného řešení, také musí být optimální. Kdyby totiž existovala výhodnější cesta z B do C , pak spojení této cesty s cestou z A do B dávalo lepší řešení, než původní nalezené řešení. A to je spor.

° OBRAZEK

Připusťme nyní, že celková účelová funkce je aditivní, tj.

$$g[r_N(X_N, D_N), r_{N-1}(X_{N-1}, D_{N-1}), \dots, r_1(X_1, D_1)] = r_N(X_N, D_N) + r_{N-1}(X_{N-1}, D_{N-1}) + \dots + r_1(X_1, D_1),$$

a pro určitost se dohodněme, že budeme hledat maximum, tj.

$$f_N(X_N) = \max_{D_N, \dots, D_1} [r_N(X_N, D_N) + r_{N-1}(X_{N-1}, D_{N-1}) + \dots + r_1(X_1, D_1)]$$

za podmínky $X_{k-1} = y_k(X_k, D_k)$ pro $k = N, \dots, 1$. Hodnota $r_N(X_N, D_N)$ nezávisí na D_{N-1}, \dots, D_1 , proto díky rovnosti

$$\max_{u_1, u_2} [h_1(u_1) + h_2(u_2)] = \max_{u_1} [h_1(u_1) + \max_{u_2} h_2(u_1, u_2)] \quad (5.2.4)$$

plyne z (5.2), že

$$f_N(X_N) = \max_{D_N} \{r_N(X_N, D_N) + \max_{D_{N-1}, \dots, D_1} [r_{N-1}(X_{N-1}, D_{N-1}) + \dots + r_1(X_1, D_1)]\}$$

za podmínky $X_{k-1} = y_k(X_k, D_k)$ pro $k = N, \dots, 1$. Pak z definice $f_N(X_N)$ a Bellmanova principu vyplývá

$$f_N(X_N) = \max_{D_N} [r_N(X_N, D_N) + f_{N-1}(X_{N-1})]$$

za podmínky $X_{k-1} = y_k(X_k, D_k)$ pro $k = N, \dots, 1$, tj.

$$f_N(X_N) = \max_{D_N} [r_N(X_N, D_N) + f_{N-1}(y_N(X_N, D_N))].$$

Položme nyní

$$Q_N(X_N, D_N) := r_N(X_N, D_N) + f_{N-1}(y_N(X_N, D_N)).$$

Pak určení $f_N(X_N)$ a odpovídajícího $D_N^* = D_N(X_N)$ je snadné, pokud již známe $f_{N-1}(X_{N-1})$, tj. máme jednoduchý optimalizační problém se vstupní proměnnou X_N , rozhodnutím D_N a výnosem Q_N neboli

$$f_N(X_N) = \max_{D_N} Q_N(X_N, D_N).$$

Podařilo se nám tak rozdělit N -krokový problém na dva menší subproblémy:

(i) $(N-1)$ -krokový optimalizační problém

$$f_{N-1}(X_{N-1}) = \max_{D_{N-1}, \dots, D_1} [r_{N-1}(X_{N-1}, D_{N-1}) + \dots + r_1(X_1, D_1)]$$

za podmínky $X_{k-1} = y_k(X_k, D_k)$ pro $k = N-1, \dots, 1$;

(ii) jednorozměrný problém

$$f_N(X_N) = \max_{D_N} Q_N(X_N, D_N) = \max_{D_N} [r_N(X_N, D_N) + f_{N-1}(y_N(X_N, D_N))].$$

Analogicky můžeme pokračovat i dále a $f_{N-1}(X_{N-1})$, $f_{N-2}(X_{N-2})$, ... $f_2(X_2)$ rozdělit na subproblémy, čímž získáme *rekurentní schéma*

$$f_k(X_k) = \max_{D_k} Q_k(X_k, D_k), \quad \text{pro } k = N, \dots, 1, \text{ kde}$$

$$Q_k = \begin{cases} r_k(X_k, D_k), & k = 1, \\ r_k(X_k, D_k) + f_{k-1}(y_k(X_k, D_k)), & k = 2, \dots, N. \end{cases}$$

Pak pomocí tohoto schématu nalezneme optimální řešení původní problému tak, že jej začneme řešit pro $k = 1$ a pokračujeme¹⁰ až po $k = N$. Tím nalezneme výnos $f_N(X_N)$ v N -tém kroku, odpovídající optimální rozhodnutí $D_N^* = D_N(X_N)$ a *rozhodovací funkci* $D_k = D_k(X_k)$ pro $k = N-1, \dots, 1$. Následně získáme optimální rozhodnutí a aktuální stavy jakožto funkce X_N , tj.

$$X_{N-1}^* = y_N(X_N, D_N^*) = y_N(X_N, D_N(X_N)) = y_N(X_N)$$

¹⁰ Proto to indexování v opačném pořadí.

a

$$D_{N-1}^* = D_{N-1}(X_{N-1}^*) = D_N(y_N(X_N, D_N^*)) = D_{N-1}(X_N)$$

a rekurzivně dále

$$X_{k-1}^* = y_k(X_k, D_k^*) = y_k(X_k), \quad D_{k-1}^* = D_{k-1}(X_{k-1}^*) = D_{k-1}(X_N), \quad k = N-1, \dots, 1.$$

Speciálním případem takového rozhodovacího procesu je *optimalizace cílového stavu*. Jedná se o úlohu, jejímž cílem je

$$f_N(X_N) = \text{opt}_{D_N, \dots, D_1} g(X_0)$$

za podmínky $X_{k-1} = y_k(X_k, D_k)$ pro $k = N, \dots, 1$. V takovém případě můžeme využít předchozí schéma, kde položíme

$$r_N(X_N, D_N) = r_{N-1}(X_{N-1}, D_{N-1}) = \dots = r_2(X_2, D_2) = 0 \quad \text{a} \quad r_1(X_1, D_1) = g[y_1(X_1, D_1)] = g(X_0).$$

Potom platí

$$g(X_0) = r_N(X_N, D_N) + \dots + r_1(X_1, D_1)$$

a vlastně tedy chceme vyřešit problém

$$f_N(X_N) = \text{opt}_{D_N, \dots, D_1} [r_N(X_N, D_N) + \dots + r_1(X_1, D_1)] = \text{opt}_{D_N, \dots, D_1} \sum_{k=1}^N r_k(X_k, D_k)$$

za podmínky $X_{k-1} = y_k(X_k, D_k)$ pro $k = N, \dots, 1$.

Podobným postupem lze řešit také úlohy, kdy funkce g není aditivní. Ovšem abychom mohli aplikovat Bellmanův princip, nemůže být funkce g zcela libovolná. Musí být tzv. *separovatelná*, tj.

$$g[r_N(X_N, D_N), \dots, r_1(X_1, D_1)] = g_1\{r_N(X_N, D_N), g_2[r_{N-1}(X_{N-1}, D_{N-1}), \dots, r_1(X_1, D_1)]\}$$

a musí zároveň splňovat podmínku *monotonie*:

$$g_1 \text{ je neklesající funkcí proměnné } g_2 \text{ pro libovolnou hodnotu } r_N,$$

viz (5.2.4). Potom lze optimalizační problém opět rozložit a získat rekurzivní schéma

$$f_k(X_k) = \max_{D_k} Q_k(X_k, D_k), \quad \text{pro } k = N, \dots, 1, \text{ kde}$$

$$Q_k = \begin{cases} r_k(X_k, D_k), & k = 1, \\ r_k(X_k, D_k) \circ f_{k-1}(y_k(X_k, D_k)), & k = 2, \dots, N, \end{cases}$$

kde \circ je operátor „skládání“ (dříve tam bylo znaménko +), protože díky separovatelnosti můžeme $R_N = g(r_N, \dots, r_1)$ zapsat jako $R_N = g_1[X_N, g_2(r_{N-1}, \dots, r_1)]$ neboli $R_N = r_N \circ R_{N-1}$. V takovém případě máme

$$f_N(X_N) = \max_{D_N} \{r_N(X_N, D_N) \circ \max_{D_{N-1}, \dots, D_1} [r_{N-1}(X_{N-1}, D_{N-1}) \circ \dots \circ r_1(X_1, D_1)]\} = \text{opt}_{D_N, \dots, D_1}$$

za podmínky $X_{k-1} = y_k(X_k, D_k)$ pro $k = N, \dots, 1$ neboli

$$f_N(X_N) = \max_{D_N} [r_N(X_N, D_N) \circ f_{N-1}(X_{N-1})].$$

Např. místo $\sum_{k=1}^N r_k$ můžeme mít $\prod_{k=1}^N r_k$ nebo $\ln(r_N + \dots + r_1)$ atd.

Příklad 5.2.2

Rozhodněme, která z funkcí

(a) $r_N \cdot r_{N-1} \cdots r_1$

(b) $r_3 \cdot r_2 + r_1$

(c) $r_3 + r_2 \cdot r_1$

(d) $r_4 + r_3 \cdot r_2 + r_1$

splňuje podmínku separovatelnosti a monotonie.

Nyní si ilustrujeme využití dynamického programování na několika příkladech. První z těchto úloh je *nalezení optimální cesty v grafu*.

Příklad 5.2.3

Problém dostavníku:   https://odkaz.page.link/video_DP1

  https://odkaz.page.link/zapisnik_DP1

Předchozí příklad můžeme interpretovat také jako *problém kritické cesty*. To je úloha, ve které jednotlivé uzly představují určité činnosti a jsou uspořádány tak, jak musí následovat za sebou (např. při stavbě domu je nutné před instalací oken¹¹ nejdříve postavit zeď), tj. spojení uzlů pak vyjadřuje časovou souslednost jednotlivých činností. Cílem je nalezení *kritických činností* (tj. těch, jejichž zdržení bude mít za následek zdržení celého projektu)¹² a maximálního nutného času pro dokončení projektu a případně i identifikovat činnosti, které mohou probíhat současně.

¹¹ Nikoli „windows“ – to by úlohu učinilo neřešitelnou. . .

¹² V případě *nekritických činností* máme jistou časovou rezervu pro jejich dokončení.

Ve většině úloh řešených pomocí dynamického programování lze nalézt řešení i jinými metodami (viz předchozí příklad). Ovšem využití dynamického programování přináší mnohdy značné zjednodušení (zejména v úlohách variačního počtu). Uvažme například N -krokový rozhodovací proces, kde v každém kroku lze učinit k rozhodnutí. Celkem máme k^N možných variant, jenže při použití dynamického programování máme pouze $k \cdot N$ variant. V situaci ilustrované na obrázku není rozdíl příliš markantní $2^4 = 16$ vs. $2 \cdot 4 = 8$. Jenže např. pro $k = 2$ a $N = 100$ máme $2^{100} \approx 1,267 \cdot 10^{30}$ vs. $2 \cdot 100 = 200$. Na druhou stranu Bellmanův princip není exaktní matematické tvrzení, ačkoli je intuitivně přijatelné. Také neexistuje žádná univerzální metoda řešení (na rozdíl např. od lineárního programování a simplexové metody). Každá úloha (nebo typ úloh) má svůj vlastní přístup k řešení.

Samozřejmě ne v každé úloze, kterou lze řešit pomocí dynamického programování, musí být dynamika celého procesu způsobena přítomností nějakého časového údaje. Někdy je možné závislost na čase do daného problému implementovat uměle, tak jako to uděláme v následujícím řešení *problému batohu*, s nímž jsme se už několikrát setkali v předchozích kapitolách. To je typický a zároveň nejjednodušší příklad tohoto typu.

Příklad 5.2.4

Problém batohu:   https://odkaz.page.link/video_DP2

  https://odkaz.page.link/zapisnik_DP2

Už víme, že problém batohu je velmi speciální případ *alokačního problému*, takže se nyní podíváme, jak lze řešit i obecnější úlohy pomocí dynamického programování.

Příklad 5.2.5

Alokační problém: 📄 🎥 📄 https://odkaz.page.link/video_DP3
📄 🎥 📄 https://odkaz.page.link/zapisnik_DP3

Všechny dosavadní řešené problémy byly ukázkou diskretních rozhodovacích procesů, ve kterých jsme mohli uvážit „všechny“ dostupné možnosti pomocí jednoduchých výčtových tabulek. Toto ale již není možné v případě *spojitého* procesu a my si v následujícím příkladu ukážeme řešení jednoho z nich.

Příklad 5.2.6

Spojité rozhodovací proces: 📄 🎥 📄 https://odkaz.page.link/video_DP4
📄 🎥 📄 https://odkaz.page.link/zapisnik_DP4

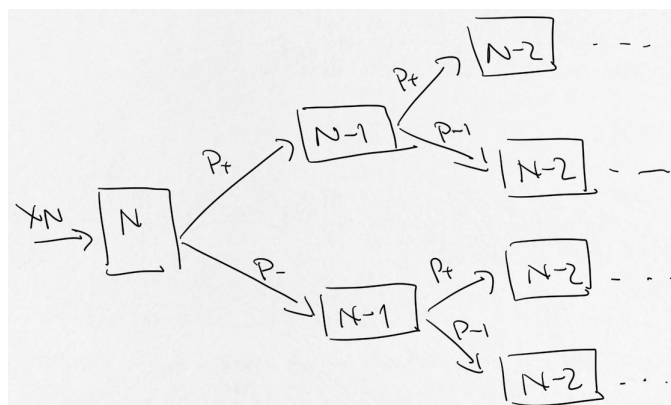
5.3 Stochastický proces

Ještě jedna malá ukáзка *stochastického* rozhodovacího procesu, ve kterém už samozřejmě není možné nalézt jednoznačnou optimální strategii jako v předchozí části. Přesuneme se proto do kasina v Las Vegas. Na počátku si koupíme V žetonů. Budeme hrát celkem N -krát jednoduchou hru, ve které je pravděpodobnost výhry p_+ a prohry přirozeně $p_- = 1 - p_+$. Při výhře se výhra zdvojnásobí, zatímco pro při prohře naši sázku ztratíme. Naším cílem je nalezení takové optimální strategie (tj. té s největší pravděpodobností), že na konci budeme mít alespoň Z žetonů. Pro $k = N, \dots, 1$ si označíme jako

X_k ... počet žetonů na začátku k -té hry (tj. ještě jsme nevsadili);

D_k ... počet žetonů vsazených v k -té hře.

Jedno kolo hry budeme brát jako jeden krok v dynamickém procesu, jehož začátek je schématicky zakreslen na obrázku níže.



Obrázek 5.5: Začátek stochastického rozhodovacího procesu.

Potom $f_k(X_k, D_k)$ vyjadřuje pravděpodobnost, že po k hrách (tolik ji zbývá dokonce při našem zpětném číslování) budeme mít alespoň Z žetonů, pokud na začátku k -té hry máme X_k žetonů a do této hry jich vsadíme D_k , přičemž až „doposud“ (tj. v krocích $N, \dots, k+1$) jsme dělali optimální rozhodnutí. My chceme v tomto kroku najít

$$f_k(X_k) = \max_{D_k \in \{0, \dots, X_k\}} f_k(X_k, D_k).$$

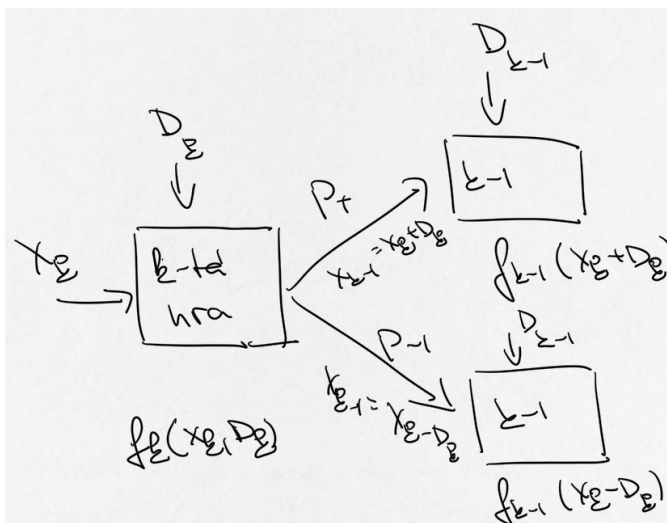
Vyjádření $f_k(X_k, D_k)$ musí zohledňovat skutečnost, že je možné mít alespoň Z žetonů i v případě prohry v k -té hře. Pokud v k -té hře prohráme, pak budeme mít $X_{k-1} = X_k - D_k$ a pravděpodobnost úspěchu je $f_{k-1}(X_k - D_k)$. Podobně v případě výhry bude $X_{k-1} = X_k + D_k$ s pravděpodobností úspěchu $f_{k-1}(X_k + D_k)$. Proto platí

$$f_k(X_k, D_k) = p_+ f_{k-1}(X_k + D_k) + p_- f_{k-1}(X_k - D_k),$$

přičemž bereme

$$f_0(x_0) := \begin{cases} 0, & \text{je-li } X_0 < Z, \\ 1, & \text{je-li } X_0 \geq Z, \end{cases}$$

viz obrázek níže.




Obrázek 5.6: Jeden krok stochastického rozhodovacího procesu.

Dostáváme tak rekurzivní schéma

$$f_k(X_k) = \max_{D_k \in \{0, \dots, X_k\}} p_+ f_{k-1}(X_k + D_k) + p_- f_{k-1}(X_k - D_k)$$

pro $k = N, \dots, 1$ a s $f_0(X_0)$ definovaným výše.

Příklad 5.3.1

Stochastický rozhodovací proces:  https://odkaz.page.link/video_DP5

 https://odkaz.page.link/zapisnik_DP5

KAPITOLA 6

Variační počet

6.1 <i>Není to jen o nejrychlejší klouzačce</i>	142
6.2 <i>Variuji, variuješ, variujeme</i>	142
6.3 <i>Základní úloha variačního počtu</i>	147
6.4 <i>Úloha o brachistochroně</i>	153

¹ *Vaše řešení isoperimetrického problému neponechává nic, co by se mělo ještě objasnit, a já jsem nesmírně rád, že šťastný los připadl na Vás, abyste řešení problému dotáhl na nejvyšší stupeň dokonalosti a vytvořil teorii, kterou jsem od samého počátku rozvíjel jen já. Důležitost tohoto tématu mě podnítila za pomoci Vašich k vybudování analytického řešení, které budu držet v tajnosti tak dlouho, dokud Vaše rozvahy nebudou publikovány, abych Vám neubral část slávy, kterou si zasloužíte.*

(Your solution of the isoperimetric problem contains, insofar as I can see, all that may be desired in this area, and I am extremely happy that this theory, which I hardly touched upon after my first attempts, has been brought to such very great perfection by you. The importance of that subject has stimulated me to develop, aided by your lights, an analytical solution which I will keep secret as long as your own meditations are not published, lest I take away from you a part of the glory which you deserve.)

² *Mně se zdá, že Berlín, dokud tam je pan Euler, by pro mě nebyl vůbec vhodný.*

³ *Kdybych byly bohatý, pravděpodobně bych se nevěnoval matematice.*

Tua solutio problematis isoperimetrici continet, ut video, quidquid in hac questione desiderari potest; et ego maxime gaudeo hoc argumentum, quod fere solus post primos conatus tractaveram, a te potissimum ad summum perfectionis fastigium esse evectum. Rei dignitas me excitavit ut tuis luminibus adjunctus, ipse solutionem analyticam conscripserim quam tamen celare statui, donec ipse tuas meditationes publici juris feceris, ne ullam partem gloriae tibi debitae praeripiam.¹

Leonhard Euler

It seems to me that Berlin would not be at all suitable for me while M Euler is there.²

If I had been rich, I probably would not have devoted myself to mathematics.³

Eulerův výrok pochází z dopisu Lagrangeovi datovaném 2. října 1759, viz soupis jejich vzájemné korespondence <https://odkaz.page.link/EulerLagrange>. Další dva výroky pochází z Lagrangeovy biografie <https://odkaz.page.link/Lagrange1>, přičemž ten první je úryvkem z Lagrangeova druhého odmítnutí pozice na Královské Pruské akademii věd v Berlíně z roku 1766 (poprvé mu byla tato pozice nabídnuta již v roce 1756 – kdepak asi v té době byl pan Euler?).

6.1 Není to jen o nejrychlejší klouzačce

6.2 Variuji, variuješ, variujeme

Problémy variačního počtu zmíněné v úvodní kapitole patří mezi tzv. *základní úlohy*, ve kterých má funkcionál $\mathcal{F}[\cdot]$ velmi specifický tvar. Než se ale pustíme do jejich řešení, musíme si vybudovat alespoň některé základní teoretické nástroje, které vlastně budou analogií tvrzení známých z diferenciálního počtu více proměnných. Připomeňme, že funkci $f: \mathbb{R}^n \rightarrow \mathbb{R}$ definovanou v nějakém okolí bodu x_0 nazveme v tomto bodě *diferencovatelnou*, jestliže pro funkci $\omega: \mathbb{R}^n \rightarrow \mathbb{R}$ danou vztahem

$$\omega(h) := f(x_0 + h) - f(x_0) - (a_1 h_1 + \dots + a_n x_n) = f(x_0 + h) - f(x_0) - \langle a, h \rangle$$

existuje n -tice $a := (a_1, \dots, a_n)^\top \in \mathbb{R}^n$ taková, že

$$\lim_{h \rightarrow 0} \frac{\omega(h)}{\|h\|} = 0. \quad (6.2.1)$$

Takový vektor a pak je nutně určen jednoznačně. Nazýváme jej *gradientem* funkce f v bodě x_0 a pro jeho složky platí $a_1 = f_{x_1}(x_0)$, \dots , $a_n = f_{x_n}(x_0)$, zatímco odpovídající skalární součin jakožto funkce $\langle a, h \rangle: \mathbb{R}^n \rightarrow \mathbb{R}$ se nazývá (*totálním, silným* nebo *Fréchetovým*) *diferenciálem*. To znamená, že v okolí bodu x_0 platí

$$\Delta f(x_0) := f(x_0 + h) - f(x_0) = \langle a, h \rangle + \omega(h), \quad (6.2.2)$$

tj. máme *přírůstek závisle proměnné* f na levé straně vyjádřený pomocí lineárního části $\langle a, h \rangle$ a nelineární části $\omega(h)$, která je v jistém smyslu „malá“.

Naším cílem nyní bude zobecnit vyjádření (6.2.2) pro funkcionály a s jeho pomocí odvodit některá tvrzení důležitá pro řešení optimalizačních úloh. Ovšem naše výsledky budou muset být formulovány trochu jinak, než jsme to dělali v základních kurzech matematické analýzy. V nich jsme totiž podmínky pro lokální/globální extrémů budovali pomocí (parciálních) derivací nikoli pomocí diferenciálu, který jsme si zavedli „tak trochu bokem“. Budete-li mít ale stále na paměti spojení mezi (totálním) diferenciálem a (parciálními) derivacemi, tak analogie snad bude více méně zřejmá (alespoň na začátku). Také bude potřeba zohlednit to, že už obecně nebude možné perturbovat nezávislou proměnnou libovolným „vektorem“, jak jsme to dělali v případě $x_0 + h$, tj. budeme se moci pohybovat pouze v jisté přípustné množině.

Mějme nyní dán funkcionál $\mathcal{F}[\cdot]: \mathcal{V} \rightarrow \mathbb{R}$ na normovaném vektorovém prostoru \mathcal{V} s normou $\|\cdot\|$. Naším úkolem je najít minimum/maximum tohoto funkcionálu na *přípustné* množině $\mathcal{N} \subseteq \mathcal{V}$. Taková množina ovšem samozřejmě nemusí být vektorovým podprostorem, jak lze snadno vidět na příkladu $\mathcal{N} = \{y \in \mathcal{V} \mid y(a) = \alpha, y(b) = \beta\}$ s volbou $\alpha \neq 0$ nebo $\beta \neq 0$. Pak totiž pro $y_1, y_2 \in \mathcal{N}$ a $c_1, c_2 \in \mathbb{R}$ máme $c_1 y_1(a) + c_2 y_2(a) = (c_1 + c_2) \alpha$ a $c_1 y_1(b) + c_2 y_2(b) = (c_1 + c_2) \beta$, z čehož lze snadno vyvodit, že vyjma několika velmi speciálních voleb platí $c_1 y_1 + c_2 y_2 \notin \mathcal{N}$. Společně s množinou \mathcal{N} budeme potřebovat ještě také tzv. *přípustné variace* nezávisle proměnné, jejichž přičtením k $y \in \mathcal{N}$ se neporuší přípustnost, tj. funkce $\eta \in \mathcal{V}$ takové, že $y + \eta \in \mathcal{N}$ pro libovolné $y \in \mathcal{N}$. Tuto množinu označíme jako \mathcal{N}_0 . Např. pro výše zmíněnou množinu \mathcal{N} budou přípustné variace tvořit vektorový podprostor $\mathcal{N}_0 = \{\eta \in \mathcal{V} \mid \eta(a) = 0 = \eta(b)\}$.

Před následující definicí ještě bude užitečné připomenout, že (*spojitým*) *lineárním funkcionálem* na normovaném vektorovém prostoru \mathcal{N}_0 rozumíme zobrazení $\varphi : \mathcal{N}_0 \rightarrow \mathbb{R}$ splňující

$$\varphi(c_1 \eta_1 + c_2 \eta_2) = c_1 \varphi(\eta_1) + c_2 \varphi(\eta_2) \quad \text{pro libovolná } c_1, c_2 \in \mathbb{R} \text{ a } \eta_1, \eta_2 \in \mathcal{N}_0$$

a spojitě pro všechna $\eta \in \mathcal{N}_0$, tj. pro každé $\eta \in \mathcal{N}_0$ a libovolné $\varepsilon > 0$ existuje $\delta > 0$ takové, že $|\varphi(\eta) - \varphi(\tilde{\eta})| < \varepsilon$ pro všechna $\tilde{\eta} \in \mathcal{N}_0$ splňující $\|\eta - \tilde{\eta}\| < \delta$. Takovými funkcionály jsou např.

- (i) $\varphi(\eta) := \eta(x_0)$ pro nějaké pevně dané $x_0 \in [a, b]$,
- (ii) $\varphi(\eta) := \int_a^b \eta(x) dx$ pro $\eta \in C[a, b]$,
- (iii) $\varphi(\eta) := \int_a^b \alpha(x) \eta(x) dx$ pro $\eta \in C[a, b]$ a pevně zvolenou funkci $\alpha \in C[a, b]$,
- (iv) $\varphi(\eta) := \int_a^b [\alpha_0(x) \eta(x) + \alpha_1(x) \eta'(x) + \dots + \alpha_n(x) \eta^{(n)}(x)] dx$ pro $\eta \in C^n[a, b]$ a pevně zvolené funkce $\alpha_0, \dots, \alpha_n \in C[a, b]$.

DEFINICE 6.2.1

Nechť $y \in \mathcal{N}$ je dáno. Přírůstek funkcionálu $\mathcal{F}[y]$ odpovídající přírůstku $\eta \in \mathcal{N}_0$ v „nezávislé proměnné“ y označíme jako

$$\Delta \mathcal{F}_y[\eta] = \Delta \mathcal{F}[y; \eta] := \mathcal{F}[y + \eta] - \mathcal{F}[y].$$

Funkcionál $\mathcal{F}[\cdot]$ nazveme *diferencovatelným* v y , jestliže

$$\Delta \mathcal{F}_y[\eta] = \varphi(\eta) + \omega(\eta) \|\eta\|, \tag{6.2.3}$$

pro všechny přípustné variace $\eta \in \mathcal{N}_0$, nějaký lineární funkcionál $\varphi(\eta)$ a nelineární funkcionál $\omega(\cdot)$ splňující $\omega(\eta) \rightarrow 0$ pro $\|\eta\| \rightarrow 0$. Funkcionál φ se nazývá *variací* (nebo *diferenciál*) *funkcionálu* $\mathcal{F}[\cdot]$ v y a značí se $\mathcal{F}'[y; \eta]$, příp. jen $\mathcal{F}'[y]$ nebo $\delta \mathcal{F}[y]$.

VĚTA 6.2.2

Variace $\mathcal{F}'[y; \eta]$ diferencovatelného funkcionálu $\mathcal{F}[\cdot]$ je určena jednoznačně.

Důkaz. Nejdříve ukážeme, že jediným lineárním funkcionálem φ splňujícím $\varphi(\eta)/\|\eta\| \rightarrow 0$ pro $\|\eta\| \rightarrow 0$ je nulový funkcionál, tj. $\varphi \equiv 0$. Pripusťme proto, že $\varphi(\eta)/\|\eta\| \rightarrow 0$ a současně $\varphi(\eta_0) \neq 0$ pro nějaké $\eta_0 \in \mathcal{N}_0 \setminus \{0\}$. Pak $\lambda := \varphi(\eta_0)/\|\eta_0\| \neq 0$ a pro nenulovou posloupnost $\eta_n := \frac{1}{n} \eta_0$ platí

$$\|\eta_n\| = \frac{1}{n} \|\eta_0\| \rightarrow 0 \quad \text{pro } n \rightarrow \infty,$$

takže vzhledem k linearitě funkcionálu φ máme

$$\frac{\varphi(\eta_n)}{\|\eta_n\|} = \frac{\frac{1}{n} \varphi(\eta_0)}{\frac{1}{n} \|\eta_0\|} = \lambda \neq 0,$$

což je ve sporu s předpokladem $\varphi(\eta)/\|\eta\| \rightarrow 0$. \neq

Nechť nyní platí

$$\Delta \mathcal{F}_y[\eta] = \varphi_1(\eta) + \omega_1(\eta) \|\eta\|,$$

$$\Delta \mathcal{F}_y[\eta] = \varphi_2(\eta) + \omega_2(\eta) \|\eta\|$$

pro nějaké funkcionály $\varphi_1, \varphi_2, \omega_1, \omega_2$ splňující požadavky Definice 6.2.1. Pak

$$0 = \varphi_1(\eta) - \varphi_2(\eta) + [\omega_1(\eta) - \omega_2(\eta)] \|\eta\|,$$

takže $[\varphi_1(\eta) - \varphi_2(\eta)] / \|\eta\| = \omega_1(\eta) - \omega_2(\eta) \rightarrow 0$ pro $\|\eta\| \rightarrow 0$. Z první části důkazu proto vyplývá, že $\varphi_1(\eta) - \varphi_2(\eta) \equiv 0$, tj. $\varphi_1(\eta) = \varphi_2(\eta)$, což dokazuje uvedenou jednoznačnost. ■

Pro funkci $f : \mathbb{R}^n \rightarrow \mathbb{R}$ definujeme lokální extrém v bodě x_0 pomocí nerovností $f(x_0) \leq f(y)$ a $f(x_0) \geq f(y)$ platné pro všechna x z nějakého okolí $\mathcal{O}(x_0)$. Zcela analogicky to učiníme i pro funkcionály.

DEFINICE 6.2.3

Funkcionál $\mathcal{F}[\cdot]$ na nabývá na množině \mathcal{N} *lokálního extrému* pro funkci $\tilde{y} \in \mathcal{N}$, pokud výraz $\mathcal{F}[y] - \mathcal{F}[\tilde{y}]$ nemění znaménko v nějakém okolí \tilde{y} , tj. pro všechny funkce $y \in \mathcal{N}$ splňující $\|y - \tilde{y}\| < \varepsilon$ pro nějaké $\varepsilon > 0$.

Ačkoliv po formální stránce jsou definice lokálních extrémů pro funkci f a funkcionál \mathcal{F} „totožné“, je mezi nimi jeden zásadní rozdíl. Ten tkví v pojmu okolí bodu $x_0 \in \mathbb{R}^n$ a funkce $y \in \mathcal{N} \subseteq \mathcal{V}$, které je vždy určeno nějakou metrikou či normou. V \mathbb{R}^n to je typicky *euklidovská* norma, avšak lokální extrém se nezmění, pokud zvolíme jinou normu, neboť k konečně dimenzionálním prostorem jsou všechny normy ekvivalentní. Jenže vektorový prostor \mathcal{V} je nekonečnědimenzionální, takže volba konkrétní metriky zde již může hrát zcela zásadní roli. Toto si ilustrujeme v následujícím velmi důležitém příkladě s prostorem spojitě diferencovatelných funkcí $C^1[a, b]$, se kterým budeme pracovat v dalších částech této kapitoly.



Příklad 6.2.4

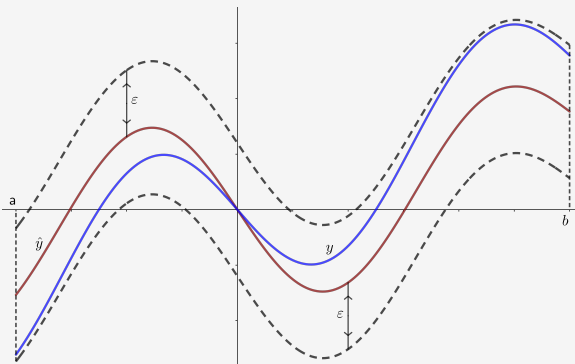
Na prostoru spojitých funkcí $C[a, b]$ můžeme zavést *maximální normu*

$$\|y\|_0 := \max_{x \in [a, b]} |y(x)|,$$

takže funkce y leží ve vzdálenosti ε od funkce \tilde{y} , jestliže

$$|y(x) - \tilde{y}(x)| < \varepsilon \quad \text{pro všechna } x \in [a, b],$$

tj. „funkční hodnoty na $[a, b]$ leží dostatečně blízko“, viz Obrázek 6.1.



Obrázek 6.1: Funkce \tilde{y} a její ε -okolí zahrnující i funkci y .

Na prostoru spojitě diferencovatelných funkcí $C^1[a, b]$ můžeme také zavést tuto normu,

⁴ V literatuře se lze setkat i s definicí okolí pomocí hodnot funkce a její derivace nikoli jako součtu obou maxim, ale požadavku aby obě maxima byla menší než ε , jak jsme to před chvílí částečně učinili my. Snad není příliš těžké se přesvědčit o tom, že v kontextu lokálních extrémů jsou tyto přístupy ekvivalentní.

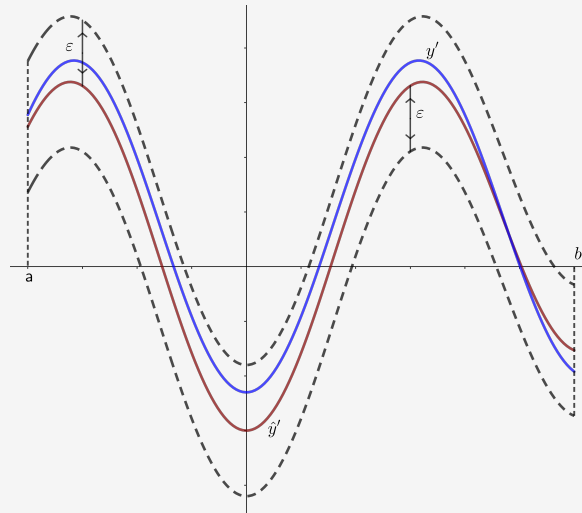
avšak mnohem výhodnější je využít v normě i derivaci funkce y , tj.

$$\|y\|_1 := \max_{x \in [a,b]} |y(x)| + \max_{x \in [a,b]} |y'(x)|.$$

Pak funkce y leží ve vzdálenosti ε od \bar{y} , jestliže

$$|y(x) - \bar{y}(x)| < \varepsilon/2 \quad \text{a} \quad |y'(x) - \bar{y}'(x)| < \varepsilon/2 \quad \text{pro všechna } x \in [a, b],$$

tj. „funkční hodnoty a hodnoty derivace na $[a, b]$ leží dostatečně blízko“⁴, viz také Obrázek 6.2.



Obrázek 6.2: Funkce y' a její ε -okolí zahrnující i derivaci funkce y .

⁵ Na intervalu $(0, \pi/4]$ máme $(x - \sin x)' = 1 - \cos x > 0$, tj. funkce $x - \sin x$ je na daném intervalu rostoucí. Podobně $(1 - \cos x)' = \sin x > 0$ pro $x \in (0, \pi/4]$.

Např. pro vzdálenosti funkcí $y_1(x) = x$ a $y_2(x) = \sin x$ na intervalu $[0, \pi/4]$ platí⁵

$$\|y_1 - y_2\|_0 = \max_{x \in [0, \pi/4]} |x - \sin x| = \pi/4 - \sqrt{2}/2,$$

zatímco derivací dostaneme vzdálenost

$$\|y_1 - y_2\|_1 = \max_{x \in [0, \pi/4]} |x - \sin x| + \max_{x \in [0, \pi/4]} |1 - \cos x| = \pi/4 + 1 - \sqrt{2}.$$

Avšak situace může i výrazně odlišná. Uvažme např. funkci

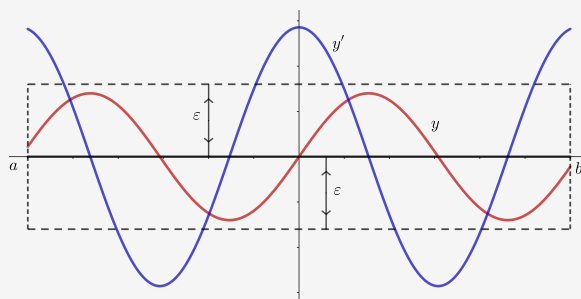
$$y(x) = c \sin(x/c^2)$$

na intervalu $[a, b]$ jistě patří do ε -okolí nulové funkce $y \equiv 0$ pro libovolné $0 < c < \varepsilon$ vzhledem k normě $\|\cdot\|_0$, neboť $\|y - 0\|_0 = \max_{x \in [a,b]} |y(x)| \leq c$. Avšak již nemusí nepatřit do ε -okolí nulové funkce $y \equiv 0$ určeného normou $\|\cdot\|_1$, neboť z rovnosti $y'(x) = \frac{1}{c} \cos(x/c^2)$ plyne

$$\|y - 0\|_1 = \max_{x \in [a,b]} |y(x)| + \max_{x \in [a,b]} |y'(x)| \leq c + 1/c.$$

Zejména je-li $k\pi c^2 \in [a, b]$ a $(\pi/2 + m\pi)c^2 \in [a, b]$ pro nějaké $k, m \in \mathbb{Z}$, pak dokonce $\|y - 0\|_1 = c + 1/c$. Má-li ale být splněno $c + 1/c < \varepsilon$ neboli $c^2 - c\varepsilon + 1 < 0$, pak nutně $\varepsilon^2 - 4 > 0$, tj.⁶ $\varepsilon > 2$.

⁶ Však také $\min_{c>0} \{c + 1/c\} = 2$.

Obrázek 6.3: Okolí nulové funkce vs. y a y' pro $c = 0,7$ a $\varepsilon = 0,8$.

Funkce y ukazuje, že okolí určené normou $\|\cdot\|_0$ obsahuje více funkcí než v případě normy $\|\cdot\|_1$, tj. $\mathcal{O}_{\|\cdot\|_0}(y) \supseteq \mathcal{O}_{\|\cdot\|_1}(y)$. Budeme-li pak chtít ověřit, zda \bar{y} je lokálním extrémem, budeme mít pro $\|\cdot\|_0$ více funkcí k testování neměnnosti znaménka výrazu $\mathcal{F}[y] - \mathcal{F}[\bar{y}]$, a tudíž i více možností v nichž může být tato podmínka porušena. Proto se v této souvislosti hovoří o $\|\cdot\|_0$ jako o *silné* normě a odpovídajícím \bar{y} jako o *silném* lokálním extrému, zatímco v případě $\|\cdot\|_1$ se hovoří o *slabé* normě a *slabém* lokálním extrému. Jaká je motivace pro právě tuto terminologii⁷? Odpovědí je vztah mezi silnými a slabými lokálními extrémami. Je-li \bar{y} silným lokálním extrémem, pak je nutně i slabým lokálním extrémem díky výše popsané inkluzi okolí \bar{y} , tj. pokud $\mathcal{F}[y] - \mathcal{F}[\bar{y}]$ nemění znaménko pro všechna $y \in C^1[a, b]$ splňující $\|y - \bar{y}\|_0 < \varepsilon$, pak totéž jistě platí i pro funkce vyhovující dokonce nerovnosti $\|y - \bar{y}\|_1 < \varepsilon$. Opačná implikace už samozřejmě neplatí, ovšem kontrapozicí alespoň dostaneme, že v případě absence slabého extrému nebude existovat ani silný extrém. Proto každá postačující podmínka pro silný extrém je i postačující podmínkou pro slabý extrém, zatímco každá nutná podmínka pro slabý extrém je nutnou podmínkou i pro silný extrém. Vzhledem k výše uvedenému snad nepřekvapí, že nalezení slabých extrémů je jednodušší a že se právě na ně zaměříme⁸.

Pro úplnost ještě dodejme, že analogicky můžeme pro funkce $y \in C^n[a, b]$ definovat *maximální normu k-tého řádu* pro $k \in \{0, \dots, n\}$ jako

$$\|y\|_k := \sum_{j=0}^k \max_{x \in [a, b]} |y^{(j)}(x)|.$$

Bez ohledu na konkrétní volbu prostoru \mathcal{V} a jeho normy musí každý lokální extrém nutně splňovat následující podmínku.

VĚTA 6.2.5

Nechť funkcionál $\mathcal{F}[\cdot]$ je diferencovatelný a má na \mathcal{N} lokální extrém v \bar{y} . Pak $\mathcal{F}'[\bar{y}; \eta] = 0$ pro všechny přípustné variace $\eta \in \mathcal{N}_0$.

Důkaz. Nechť pro jednoduchost $\mathcal{F}[\cdot]$ má na \mathcal{N} lokální minimum v \bar{y} . Pak z Definic 6.2.1 a 6.2.3 vyplývá, že

$$0 \leq \mathcal{F}[\bar{y} + \eta] - \mathcal{F}[\bar{y}] = \Delta \mathcal{F}_{\bar{y}}[\eta] = \mathcal{F}'[\bar{y}; \eta] + \omega(\eta) \|\eta\| \quad (6.2.4)$$

pro všechny přípustné variace $\eta \in \mathcal{N}_0$ vyhovující podmínce $\|\eta\| < \varepsilon$ pro nějaké $\varepsilon > 0$. Připusťme, že $\mathcal{F}'[\bar{y}; \eta_0] \neq 0$ pro nějaké $\eta_0 \in \mathcal{N}_0 \setminus \{0\}$. Pak pro libovolné $\delta \in (-\varepsilon/\|\eta_0\|, \varepsilon/\|\eta_0\|)$ bude $\delta \eta_0 \in \mathcal{N}_0$

⁷ Méně vhodných kandidátů \approx „slabá terminologie“.

⁸ Také Lagrange se ve své práci věnoval právě slabým extrémům. Počátky studia silných extrémů jsou spojeny až s prací Weierstrasse.

splňovat $\|\delta\eta_0\| < \varepsilon$ a z rovnosti (6.2.4) díky linearitě $\varphi(\cdot) = \mathcal{F}'[\bar{y}; \cdot]$ dostaneme

$$0 \leq \Delta \mathcal{F}_{\bar{y}}[\delta\eta_0] = \delta\varphi(\eta_0) + \omega(\delta\eta_0) \|\delta\eta_0\| \quad \text{neboli} \quad 0 \leq \text{sgn}(\delta)\varphi(\eta_0) + \omega(\delta\eta_0) \|\eta_0\|,$$

kde výraz $\text{sgn}(\delta)\varphi(\eta_0)$ je konstantní, zatímco hodnota $\omega(\delta\eta_0) \|\eta_0\|$ se pro $\delta \rightarrow 0$ blíží k nule. Pak ale jistě lze nalézt δ dostatečně malé a s opačným znaménkem než $\varphi(\eta_0)$ tak, že poslední nerovnost nebude splněna. \neq

Podobně jako jsme v případě funkcí $f: \mathbb{R}^n \rightarrow \mathbb{R}$ pojmenovali body splňující nutnou podmínku prvního řádu *stacionárními*, bude i nyní velmi užitečné zavést si pro funkce splňující nutnou podmínku z Věty 6.2.5 „zvláštní“ terminologii.

DEFINICE 6.2.6

Funkce $y \in \mathcal{N}$ splňující $\mathcal{F}'[y; \eta] = 0$ pro všechny přípustné variace $\eta \in \mathcal{N}_0$ se nazývá *extremálou*.

Jak ale ověřit, zda nalezená extremála je lokálním extrémem? K tomu budeme potřebovat pojem *kvadratického funkcionálu*. Zobrazení $\varphi: \mathcal{N}_0 \times \mathcal{N}_0 \rightarrow \mathbb{R}$, které je lineární v obou složkách, se nazývá *bilineární funkcionál*, tj. $\varphi(\eta, \zeta)$ pro pevné $\zeta \in \mathcal{N}_0$ a libovolné $c_1, c_2 \in \mathbb{R}$, $\eta_1, \eta_2 \in \mathcal{N}_0$ splňuje

$$\varphi(c_1\eta_1 + c_2\eta_2, \zeta) = c_1\varphi(\eta_1, \zeta) + c_2\varphi(\eta_2, \zeta)$$

a podobně pro pevné $\eta \in \mathcal{N}_0$ a libovolné $c_1, c_2 \in \mathbb{R}$, $\zeta_1, \zeta_2 \in \mathcal{N}_0$ platí

$$\varphi(\eta, c_1\zeta_1 + c_2\zeta_2) = c_1\varphi(\eta, \zeta_1) + c_2\varphi(\eta, \zeta_2).$$

Kvadratický funkcionál pak získáme volbou $\eta = \zeta$, tj. jako $\varphi(\eta, \eta)$. S tímto jsme se již setkali v konečnědimenzionálním případě (matice), kdy jsme hovořili o *kvadratických formách*.

Je-li navíc k (6.2.3) možné získat dokonce vyjádření

$$\Delta \mathcal{F}[y; \eta] = \varphi_1(\eta) + \varphi_2(\eta) + \omega(\eta) \|\eta\|^2$$

pro nějaký lineární funkcionál $\varphi_1(\eta)$, kvadratický funkcionál $\varphi_2(\eta)$ a nelineární funkcionál $\omega(\cdot)$ splňující $\omega(\eta) \rightarrow 0$ pro $\|\eta\| \rightarrow 0$, pak o funkcionálu $\mathcal{F}[\cdot]$ řekneme, že je *dvakrát diferencovatelný*, a kvadratický funkcionál $\varphi_2(\eta)$ nazveme jeho *druhou variací* v y a označíme ji $\mathcal{F}''[y; \eta]$ příp. $\delta^2 \mathcal{F}[y]$. Také $\mathcal{F}''[y; \eta]$ je určena jednoznačně a s její pomocí získáme nutnou i postačující podmínku pro lokální extrém. Je-li totiž \bar{y} lokální minimum funkcionálu $\mathcal{F}[\cdot]$, pak nutně

$$\mathcal{F}''[\bar{y}; \eta] \geq 0 \quad \text{pro všechna } \eta \in \mathcal{N}_0.$$

Na druhou stranu, je-li \bar{y} extremála a $\mathcal{F}''[\bar{y}; \eta]$ je pozitivně definitní⁹, pak \bar{y} je lokální minimum. Pochopitelně analogické podmínky platí pro lokální maximum.

⁹ To znamená, že existuje $k > 0$ takové, že $\mathcal{F}''[\bar{y}; \eta] \geq k\|\eta\|^2$ pro všechny přípustné variace $\eta \in \mathcal{N}_0$.

6.3 Základní úloha variačního počtu

Nyní s využitím předchozích výsledků odvodíme nutné a postačující podmínky pro řešení úlohy o brachistochroně a dalších podobných úloh, ve kterých mezi všemi funkcemi $y \in C^1[a, b]$ vyhovujícím podmínkám

$$y(a) = \alpha \quad y(b) = \beta \tag{6.3.1}$$

hledáme takovou, že

$$\mathcal{F}[y] := \int_a^b F(x, y(x), y'(x)) dx \rightarrow \text{opt} \quad (6.3.2)$$

pro danou spojitou funkci $F: \mathbb{R}^3 \rightarrow \mathbb{R}$ se spojitými parciálními derivacemi až do druhého řádu vzhledem ke všem třem proměnným. Funkce F se nazývá lagrangián a právě tomuto typu úloh se Lagrange ve své práci věnoval. Naše řešení má proto mnoho společného s Lagrangeovým čistě analytickým přístupem nevyžadujícím žádné geometrické nástroje, který poprvé představil Eulerovi v dopise z 12. srpna 1755. V tomto dopise ilustroval sílu své metody na řešení třech úloh z Eulerovy knihy [35] a následně s její pomocí v dopise z 20. listopadu 1755 vyřešil i úlohu o brachistochroně s tzv. *volným koncem*, viz [41] a <https://odkaz.page.link/EulerLagrange>. Toto je sice z dešního pohledu značné zjednodušení, neboť máme velmi specifické počáteční podmínky, řád derivací i tvar lagrangianu, ale v praxi se řada fyzikálních nebo geometrických úloh redukuje právě do této podoby. Připomeňme, že řešení úlohy (6.3.1)–(6.3.2) závisí na volbě normy v prostoru $C^1[a, b]$. Tomuto problému jsme se podrobně věnovali již v Příkladu 6.2.4 a my se v našich následujících úvahách omezíme na hledání slabých extrémů.

Abychom mohli využít tvrzení Věty 6.2.5 budeme potřebovat pár pomocných výsledků týkajících se „nulovosti některých integrálů“.

LEMMA 6.3.1

Nechť $\alpha \in C[a, b]$. Pokud $\int_a^b \alpha(x) \eta(x) dx = 0$ pro všechna $\eta \in C[a, b]$ splňující $\eta(a) = 0 = \eta(b)$, pak $\alpha(x) = 0$ pro všechna $x \in [a, b]$.

Důkaz. Připusťme, že $\alpha \neq 0$ na $[a, b]$, tj. existuje $x_0 \in [a, b]$ takové, že $\alpha(x_0) \neq 0$. V takovém případě ale ze spojitosti funkce $\alpha(\cdot)$ vyplývá, že musí být $\alpha(x) \neq 0$ na nějakém netriviálním podintervalu $[x_1, x_2] \subseteq [a, b]$. Bez újmy na obecnosti vezměme $\alpha(x) > 0$ na $[x_1, x_2]$. Položíme-li

$$\eta(x) := \begin{cases} (x - x_1)(x_2 - x), & x \in [x_1, x_2], \\ 0, & \text{jinak,} \end{cases}$$

pak jistě $\eta \in C[a, b]$ a $\eta(x_1) = 0 = \eta(x_2)$, takže podle předpokladů by měl být $\int_a^b \alpha(x) \eta(x) dx = 0$. Jenže $\alpha(x) > 0$ a $\eta(x) > 0$ na (x_1, x_2) , což znamená, že dostáváme spor, neboť

$$\int_a^b \alpha(x) \eta(x) dx = \int_{x_1}^{x_2} \alpha(x) \eta(x) dx > 0. \quad \text{✗}$$

Proto nutně $\alpha \equiv 0$ na $[a, b]$. ■

Tvrzení předchozího lemma zůstane v platnosti i v případě, že uvažujeme $\eta \in C^n[a, b]$. V takovém případě ale nelze v důkazu vzít totéž $\eta(\cdot)$, protože $\eta'(x) = -2x + x_1 + x_2 \neq 0$ pro $x \in \{x_1, x_2\}$ implikuje $\eta \notin C^1[a, b]$. Místo toho je potřeba zvolit

$$\eta(x) := \begin{cases} (x - x_1)^{n+1} (x_2 - x)^{n+1}, & x \in [x_1, x_2], \\ 0, & \text{jinak,} \end{cases}$$

což již vyhovuje požadavku $\eta \in C^n[a, b]$.

Následující tvrzení bývá v literatuře označován jako *fundamentální lemma varičního počtu*. Avšak pro nás bude mnohem důležitější jeho důsledek.

LEMMA 6.3.2

Nechť $\alpha \in C[a, b]$. Pokud $\int_a^b \alpha(x) \eta'(x) dx = 0$ pro všechna $\eta \in C^1[a, b]$ splňující $\eta(a) = 0 = \eta(b)$, pak $\alpha \equiv c$ na $[a, b]$ pro vhodné $c \in \mathbb{R}$.

Důkaz. Nechť c je střední hodnota funkce $\alpha(\cdot)$ na $[a, b]$, tj. $c := \frac{1}{b-a} \int_a^b \alpha(x) dx$ a vzhledem ke spojitosti $\alpha(\cdot)$ dokonce máme $c = \alpha(x_0)$ pro nějaké $x_0 \in [a, b]$, viz [31, Důsledek 2.36]. Položíme-li $\eta(x) := \int_a^x [\alpha(t) - c] dt$, pak ze spojitosti $\alpha(x) - c$ plyne $\eta \in C^1[a, b]$, viz [31, Věta 2.39]. Navíc zjevně máme $\eta(a) = 0$ a také

$$\eta(b) = \int_a^b [\alpha(x) - c] dx = \int_a^b \alpha(x) dx - c(b-a) = 0.$$

Odtud s využitím předpokladů lemma získáme přímým výpočtem

$$\int_a^b [\alpha(x) - c]^2 dx = \int_a^b [\alpha(x) - c] \eta'(x) dx = \underbrace{\int_a^b \alpha(x) \eta'(x) dx}_{=0} - c \int_a^b \eta'(x) dx = -c [\eta(x)]_a^b = 0,$$

jenže $[\alpha(x) - c]^2$ je spojitá a nezáporná funkce na $[a, b]$, tudíž nutně $[\alpha(x) - c]^2 \equiv 0$ na $[a, b]$ neboli $\alpha(x) \equiv c$. ■

DŮSLEDEK 6.3.3

Nechť $\alpha, \beta \in C[a, b]$. Pokud

$$\int_a^b [\alpha(x) \eta(x) + \beta(x) \eta'(x)] dx = 0$$

pro všechna $\eta \in C^1[a, b]$ splňující $\eta(a) = 0 = \eta(b)$, pak funkce β je diferencovatelná (do konce $\beta \in C^1[a, b]$) a platí $\beta'(x) = \alpha(x)$ pro všechna $x \in [a, b]$.

Důkaz. Nechť $A(x)$ je primitivní funkce k $\alpha(x)$, tj.

$$A(x) = \int_a^x \alpha(t) dt, \quad x \in [a, b],$$

viz [31, Důsledek 2.40]. Pak s využitím integrace per-partes získáme

$$\int_a^b \alpha(x) \eta(x) dx \left| \begin{array}{l} u = \eta(x) \quad u' = \eta'(x) \\ v = A(x) \quad v' = \alpha(x) \end{array} \right| = \underbrace{[\tau(x) A(x)]_a^b}_{=0} - \int_a^b A(x) \eta'(x) dx = - \int_a^b A(x) \eta'(x) dx,$$

díky čemuž máme

$$0 = \int_a^b [\alpha(x) \eta(x) + \beta(x) \eta'(x)] dx = \int_a^b [-A(x) + \beta(x)] \eta'(x) dx.$$

Jelikož poslední rovnost platí pro libovolnou funkci η vyhovující daným předpokladům, plyne z Lemma 6.3.2, že $-A(x) + \beta(x) \equiv c$ na $[a, b]$. Tudíž $\beta \in C^1[a, b]$ a $\beta'(x) = A'(x) = \alpha(x)$ pro všechna $x \in [a, b]$. ■

Všechna tři uvedená tvrzení jsou platná dokonce i jako ekvivalence, přičemž opačný se snadno ověří přímým výpočtem. Zejména v Lemma 6.3.2 využijeme rovnosti $\int_a^b \eta'(x) dx = [\eta(x)]_a^b$ a v Důsledku 6.3.3 pro volbou $\alpha = \beta'$ dostaneme

$$\int_a^b [\beta'(x) \eta(x) + \beta(x) \eta'(x)] dx = \int_a^b [\beta(x) \eta(x)]' dx = [\beta(x) \eta(x)]_a^b.$$

Nyní určíme první variaci funkcionálu z úlohy (6.3.2). Připomeňme, že $\mathcal{V} = C^1[a, b]$, $\mathcal{N} = \{y \in \mathcal{V} \mid y(a) = \alpha, y(b) = \beta\}$ a $\mathcal{N}_0 = \{\eta \in \mathcal{V} \mid \eta(a) = 0 = \eta(b)\}$. Pak

$$\Delta \mathcal{F}[y; \eta] = \mathcal{F}[y + \eta] - \mathcal{F}[y] = \int_a^b [F(x, y + \eta, y' + \eta') - F(x, y, y')] dx,$$

na což můžeme díky spojitosti parciálních derivací aplikovat Taylorovu větu, čímž obdržíme¹⁰

$$\begin{aligned} \Delta \mathcal{F}[y; \eta] &= \int_a^b \left[\underbrace{F_x(x, y, y')(x - x)}_{=0} + \underbrace{F_y(x, y, y')(y + \eta - y)}_{=\eta} \right. \\ &\quad \left. + F_z(x, y, y') \underbrace{(y' + \eta' - y')}_{=\eta'} + R(\eta^2, \eta'^2) \right] dx \\ &= \int_a^b [F_y(x, y, y')\eta + F_z(x, y, y')\eta'] dx + \int_a^b R(\eta^2, \eta'^2) dx. \end{aligned}$$

Toto vyjádření odpovídá rovnosti (6.2.3), přičemž první integrál na pravé straně je lineární vzhledem k η , zatímco druhý integrál obsahuje kvadratické a smíšené členy. Proto variace daného funkcionálu je rovna

$$\mathcal{F}'[y; \eta] = \int_a^b [F_y(x, y, y')\eta + F_z(x, y, y')\eta'] dx.$$

Má-li tedy $\bar{y} \in C^1[a, b]$ být lokálním řešením úlohy (6.3.1)–(6.3.2) musí dle Věty 6.2.5 platit

$$0 = \mathcal{F}'[\bar{y}; \eta] = \int_a^b [F_y(x, \bar{y}, \bar{y}')\eta + F_z(x, \bar{y}, \bar{y}')\eta'] dx \quad (6.3.3)$$

pro všechna $\eta \in \mathcal{N}_0$. Volbou $\alpha(x) = F_y(x, \bar{y}, \bar{y}')$ a $\beta(x) = F_z(x, \bar{y}, \bar{y}')$ v Důsledku 6.3.3 pak vyplývá, že rovnost (6.3.3) je splněna právě tehdy, když β je diferencovatelná a $\beta'(x) = \alpha(x)$ pro všechna $x \in [a, b]$, tj. funkce \bar{y} musí být řešením *Eulerovy–Lagrangeovy rovnice*

$$\frac{d}{dx} F_z(x, \bar{y}, \bar{y}') = F_y(x, \bar{y}, \bar{y}') \quad \text{na } [a, b]. \quad (6.3.4)$$

Řešením této rovnice obdržíme všechny (silné i slabé) extremály úlohy (6.3.1)–(6.3.1). Obecně se jedná o nelineární diferenciální rovnici 2. řádu. Podaří-li se nám najít její řešení, pak bude záviset na dvou konstantách, k jejichž určení využijeme okrajové podmínky. Eulerova–Lagrangeova rovnice tak převádí úlohu varičního počtu (6.3.1)–(6.3.1) na úlohu nalezení partikulárního řešení příslušné diferenciální rovnice (případně zjištění, zda vůbec takové řešení existuje). To ale v obecném případě může být dosti obtížné. V úloze (6.3.1)–(6.3.1) není nic vyžadováno pro \bar{y}'' , ovšem pokud budeme mít zaručenu jeho existenci, můžeme levou stranu rovnice (6.3.4) rozpočítat pomocí derivace složené funkce, čímž získáme explicitní tvar zmíněné nelineární diferenciální rovnice 2. řádu

$$F_y(x, \bar{y}, \bar{y}') - F_{zx}(x, \bar{y}, \bar{y}') - F_{zy}(x, \bar{y}, \bar{y}') \bar{y}'' - F_{zz}(x, \bar{y}, \bar{y}') \bar{y}'' = 0. \quad (6.3.5)$$

Ať už je to s existencí \bar{y}'' jakkoli, skýtají některé speciální typy funkce F výrazné zjednodušení Eulerovy–Lagrangeovy rovnice:

- (i) Pokud F explicitně nezávisí na y , tj.

$$\mathcal{F}[y] = \int_a^b \bar{F}(x, y') dx,$$

pak $F_y(x, \bar{y}, \bar{y}') \equiv 0$ a Eulerova–Lagrangeova je tvaru

$$\frac{d}{dx} F_z(x, \bar{y}, \bar{y}') = 0 \quad \text{na } [a, b],$$

tj. $F_z(x, \bar{y}, \bar{y}') \equiv \text{konst.}$, což je diferenciální rovnice 1. řádu.

¹⁰ Abychom předešli případným nejasnostem, měli bychom zdůraznit, že výrazem $F_z(x, y, y')$ rozumíme parciální derivaci funkce $F(\cdot, \cdot, \cdot)$ vzhledem ke třetí proměnné vyčíslenou v „bodě“ $[x, y, y']$.

(ii) Pokud F explicitně nezávisí na y' , tj.

$$\mathcal{F}[y] = \int_a^b \widehat{F}(x, y) dx,$$

pak $F_y(x, \bar{y}, \bar{y}') \equiv 0$ a a Eulerova–Lagrangeova je tvaru

$$F_y(x, \bar{y}, \bar{y}') = 0 \quad \text{na } [a, b],$$

neboli $F(x, \bar{y}, \bar{y}') \equiv \text{konst.}$, což je dokonce „pouze“ algebraická rovnice.

(iii) Pokud F explicitně nezávisí na x , tj.

$$\mathcal{F}[y] = \int_a^b \widehat{F}(y, y') dx,$$

pak lze s využitím podobných úprav jako v (6.3.5) přepsat Eulerovu–Lagrangeovu rovnici do tvaru

$$\begin{aligned} 0 &= y' \left[F_y(x, \bar{y}, \bar{y}') - \frac{d}{dx} F_z(x, \bar{y}, \bar{y}') \right] = \\ &= y' \left[F_y(x, \bar{y}, \bar{y}') - \bar{y}' F_{zy}(x, \bar{y}, \bar{y}') - \bar{y}'' F_{zz}(x, \bar{y}, \bar{y}') \right] + \bar{y}'' F_z(x, \bar{y}, \bar{y}') - \bar{y}'' F_z(x, \bar{y}, \bar{y}') = \\ &= \frac{d}{dx} [F(x, \bar{y}, \bar{y}') - \bar{y}' F_z(x, \bar{y}, \bar{y}')] \end{aligned}$$

z čehož ihned plyne tzv. *Beltramiho rovnost*

$$F(x, \bar{y}, \bar{y}') - \bar{y}' F_z(x, \bar{y}, \bar{y}') \equiv \text{konst.} \quad \text{na } [a, b].$$

(iv) Je-li F lineární vzhledem k y , tj. $F(x, y, y') = A(x, y) y' + B(x, y)$, pak Eulerova–Lagrangeova rovnice má podobu

$$A_x(x, \bar{y}) + A_y(x, \bar{y}) \bar{y}' = A_y(x, \bar{y}) y' + B_y(x, \bar{y}) \quad \text{neboli} \quad A_x(x, \bar{y}) = B_y(x, \bar{y}).$$

To vlastně není diferenciální rovnice, ale spíše implicitní vyjádření, které určuje funkci y proměnné x .

Příklad 6.3.4

Určeme extrémaly variačního problému

$$\mathcal{F}[y] = \int_1^2 y' (1 + x^2 y') dx \rightarrow \text{opt}, \quad y \in C^1[1, 2], \quad y(1) = 4, \quad y(2) = 3.$$

Řešení. Máme $F(x, y, y') = \widehat{F}(x, y') = y' (1 + x^2 y')$, tudíž $F_y(x, y, y') = 0$ a $F_z(x, y, y') = 1 + x^2 y' + y' x^2$. Využijeme-li přímo Eulerovu–Lagrangeovu rovnici, musí každá extrémála $y \in C^1[1, 2]$ dané úlohy vyhovovat diferenciální rovnici

$$\frac{d}{dx} F_z(x, y, y') = 4x y' + 2x^2 y'' = 0 = F_y(x, y, y') \quad \text{neboli} \quad x^2 y'' + 2x y' = 0.$$

To je tzv. *Eulerova diferenciální rovnice*, k jejímuž řešení využijeme substituci $t = \ln|x|$. Jelikož $x \in [1, 2]$, je $x = e^t$ a platí

$$y' = \frac{dy}{dx} = \frac{dy}{dt} \frac{dt}{dx} = \dot{y} \frac{1}{x} \quad \text{a} \quad y'' = \frac{d}{dx} \left(\dot{y} \frac{1}{x} \right) = \frac{1}{x} \frac{\dot{y}}{dx} + \dot{y} \frac{d}{dx} \left(\frac{1}{x} \right) = \frac{1}{x^2} (\ddot{y} - \dot{y}).$$

Dosažením do Eulerova diferenciální rovnice dostaneme lineární diferenciální rovnici s konstantními koeficienty

$$0 = x^2 y'' + 2x y' = \ddot{y} + \dot{y},$$

jejíž charakteristická rovnice $\lambda^2 + \lambda = 0$ má kořeny $\lambda = 0$ a $\lambda = -1$, takže její obecné řešení je

$$y(t) = c_1 e^{-t} + c_2, \quad c_1, c_2 \in \mathbb{R}.$$

Zpětným dosazením a využitím daných okrajových podmínek obdržíme

$$y(x) = \frac{c_1}{x} + c_2, \quad 4 = y(1) = c_1 + c_2 \quad \& \quad 3 = y(2) = c_1/2 + c_2,$$

čemuž vyhovuje pouze funkce

$$\bar{y}(x) = \frac{2}{x} + 2.$$

Z předcházející poznámky víme, že funkcionál v dané úloze nabízí výrazné zjednodušení Eulerovy–Lagrangeovy rovnice, které jsme ale v tuto chvíli nevyužili. Kdybychom tak učinili, hledali bychom pouze řešení diferenciální rovnice 1. řádu $F_z(x, y, y') = c$ neboli

$$1 + 2x^2 y' = c, \quad \text{tj.} \quad y' = \frac{c-1}{x^2},$$

což přímým integrováním vede k funkci

$$y(x) = -\frac{c-1}{2x} + d,$$

z níž opět s využitím daných okrajových podmínek získáme hledané řešení odpovídající volbě $c = -3$ a $d = 2$.

Toto by chtělo odvodit.

Pro daný funkcionál z (6.3.2) je jeho druhá variace

$$\mathcal{F}''[y; \eta] = \int_a^b [P(x)\eta'^2 + Q(x)\eta^2] dx$$

pro $P(x) := \frac{1}{2} F_{zz}(x, y, y')$ a $Q(x) := \frac{1}{2} [F_{yy}(x, y, y') - \frac{d}{dx} F_{yz}(x, y, y')]$. Potom nutná podmínka pro to, aby extrémála \bar{y} byla slabým lokálním minimem je tzv. *Legendreova podmínka*

$$F_{zz}(y, \bar{y}, \bar{y}') \geq 0 \quad \text{pro všechna } x \in [a, b].$$

V případě ostré nerovnosti hovoříme o *zesílené Legendreově podmínce*. Legendre se (neúspěšně) snažil ukázat, že tato zesílená podmínka je postačující pro slabé lokální minimum. Jenže to není pravda a je potřeba k ní ještě jeden požadavek doplnit.

VĚTA 6.3.5

Nechť $\bar{y} \in \mathcal{N}$ je přípustná funkce taková, že

- (i) je splněna Eulerova–Lagrangeova rovnice (6.3.4),
- (ii) $F_{zz}(x, \bar{y}, \bar{y}') = 2P(x) > 0$ pro všechna $x \in [a, b]$,
- (iii) existuje řešení *Jacobiho diferenciální rovnice*

$$Q(x)\eta - \frac{d}{dx} [P(x)\eta'] = 0$$

vyhovující počátečním podmínkám $\eta(a) = 0$ a $\eta'(a) = 1$, pro které je $x = a$ jediným nulovým bodem, tj. $\eta(x) \neq 0$ pro $x \in (a, b]$.

Potom $\mathcal{F}''[\bar{y}; \eta] > 0$ pro všechny přípustné variace $\eta \in \mathcal{N}_0$, tj. \bar{y} je slabým lokálním minimem funkcionálu $\mathcal{F}[\cdot]$ neboli slabým lokálním řešením minimalizační úlohy (6.3.1)–(6.3.2).

Poslední podmínku v předchozím tvrzení lze nahradit existencí řešení tzv. *Riccatiho rovnice*

$$w^2 = P(x)[Q(x) + w'] \quad \text{nebo} \quad w' = F_{yy}(x, \bar{y}, \bar{y}') - \frac{[F_{yz}(x, \bar{y}, \bar{y}') - w]^2}{F_{zz}(x, \bar{y}, \bar{y}')}$$

Toto rozhodně ještě není hotové!

definovaného na celém intervalu $[a, b]$.

Příklad 6.3.6

Ověřme, že extrémála nalezená v Příkladě 6.3.4 je slabým lokálním extrémem a určete jeho charakter.

Řešení. Jedinou nalezenou extrémálou je funkce $\bar{y} = 2/x + 2$. Pak

$$F_{zz}(x, \bar{y}, \bar{y}') = 2x^2 > 0, \quad x \in [1, 2],$$

tj. je splněna zesílená Legendreova podmínka. Jelikož $P(x) = F_{zz}(x, \bar{y}, \bar{y}')/2 = x^2$ a $Q(x) = \frac{1}{2}[F_{yy}(x, \bar{y}, \bar{y}') - \frac{d}{dx}F_{yz}(x, \bar{y}, \bar{y}')] = 0$, máme Jacobiho rovnici

$$-\frac{d}{dx}(x^2 \eta') = 0 \quad \text{neboli} \quad x^2 \eta' = c,$$

jejímž obecným řešením je $\eta(x) = -c/x + d$. S využitím počátečních podmínek pak nalezneme její řešení $\eta(x) = -1/x + 1$, které jistě na intervalu $(1, 2]$ nemá žádný nulový bod. Poněvadž $F_{zz}(x, \bar{y}, \bar{y}') > 0$ jedná se o slabé lokální minimum.

Kdybychom místo Jacobiho rovnice chtěli využít Riccatiho rovnici dostaneme

$$w^2 = x^2 w' \quad \text{neboli} \quad \frac{w'}{w^2} = x^2.$$

Její obecným řešením je funkce $w(x) = x/(1 - cx)$, která je pro libovolnou volbu integrační konstanty $c \in (-\infty, 1/2) \cup (1, \infty)$ definovaná na celém intervalu $[1, 2]$.

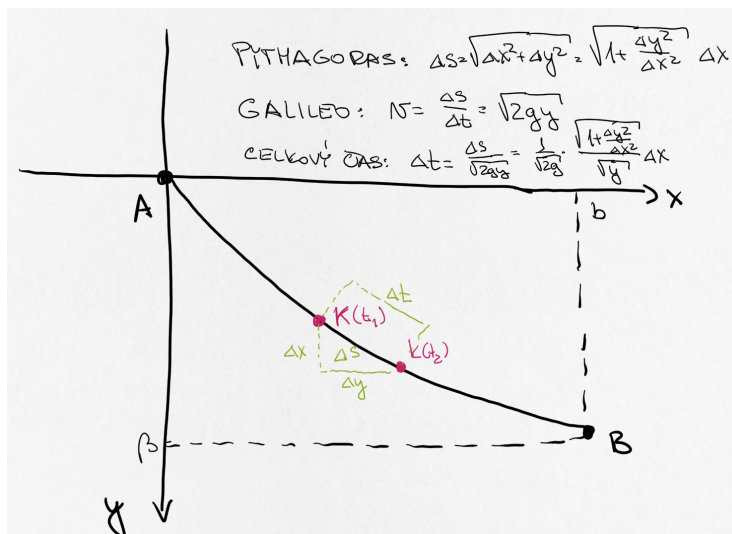
6.4 Úloha o brachistochroně

Nyní už je na čase pustit se do řešení slovučné úlohy o brachistochroně.

Uvažme proto kuličku (tedy spíše hmotný bod) o hmotnosti m v tíhovém poli Země startující v bodě $A = [0, 0]$, která se má při zanedbání tření a s nulovou počáteční rychlostí dostat v nejkratším čase do bodu $B = [b, \beta]$, kde $b > 0$ a $\beta \geq 0$ ¹¹. Po jaké dráze toho docílíme?

Celou situaci máme zakreslenou na Obrázku 6.4 níže, kde jsme pro přehlednost otočili orientaci osy y .

¹¹ Pro určitost volíme $b > 0$, abychom se pohybovali směrem doprava od A (mohli bychom ale volit i $b < 0$). V našem řešení si otočíme orientaci osy y , takže podmínka $\beta \geq 0$ zaručuje, že nechceme, aby kulička došla výše než byl výchozí bod, což totiž vzhledem k předpokladu nulové počáteční rychlosti není možné.



Obrázek 6.4: Situační obrázek pro úlohu o brachistochroně.

Podle definice rychlosti platí

$$v(t) = \frac{ds}{dt} \quad \text{neboli} \quad dt = \frac{ds}{v},$$

kde ds značí délku dráhy uražené za čas dt . Také víme, že pro délku segmentu křivky určené částí grafu funkce $y = y(x)$ v rozsahu dx platí

$$ds = \sqrt{1 + y'^2(x)} dx.$$

Rychlost v pohybujícího se hmotného bodu v tíhovém poli Země dokážeme stanovit pomocí zákona zachování energie pro izolovaný systém. Protože součet kinetické energie $E_k = mv^2/2$ a potenciální energie $E_p = mgy$ se nemění, musí vzhledem k otočené orientaci osy y platit¹²

$$0 = mv^2 - mgy \quad \text{neboli} \quad v = \sqrt{2gy},$$

kde $g \approx 9,8 \text{ m/s}^2$ představuje tíhové zrychlení na Zemi. Proto celková doba pohybu kuličky po dráze je rovna

$$t = \int_0^b \sqrt{\frac{1 + y'^2(x)}{2gy(x)}} dx.$$

Naším cílem je tedy nalézt mezi všemi funkcemi $y \in C^1[0, b]$ splňujícími $y(0) = 0$ a $y(b) = \beta$ minimum funkcionálu

$$\mathcal{F}[y] = \int_0^b \sqrt{\frac{1 + y'^2}{2gy}},$$

který explicitně nezávisí na x . V takové případě se Eulerova-Lagrangeova rovnice redukuje na Beltramiho rovnost

$$\sqrt{\frac{1 + y'^2}{2gy}} - \frac{y'^2}{\sqrt{2gy(1 + y'^2)}} = c \quad \text{neboli} \quad \sqrt{y(1 + y'^2)} = K$$

pro nějaké $c > 0$ a $K := (C\sqrt{2g})^{-1}$. Nahradíme-li $y' = \operatorname{tg} \varphi$, pak z předchozí rovnosti získáme $\sqrt{y(1 + \operatorname{tg}^2 \varphi)} = K$, což dává parametrické vyjádření y -ové souřadnice hledaného řešení jako¹³

$$y = \frac{K^2}{1 + \operatorname{tg}^2 \varphi} = K^2 \cos^2 \varphi = L(1 + \cos 2\varphi)/2$$

¹² Součet těchto energií musí být konstantní. Jenže vzhledem k počátečním podmínkám $y(0) = 0$ a $v(0) = 0$ máme $E_k(0) = 0 = E_p(0)$, takže tento součet je dokonce nulový.

¹³ Zde se pochopitelně sluší připomenout některé užitečné rovnosti pro goniometrické funkce: $\frac{1}{1 + \operatorname{tg}^2 \varphi} = \cos^2 \varphi = (1 + \cos 2\varphi)/2$, $\sin 2\varphi = 2 \sin \varphi \cos \varphi$, $\int \cos^2 \varphi d\varphi = (2\varphi + \sin 2\varphi)/4$, $\sin(\pi - t) = \sin t$ a $\cos(\pi - t) = -\cos t$.

pro $L := K^2$. Derivováním podle x odtud dostaneme diferenciální rovnici

$$\operatorname{tg} \varphi = y' = -L \varphi' \sin 2\varphi = -L \sin 2\varphi \frac{d\varphi}{dx}, \quad \text{tj.} \quad dx = -L \frac{2 \sin \varphi \cos \varphi}{\operatorname{tg} \varphi} d\varphi = -2L \cos^2 \varphi d\varphi,$$

z čehož přímým integrováním získáme parametrické vyjádření x -ové osy ve tvaru

$$x = -2L[\varphi/2 + (\sin 2\varphi)/4] + C = -L(2\varphi + \sin 2\varphi)/2 + C.$$

Zavedeme si nyní dva nové parametry¹⁴ t a r tak, že položíme $t := \pi - 2\varphi$ (proměnná) a $r := L/2$ (konstanta). Potom

$$x = -r[\pi - t + \sin(\pi - t)] + C = r(t - \sin t) + C - r\pi \quad \text{a} \quad y = r(1 - \cos t).$$

Jelikož $x(0) = 0$, platí $C = r\pi$, takže máme parametrické vyjádření řešení

$$x = -r[\pi - t + \sin(\pi - t)] + C = r(t - \sin t) \quad \text{a} \quad y = r(1 - \cos t),$$

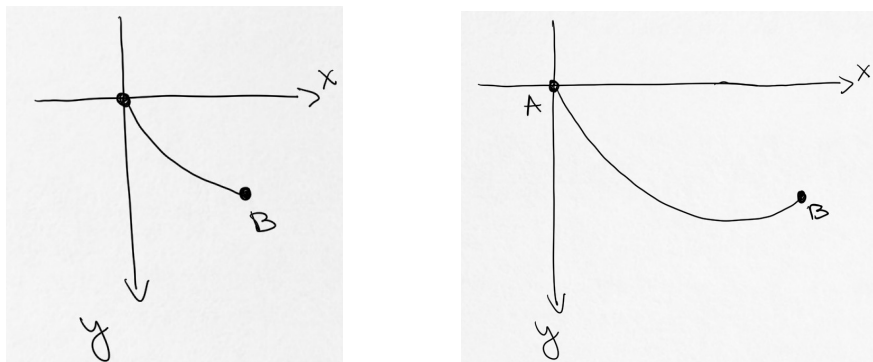
což je rovnice cykloidy určené kružnicí o poloměru r . Podmínka $y(0) = 0$ znamená, že výchozí hodnota parametru t je $2k\pi$ pro nějaké $k \in \mathbb{Z}$. Pro jednoduchost můžeme zvolit $k = 0$ a jakákoli jiná volba $k \neq 0$ vše pouze posune o $2k\pi$. Maximální hodnotu parametru t pak získáme řešením rovnice

$$\frac{y(t_{\max})}{x(t_{\max})} = \frac{\beta}{b} = \frac{1 - \cos t_{\max}}{t_{\max} - \sin t_{\max}},$$

které bude $t_{\max} \in (0, 2\pi)$. Odtud také dostaneme poloměr vymežující kružnice

$$r = \frac{\beta}{1 - \cos t_{\max}} = \frac{b}{t_{\max} - \cos t_{\max}}.$$

Jestliže $t_{\max} < \pi$ neboli $\frac{\beta}{b} > \frac{2}{\pi}$, tak kulička podél celé dráhy pouze klesá, a to až do nejnižšího bodu B . Je-li $t_{\max} \geq \pi$ neboli $\frac{\beta}{b} < \frac{2}{\pi}$, tak kulička podél brachistochrony klesá do nejnižšího bodu $[r\pi, 2r]$ odpovídajícího $t = \pi$ a pak stoupá až do bodu B .



Obrázek 6.5: Ilustrace situací $t_{\max} \leq \pi$.

Např. pro $B = [1, 1]$ je $t_{\max} \approx 2,412011$ a $r \approx 0,5729$. Čas potřebný pro cestu z A do B pak je roven $0,5831927828$ s. Podobně pro bod $B = [\pi, 2]$ dostaneme $t_{\max} = \pi$ a $r = 1$ s potřebným časem $0,7705$ s.

¹⁴ Toto je jiné „ t “ než na začátku, tj. není to čas.

LITERATURA

- [1] *Acta Eruditorum*, ročník 1696. Dostupné online na adrese <https://odkaz.page.link/ae96>.
- [2] *Acta Eruditorum*, ročník 1697. Dostupné online na adrese <https://odkaz.page.link/ae97>.
- [3] D. J. Albers, C. Reid, G. B. Dantzig, *An interview with George B. Dantzig: The father of linear programming*, *College Math. J.* **17** (1986), č. 4, str. 292–314. Dostupné online na adrese <https://odkaz.page.link/dantzig1>.
- [4] M. Ashbaugh, R. Benguria, *The problem of queen Didó: Overview of the Subject of Isoperimetry*, 2010. Dostupné online na adrese <https://odkaz.page.link/dido1>.
- [5] A. A. Assad, S. I. Gass, *Profiles in Operations Research: Pioneers and Innovators*, International Series in Operations Research & Management Science **147**, Springer, New York, 2011. ISBN 978-1-4419-6280-5.
- [6] *Bible*, kniha *Genesis* **3**, Český ekumenický překlad, dostupné online na adrese <https://odkaz.page.link/bible1>.
- [7] *Bible*, kniha *Genesis* **41**, Český ekumenický překlad, dostupné online na adrese <https://odkaz.page.link/bible2>.
- [8] D. R. Bellhouse, *The problem of Waldegrave*, *J. Électron. Hist. Probab. Stat.* **3** (2007), č. 2, 12 str. (elektronicky).
- [9] D. R. Bellhouse, N. Fillion, *Le her and other problems in probability discussed by Bernoulli, Montmort and Waldegrave*, *Statist. Sci.* **30** (2015), č. 1, str. 26–39.
- [10] R. E. Bellman, *Dynamic Programming*, Princeton University Press, Princeton, 1957.
- [11] R. E. Bellman, *Eye of the Hurricane: An Autobiography*, World Scientific Publishing, Singapore, 1984. ISBN 9971-966-01-8.
- [12] D. Bertsekas, *Nonlinear Programming*, Athena Scientific, Belmont, 1999.
- [13] D. Bertsekas, *Convex Optimization Theory*, Athena Scientific, Belmont, 2009.
- [14] D. Bertsekas, A. Nedic, A. Ozdeklar, *Convex Analysis and Optimization*, Athena Scientific, Belmont, 2003.

- [15] C. Blair Jr., *Passing of a great mind*, LIFE **42** (25. února 1957), č. 8, str. 89–104. Dostupné online na adrese <https://odkaz.page.link/vonneumann1>.
- [16] R. W. Cottle, *Symmetric dual quadratic programs*, Research Report **19**, Operations Research Center, University of California, Berkeley, 1962.
- [17] R. W. Cottle, B. C. Eaves, M. N. Thapa, *Dantzig, George B. (1914–2005)*. V „The New Palgrave Dictionary of Economics“, M. Palgrave (editor), Palgrave Macmillan, Londýn, 2008.
- [18] G. B. Dantzig, *Programming of interdependent activities II: Mathematical model*, *Econometrica* **17** (1949), č. 3-4, str. 200–211.
- [19] G. B. Dantzig, *Linear programming under uncertainty*, *Management Sci.* **1** (1955), č. 3-4, str. 197-206.
- [20] G. B. Dantzig, *Linear Programming and Extensions*, Princeton University Press, Princeton, 1963. Dostupné online na adrese <https://odkaz.page.link/dantzig2>.
- [21] G. B. Dantzig, *Reminiscences about the origins of linear programming*, *Operations Research Letters* **1** (1982), č. 2, str. 43–48.
- [22] G. B. Dantzig, *Origins of the simplex method*, Technical Report SOL 87-5, Department of Operations Research, Stanford University, Stanford, 1987. Publikováno také v „A History of Scientific Computing“, Association for Computing Machinery, New York, 1990, str. 141–151.
- [23] G. B. Dantzig, *The diet problem*, *Interfaces* **20** (1990), č. 4, str. 43–47.
- [24] G. B. Dantzig, *Linear programming*, v „The New Palgrave Dictionary of Economics“, M. Palgrave (editor), Palgrave Macmillan, Londýn, 2008.
- [25] G. B. Dantzig, A. Orden, P. S. Wolfe, *The generalized simplex method for minimizing a linear form under linear inequality restraints*, *Pacific J. Math.* **5** (1955), str. 183–195.
- [26] P. Davis, R. Hersh, *The Mathematical Experience*, Houghton Mifflin, Boston, 1981.
- [27] D. A. d’Esopo, *A convex programming procedure*, *Naval Res. Logist. Quart.* **6** (1959), str. 33–42.
- [28] J. Dongarra, F. Sullivan, *Guest Editors’ Introduction: The Top 10 Algorithms*, *Computing in Science and Engg.*, Vol. 2 (2000), str. 22–23.
- [29] R. Dorfman, *Mathematical, or “linear,” programming: A nonmathematical exposition*, *The American Economic Review* **43** (1953), č. 5, str. 797–825.
- [30] O. Došlý, *Základy konvexní analýzy a optimalizace v \mathbb{R}^n* , Masarykova univerzita, 2005.
- [31] O. Došlý, P. Zemánek, *Integrální počet v \mathbb{R}* , Masarykova univerzita, 2011.
- [32] W. Dunham, *Journey through Genius: The Great Theorems of Mathematics*, John Wiley and Sons, New York, 1990.
- [33] F. Dyson, *Prof. Hermann Weyl, For.Mem.R.S.*, *Nature* **177** (1956), 457–458.
- [34] L. Euler, *De linea brevissima in superficie quacunq̄ue duo quaelibet puncta jungente* [O nejkratší čáře spojující dva libovolné body jakékoli plochy (On the shortest line joining two points on a surface)], *Comm. Ac. Petrop.* **3** (1732), str. 110–124. Publikováno také v *Opera Omnia: Series 1, Volume 25*, Orell Fussli, Turici, 1952. Dostupné online na adrese <https://odkaz.page.link/euler7>.

- [35] L. Euler, *Methodus inveniendi lineas curvas maximi minimive proprietate gaudentes, sive solutio problematis isoperimetrici lattissimo sensu accepti* [Metoda nalézání křivek majících vlastnost maxima nebo minima neboli řešení isoperimetrického problému pojatého v nejširším smyslu (A Method for Finding Curved Lines Enjoying Properties of Maximum or Minimum, or Solution of Isoperimetric Problems in the Broadest Accepted Sense)], Apud Marcum-Michaelem Bousquet & Socios, Lausannae & Genevae, 1744. Publikováno také v Opera Omnia: Series 1, Volume 24, Orell Fussli, Turici, 1952. Dostupné online na adrese <https://odkaz.page.link/euler2>.
- [36] L. Euler, *Elementa calculi variationum* [Základy variačního počtu (The Elements of the Calculus of Variations)], N. Comm. Ac. Petrop. **10** (1766), str. 51–93. Publikováno také v Opera Omnia: Series 1, Volume 25, Orell Fussli, Turici, 1952. Dostupné online na adrese <https://odkaz.page.link/euler8>.
- [37] M. W. Fenchel, *Convex Cones, Sets and Functions*, Princeton University, Princeton, 1953. Dostupné online na adrese <https://odkaz.page.link/fenchel1>. „Modernější“ verze tohoto textu je k dispozici online na adrese <https://odkaz.page.link/fenchel2>.
- [38] R. W. Farebrother, *A linear programming procedure based on de la Vallée Poussin's minimax estimation procedure*, Comput. Statist. Data Anal. **51** (2006), č. 2, str. 453–456.
- [39] J. B. Fourier, [referováno v] *Analyse des travaux de l'Academie Royale des Sciences, pendant l'annee 1823*, Partie mathématique, Histoire de l'Academie Royale des Sciences de l'Institut de France **6** [1823] (1826), str. xxix–xli. Částečně přetištěno jako *Premier extrait* v „Oeuvres de Fourier II“, G. Darboux (editor), Gauthier–Villars, Paříž, 1890, str. 321–324. Dostupné online na adrese <https://odkaz.page.link/fourier1>.
- [40] M. Frank, P. S. Wolfe, *An algorithm for quadratic programming*, Naval Res. Logist. Quart. **3** (1956), č. 1-2, str. 95–110.
- [41] C. Fraser, *J. L. Lagrange's changing approach to the foundations of the calculus of variations*, Arch. Hist. Exact Sci. **32** (1985), č. 2, str. 151–191.
- [42] G. Galilei, *Il Saggiatore*, Giacomo Mascardi, Rome, 1623.
- [43] S. I. Gass, A. A. Assad, *An Annotated Timeline of Operations Research: An Informal History*, International Series in Operations Research & Management Science **75**, Kluwer Academic Publishers, Boston, 2005. ISBN 1-4020-8112-X.
- [44] Ch. C. Gillispie (editor), *Dictionary of Scientific Biography* 14, Charles Scribner's Sons, New York, 1976.
- [45] M. Hamala, *Nelineárne programovanie*, druhé vydání, Alfa, Bratislava, 1976.
- [46] G. Hardy, *A Mathematician's Apology*, Cambridge University Press, Cambridge, 2012.
- [47] R. Heinlein, *Time Enough For Love*, ACE Books, New York, 1973.
- [48] D. Hilbert, *Mathematische Probleme – ortrag, gehalten auf dem internationalen Mathematiker-Kongreß zu Paris 1900*, Nachrichten von der Königl. Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse, Band Heft 3 (1900), s. 253–297.
- [49] C. G. Hildreth, *A quadratic programming procedure*, Naval Res. Logist. Quart. **4** (1957), str. 79–85.

- [50] F. L. Hitchcock, *The distribution of a product from several sources to numerous localities*, J. Math. Phys. Mass. Inst. Tech. **20** (1941), str. 224–230.
- [51] A. Chakraborty, V. Chandru, M. R. Rao, *A linear programming primer: from Fourier to Kar-markar*, Ann. Oper. Res. **287** (2020), č. 2, str. 593–616.
- [52] L. V. Kantorovitch, *On the translocation of masses*, Management Sci. **5** (1958), str. 1–4. Přeloženo z ruského originálu: C. R. (Doklady) Acad. Sci. URSS (N.S.) **37** (1942), str. 199–201.
- [53] L. V. Kantorovich, *The Best Use of Economic Resources*, Pergamon Press, 1965. Dostupné online na adrese <https://odkaz.page.link/kantorovich1>.
- [54] L. V. Kantorovich, *Mathematical methods of organizing and planning production*, Management Sci. **6** (1960), č. 4, str. 366–422.
- [55] H. M. Markowitz, *Portfolio selection*, J. Finance **7** (1952), č. 1, str. 77–91.
- [56] H. M. Markowitz, *The utility of wealth*, The Journal of Political Economy **60** (1952), č. 2, str. 151–158.
- [57] H. M. Markowitz, *The optimization of a quadratic function subject to linear constraints*, Naval Res. Logist. Quart. **3** (1956), č. 1-2, str. 111–133.
- [58] H. M. Markowitz, *The elimination form of the inverse and its application to linear programming*, Management Sci. **3** (1957), č. 3, str. 255–269.
- [59] H. M. Markowitz, *Portfolio Selection: Efficient Diversification of Investments*, Cowles Foundation for Research in Economics at Yale University, John Wiley & Sons, New York, 1959.
- [60] H. M. Markowitz, B. Hausner, H. W. Karr, *SIMSCRIPT: A Simulation Programming Language*, Research Memoranda **RM-3310-PR**, RAND Corporation, Santa Monica, 1962.
- [61] F. H. Murphy, V. Panchanadam, *Understanding linear programming modeling through an examination of the early papers on model formulation*, Oper. Res. **45** (1997), č. 3, str. 341–356.
- [62] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, New Jersey, 1970.
- [63] R. T. Rockafellar, *Lagrange multipliers and optimality*, SIAM Review **35** (1993), č. 2, str. 183–238.
- [64] N. Rose, *Mathematical Maxims and Minims*, Rome Press, Raleigh, 1988.
- [65] F. J. Scott, *The Correspondence of Isaac Newton IV (1694–1709)*, The Syndics of the Cambridge University Press, Cambridge, 1967.
- [66] S. M. Sinha, *Mathematical Programming – Theory and Methods*, Elsevier, Delhi, 2006.
- [67] D. E. Smith, *A Source Book in Mathematics II*, Dover Publications, New York, 1959.
- [68] G. J. Stigler, *The cost of subsistence*, Journal of Farm Economics **27** (1945), č. 2, str. 303–314.
- [69] A. Sucharev, A. Timochov, V. Fedorov, *Kurs metodov optimizacii*, Nauka, Moskva, 1986.
- [70] W. Sun, Y.-X. Yuan, *Optimization Theory and Methods – Nonlinear Programming*, Springer, New York, 2006.

- [71] W. Thomson, *Popular Lectures and Addresses II*, Macmillan & Co., Londýn, 1894. Úryvek o isoperimetrických problémech na str. 571–592 je dostupný také online na adrese <https://odkaz.page.link/kelvin1>.
- [72] A. N. Tolstoï, *Metody nakhozheniya naimen'shego summovogo kilometrazha pri planirovanii perezozok v prostranstve* [rusky; Metody pro nalezení nejmenšího počtu najetých kilometrů při plánování nákladní přepravy v prostoru (Methods of finding the minimal total kilometrage in cargo-transportation planning in space)], v „Planirovanie Perezozok, Sbornik pervyi“ [rusky; Plánování dopravy, Díl I (Transportation Planning, Volume I)], Transpechat' NKPS [TransPress of the National Commissariat of Transportation], Moskva, 1930, str. 23–55.
- [73] A. N. Tolstoï, *Metody ustraneniya neratsional'nykh perezozok pri planirovanii* [rusky; Metody eliminace iracionální dopravy při plánování (Methods of removing irrational transportation in planning)], *Sotsialisticheskiï Transport* **9** (1939), str. 28–51. Také publikováno jako „pamflet“: *Metody ustraneniya neratsional'nykh perezozok pri sostavlenii operativnykh planov* [rusky; Metody eliminace iracionální dopravy při konstrukci operativních plánů (Methods of Removing Irrational Transportation in the Construction of Operational Plans)], Transzheldorizdat, Moskva, 1941.
- [74] C. J. de la Vallée Poussin, *Sur la méthode de l'approximation minimum*, *Ann. Soc. Sci. Bruxelles* **35** (1911), str. 1–16.
- [75] S. Vojří, *Leonid Vitalievič Kantorovich*, zveřejněno online na adrese <https://odkaz.page.link/kantorovich2>.
- [76] S. Vojří, *Tjalling Charles Koopmans*, zveřejněno online na adrese <https://odkaz.page.link/koopmans1>.
- [77] R. S. Westfall, *Never at Rest: A Biography of Isaac Newton*, Cambridge University Press, Cambridge, 2010. ISBN 978-0-521-23143-5.
- [78] A. N. Whitehead, *Science and the Modern World*, The New American Library of World Literature, New York, 1948.
- [79] H. P. Williams, *Fourier's method of linear programming and its dual*, *Amer. Math. Monthly* **93** (1986), č. 9, str. 681–695.
- [80] P. S. Wolfe, *The simplex method for quadratic programming*, *Econometrica* **27** (1959), str. 382–398.
- [81] G. M. Ziegler, *Who solved the Hirsch conjecture?*, *Doc. Math.* (2012), Extra vol.: Optimization stories, str. 75–85.
- [82] <http://mathshistory.st-andrews.ac.uk/>

