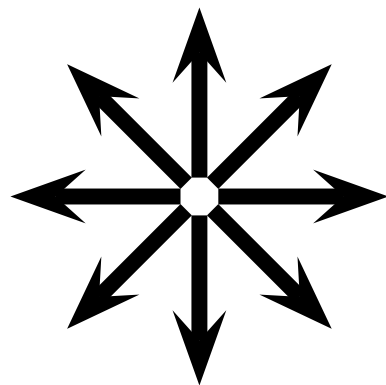


Milan Hladík

Vícekriteriální Optimalizace

text k přednášce

25. dubna 2024



Tento text zatím slouží jako pomocný text přednášky Vícekriteriální optimalizace na studiu informatiky MFF UK v Praze.

Autor čerpal především z knih Grygarová [1996]; Ehrgott [2005]. Další klasické knihy o vícekriteriálním lineárním programování jsou například The Luc [2016]; Steuer [1986]. Pro vícekriteriální nelineární programování viz Jahn [2011]; The Luc [1989]; Miettinen [1999]; Steuer [1986] nebo přehledový článek Marler and Arora [2004].

Případné připomínky a chyby zasílejte prosím na adresu hladik@kam.mff.cuni.cz.

Obsah

Obsah	5
1 Eficientní řešení	7
1.1 Úvod	7
1.2 Vícekriteriální konvexní programování	11
1.3 Vícekriteriální lineární programování	15
2 Přístupy k řešení	21
2.1 Metody typu I.	21
2.1.1 Metoda globální cílové funkce	21
2.1.2 Bensonova metoda	24
2.2 Metody typu II.	25
2.2.1 Vážený součet	25
2.2.2 Metoda ε -omezení	26
2.2.3 Lexikografická metoda	27
2.2.4 Goal attainment	29
2.2.5 Cílové programování	29
2.3 Metody typu III.	30
2.3.1 Metoda Geoffrion–Dyer–Feinberg	30
2.3.2 Metoda STEM	31
2.3.3 Algoritmy dialogu	32
2.4 Metody typu IV.	32
2.4.1 Normal boundary intersection aneb jak najít vzorek	32
2.4.2 Vnitřní a vnější aproximace	34
3 Vícekriteriální kombinatorické programování	37
3.1 Nejkratší cesta s více kritérii	37
3.2 Minimální kostra s více kritérii	39
3.3 Problém batohu s více kritérii	41
4 Další témata	43
4.1 Kuželová dominance	43
4.2 Užitková funkce	45
4.3 Intervalové vícekriteriální programování	45
4.3.1 Potenciálně eficientní řešení	45
4.3.2 Nutně eficientní řešení	47
5 Vícekriteriální rozhodování	51
5.1 DEA	51
5.2 AHP	53
6 Závěr	57
Značení	59

Kapitola 1

Eficientní řešení

1.1 Úvod

Úlohou vícekriteriálního programování rozumíme optimalizační úlohy, kde máme několik účelových funkcí. Ještě než začneme problém formálně rozebírat, ukážeme několik příkladů.

Příklad 1.1 (Příklady vícekriteriálního programování).

- *Vyhledávání dopravního spojení (operační výzkum)*. Zde kritériem může být minimální přepravní čas, cena přepravy, počet přestupů, maximalizace pohodlí (ne moc krátké a dlouhé přestupy).
- *Rozvrhování posádek letů (operační výzkum)*. Kritériem je minimální cena a maximální robustnost (odolnost proti zpoždění, např. delšími rozestupy).
- *Dynamické vyvažování zátěže ve virtualizačních klastrech (softwarové inženýrství, Grafnetter [2011])*. Chceme vybalancovat zatížení virtuálních mašin fyzickým strojům, kritérii jsou využitost CPU, paměti, prostoru, ...
- *Design elektrických obvodů počítačových čipů [Ben-Tal and Nemirovski, 2001, sekce 4.7]*. Kritéria zde jsou maximalizace rychlosti, minimalizace povrchu a uvolněného tepla.
- *Barva kůže (příroda)*. Světlá kůže absorbuje více vitamínu D, naopak tmavá lépe chrání lidskou pokožku a důležité sloučeniny, jako například foláty. Tato dvě kritéria jdou přímo proti sobě, jak to příroda vyřešila? \square

Příklad 1.2 (Optimální tvar cesty). Nechť tvar terénu je popsán křivkou $c(t)$, kde $t \in [0, T]$; pro jednoduchost předpokládáme jednorozměrný případ. Cílem je vyprojektovat tvar cesty popsany funkcí $x(t)$. Kritéria jsou:

- minimalizace odvezené či přivezené půdy

$$\min \int_0^T |x(t) - c(t)| dt,$$

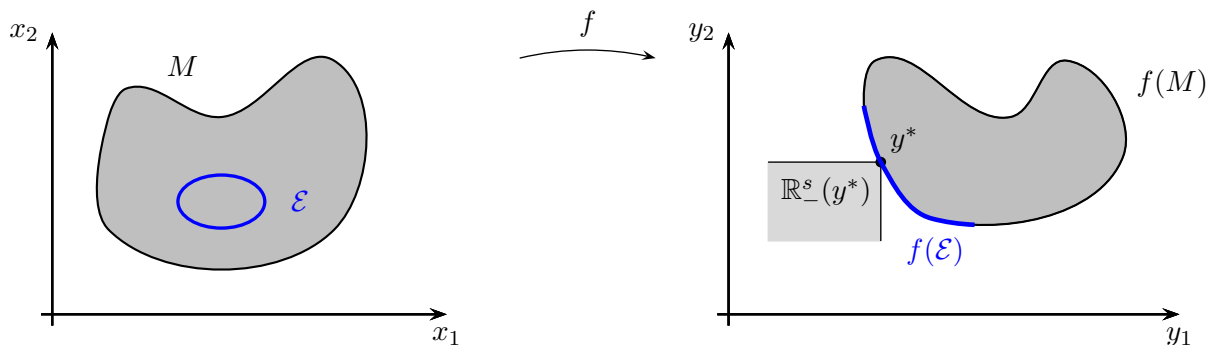
- ekologické hledisko minimalizace nejhorší změny terénu:

$$\min \max_{t \in [0, T]} |x(t) - c(t)|,$$

Omezení budou zahrnovat mj. podmínky na derivace

$$\begin{aligned} |x'(t)| &\leq \beta_1 \quad \dots \quad \text{přiměřený sklon,} \\ |x''(t)| &\leq \beta_2 \quad \dots \quad \text{přiměřená změna sklonu.} \end{aligned}$$

V praxi se takovéto úlohy typicky řeší diskretizací intervalu $[0, T]$, aproximací derivací diferencemi a redukcí na konečněrozměrný případ. \square



Obrázek 1.1: (Poznámka 1.4) Vlevo množina eficientních řešení \mathcal{E} uvnitř M a napravo jejich obraz. Paretoovská hranice $f(\mathcal{E})$ obsahuje ty body, které jsou nedominované ve směru kužele $\mathbb{R}_-^s(y^*)$.

Formulace. Přesněji, úlohou vícekriteriálního programování je

$$\min f(x) \quad \text{za podm. } x \in M, \quad (1.1)$$

kde $M \subseteq \mathbb{R}^n$ je množina přípustných řešení a $f(x) = (f_1(x), \dots, f_s(x)) : \mathbb{R}^n \rightarrow \mathbb{R}^s$ je vektorová funkce.

Tato úloha není z formálního hlediska dobře definovaná, protože na prostoru \mathbb{R}^s nemáme úplné uspořádání. Je tedy potřeba říci, co rozumíme řešením vícekriteriální úlohy.

Ideální řešení je takové řešení $x^* \in M$, pro které platí $f_i(x^*) = \min_M f_i(x)$ pro všechna $i = 1, \dots, s$. Je zřejmé, že ideální řešení prakticky nikdy neexistuje, a proto je potřeba zavést jiný koncept řešení.

Pro vektory $a, b \in \mathbb{R}^s$ relací $a \not\leq b$ rozumíme $a \leq b$ a zároveň $a \neq b$. Jinými slovy, $a \leq b$ a pro aspoň jeden index i platí $a_i < b_i$.¹⁾

Definice 1.3 (Eficientní řešení). Přípustné řešení $x^* \in M$ je *eficientní*, pokud neexistuje $x \in M$ takové, že $f(x) \not\leq f(x^*)$. Množinu všech eficientních řešení značíme \mathcal{E} .

Eficientní řešení je také nazývá *nedominované*, nebo *Pareto-optimální*²⁾ podle italského ekonoma Vilfreda Pareta,³⁾ který se však ještě nezabýval vícekriteriálním programováním v pravém slova smyslu.

Poznámka 1.4 (Geometrická interpretace eficientního řešení). Označme

$$f(M) := \{y \in \mathbb{R}^s; y = f(x), x \in M\}$$

obraz množiny M a

$$\mathbb{R}_-^s(y^*) = \{y \in \mathbb{R}^s; y \leq y^*\}$$

kužel bodů, které dominují bodu y^* . Potom obraz množiny eficientních řešení $f(\mathcal{E})$ představuje část hranice $f(M)$ vlevo dole; viz obrázek 1.1. Jsou to ty body, které nejsou dominovány žádným jiným. Není těžké nahlédnout, že tyto body skutečně leží na hranici $f(M)$, a proto se množina $f(\mathcal{E})$ také nazývá *Paretoovská hranice*.

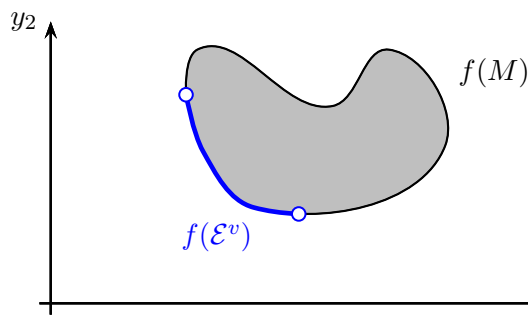
Pozorování 1.5. Množina $f(\mathcal{E})$ leží na hranici množiny $f(M)$.

Jako důsledek úvahy z předchozí poznámky dostáváme ihned

¹⁾Toto značení zavedl T.C. Koopmans roku 1951.

²⁾Tento pojem se používá i v teorii her. Strategie hráčů je Pareto-optimální, pokud neexistuje jiná strategie hráčů, která by oproti původní strategii nikomu nepohoršila a aspoň jednomu hráči polepšila.

³⁾Vilfredo Pareto (1848-1923) přišel s konceptem eficientního řešení roku 1896. Tento ekonom je také znám svým pravidlem 80/20, které se objevuje empiricky v různých kontextech: 80% země je v držení 20% obyvatel, 80% světového příjmu je kontrolováno 20% nejbohatšími, 80% času stráveného psaním diplomové práce se týká 20% textu, 80% zdravotní péče je využíváno 20% pacientů, atp. Toto pravidlo se ale nesmí přeceňovat, například 20% nejúspěšnějších knižních titulů vydělává přibližně 96% příjmu vydavatelství.

Obrázek 1.2: Paretoovská hranice vlastních eficientních řešení $f(\mathcal{E}^v)$.

Pozorování 1.6. Platí $y^* \in f(\mathcal{E})$ právě tehdy, když $\mathbb{R}_-^s(y^*) \cap f(M) = \{y^*\}$.

Jak otestovat, zda dané řešení je eficientní? Než odvodíme jiné postupy (viz sekce 2.1.2), ukážeme si jednoduchou metodu.

Věta 1.7. Platí $x^* \in \mathcal{E}$ právě tehdy, když pro každé $i = 1, \dots, s$ je

$$f_i(x^*) = \min_{x \in M, f_k(x) \leq f_k(x^*), k \neq i} f_i(x) \quad \text{za podm. } x \in M, f_k(x) \leq f_k(x^*), k \neq i. \quad (1.2)$$

Důkaz. „ \Rightarrow “ Bud' $x^* \in \mathcal{E}$ a necht' existuje i takové, že x^* není optimem úlohy (1.2). Tedy existuje $x \in M$ takové že $f_k(x) \leq f_k(x^*)$, $k \neq i$, a $f_i(x) < f_i(x^*)$. To je spor s $x^* \in \mathcal{E}$.

„ \Leftarrow “ Pro spor necht' $x^* \notin \mathcal{E}$. Tudiž existuje $x \in M$ takové, že $f(x) \not\leq f(x^*)$. To znamená $f_k(x) \leq f_k(x^*)$ a pro určité i je $f_i(x) < f_i(x^*)$. Proto x^* není optimem úlohy (1.2). \square

Existuje celá řada různých typů eficientního řešení. Kromě základního typu se budeme zabývat ještě jedním speciálnějším typem, tzv. vlastním eficientním řešením [Geoffrion, 1968].

Definice 1.8 (Vlastní eficientní řešení). Přípustné řešení $x^* \in M$ je *vlastní eficientní* pokud existuje $\beta \geq 0$ takové, že pro každé $x \in M$ a každé $i \in \{1, \dots, s\}$, pro které $f_i(x) < f_i(x^*)$ existuje k takové, že $f_k(x) > f_k(x^*)$ a navíc $f_i(x^*) - f_i(x) \leq \beta(f_k(x) - f_k(x^*))$.

Množinu všech vlastních eficientních řešení budeme značit \mathcal{E}^v . Obrázek 1.2 ilustruje její tvar v prostoru obrazů. Evidentně $\mathcal{E}^v \subseteq \mathcal{E}$.

Skalarizace

Označme jako

$$S := \{\lambda \in \mathbb{R}^s; \lambda > 0\},$$

$$\bar{S} := \{\lambda \in \mathbb{R}^s; \lambda \geq 0\}$$

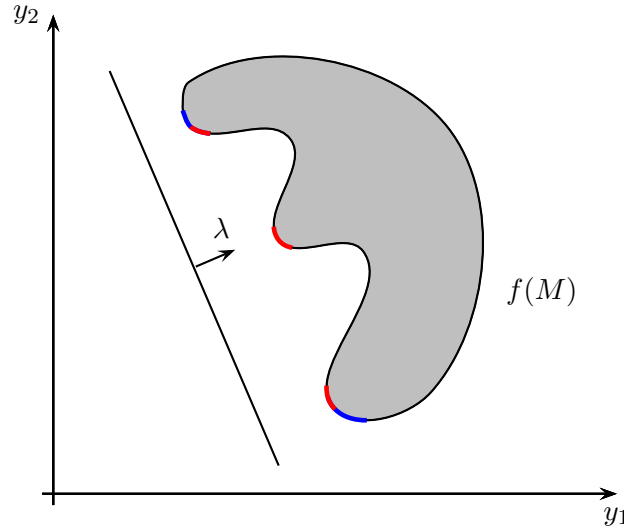
množinu všech kladných resp. nezáporných nenulových váhových vektorů. Nyní *skalarizací* úlohy (1.1) rozumíme redukci na jedno-kriteriální úlohu. Ponejvíce se budeme zabývat skalarizací

$$\min \lambda^T f(x) \quad \text{za podm. } x \in M. \quad (1.3)$$

Množinu optimálních řešení této úlohy budeme značit $M_{opt}(\lambda)$.

Důležitým výsledkem jest, že jakékoli optimum této skalarizace s kladnými vahami dává vždy eficientní řešení. Podotýkáme, že nepotřebujeme žádné předpoklady na tvar účelových funkcí a množiny přípustných řešení (spojitost, hladkost, konvexita, ...)

Věta 1.9. Platí $\cup_{\lambda \in S} M_{opt}(\lambda) \subseteq \mathcal{E}$.



Obrázek 1.3: (Poznámka 1.11) Modře obraz $\cup_{\lambda \in S} M_{opt}(\lambda)$, červeně zbytek Paretovské hranice $f(\mathcal{E})$.

Důkaz. Buďte $\lambda \in S$ a $x^* \in M_{opt}(\lambda)$ libovolné pevné. Pro spor předpokládejme, že $x^* \notin \mathcal{E}$, tedy existuje $x \in M$ splňující $f(x) \not\leq f(x^*)$. Přednásobením nerovnosti vektorem λ^T dostaneme $\lambda^T f(x) < \lambda^T f(x^*)$, což je spor s $x^* \in M_{opt}(\lambda)$. \square

Překvapivě lze zesílit toto tvrzení na vlastní eficienci.

Věta 1.10. Platí $\cup_{\lambda \in S} M_{opt}(\lambda) \subseteq \mathcal{E}^v$.

Důkaz. Buďte $\lambda \in S$ a $x^* \in M_{opt}(\lambda)$ libovolné pevné. Pro spor předpokládejme, že $x^* \notin \mathcal{E}^v$.

Označme $\beta := (s-1) \max_{i,j} \frac{\lambda_i}{\lambda_j}$. Musí tedy existovat $x \in M$ a $i \in \{1, \dots, s\}$ takové, že $f_i(x) < f_i(x^*)$ a zároveň $f_i(x^*) - f_i(x) > \beta(f_k(x) - f_k(x^*))$ pro všechna $k \neq i$. Z toho plyne

$$f_i(x^*) - f_i(x) > (s-1) \frac{\lambda_k}{\lambda_i} (f_k(x) - f_k(x^*)), \quad \forall k \neq i.$$

Přenásobením λ_i a sečtením přes všechna $k \neq i$ dostaneme

$$\sum_{k \neq i} \lambda_i (f_i(x^*) - f_i(x)) > (s-1) \sum_{k \neq i} \lambda_k (f_k(x) - f_k(x^*)),$$

neboli

$$\lambda_i (f_i(x^*) - f_i(x)) > \sum_{k \neq i} \lambda_k (f_k(x) - f_k(x^*)).$$

Přeuspořádáním sčítanců máme

$$\sum_{k=1}^s \lambda_k f_k(x^*) > \sum_{k=1}^s \lambda_k f_k(x).$$

Tedy $\lambda^T f(x^*) > \lambda^T f(x)$, což je spor s $x^* \in M_{opt}(\lambda)$. \square

Poznámka 1.11 (Geometrická interpretace skalarizace). Úlohu (1.3) lze vyjádřit jako

$$\min \lambda^T y \quad \text{za podm. } y \in f(M),$$

čili minimalizujeme lineární funkci (s kladnou normálou) na množině obrazů $f(M)$. To ilustruje, proč Paretovská hranice vypadá jak vypadá a proč obecně takto nezrekonstruujeme celou Paretovskou hranici (pokud je $f(M)$ velmi nekonvexní); viz obrázek 1.3.

Shrnutí. Ukázali jsme, že pro každou úlohu jest

$$\cup_{\lambda \in S} M_{opt}(\lambda) \subseteq \mathcal{E}^v \subseteq \mathcal{E}. \quad (1.4)$$

Pro konvexní úlohu dokážeme říci více, což bude náplní následující sekce.

1.2 Vícekriteriální konvexní programování

Úlohou *vícekriteriálního konvexního programování* rozumíme úlohu (1.1), tedy

$$\min f(x) \text{ za podm. } x \in M,$$

kde M je konvexní množina a $f_1, \dots, f_s: \mathbb{R}^n \rightarrow \mathbb{R}$ konvexní funkce.

Pro konvexní úlohu platí těsnější vztah v (1.4). Konkrétně, každé eficientní řešení je optimem nějaké skalarizace s nezápornými vahami. K důkazu budeme potřebovat větu o oddělitelnosti konvexních množin, kterou připomínáme v jednodušším znění.

Věta 1.12 (O oddělitelnosti konvexních množin). *Budte $\mathcal{A}, \mathcal{B} \subset \mathbb{R}^n$ konvexní množiny. Je-li $\text{int } \mathcal{A} \cap \text{int } \mathcal{B} = \emptyset$, pak jsou množiny oddělitelné nadrovinou ve tvaru $\lambda^T x = \mu$, $\lambda \neq 0$, tedy*

$$\begin{aligned} \lambda^T x &\leq \mu \quad \forall x \in \mathcal{A}, \\ \lambda^T x &\geq \mu \quad \forall x \in \mathcal{B}. \end{aligned}$$

Věta 1.13. *Platí $\mathcal{E} \subseteq \cup_{\lambda \in \bar{S}} M_{opt}(\lambda)$.*

Důkaz. Bud' $x^* \in \mathcal{E}$ a $y^* := f(x^*)$ jeho obraz. Dále označ

$$K := \{y \in \mathbb{R}^s; y \geq f(x) \text{ pro nějaké } x \in M\}.$$

Tato množina je konvexní ($f(M)$ konvexní být nemusí, proto pracujeme s touto), neboť pro každé $\alpha_1, \alpha_2 \geq 0$, $\alpha_1 + \alpha_2 = 1$, a pro každé $y^1, y^2 \in K$ existuje $x^1, x^2 \in M$ takové, že $f(x^1) \leq y^1$, $f(x^2) \leq y^2$. Váženým součtem těchto dvou nerovností dostaneme

$$\alpha_1 f(x^1) + \alpha_2 f(x^2) \leq \alpha_1 y^1 + \alpha_2 y^2.$$

Z konvexity f pak navíc

$$f(\alpha_1 x^1 + \alpha_2 x^2) \leq \alpha_1 f(x^1) + \alpha_2 f(x^2) \leq \alpha_1 y^1 + \alpha_2 y^2.$$

Nyní ukážeme, že $\mathbb{R}_-^s(y^*) \cap K = \{y^*\}$, viz obrázek 1.4. Zřejmě $y^* \in \mathbb{R}_-^s(y^*) \cap K$. Pro spor předpokládejme, že existuje $y \neq y^*$ takové, že $y \in \mathbb{R}_-^s(y^*) \cap K$. Protože $y \in K$, existuje $x \in M$ takové, že $f(x) \leq y$. Protože $y \in \mathbb{R}_-^s(y^*)$, tak $f(x) \leq y \not\leq y^* = f(x^*)$. To je spor s eficientností x^* .

Máme tedy dvě konvexní množiny $\mathbb{R}_-^s(y^*)$ a K s prázdným průnikem vnitřků. Podle věty 1.12 o oddělitelnosti konvexních množin existuje oddělující nadrovina, navíc prochází bodem y^* . Nadrovina má tedy popis $\lambda^T y = \lambda^T y^*$, kde $\lambda \neq 0$ je její normála. Bez újmy na obecnost nechť $\mathbb{R}_-^s(y^*)$ leží v záporném a K v kladném poloprostoru, tedy

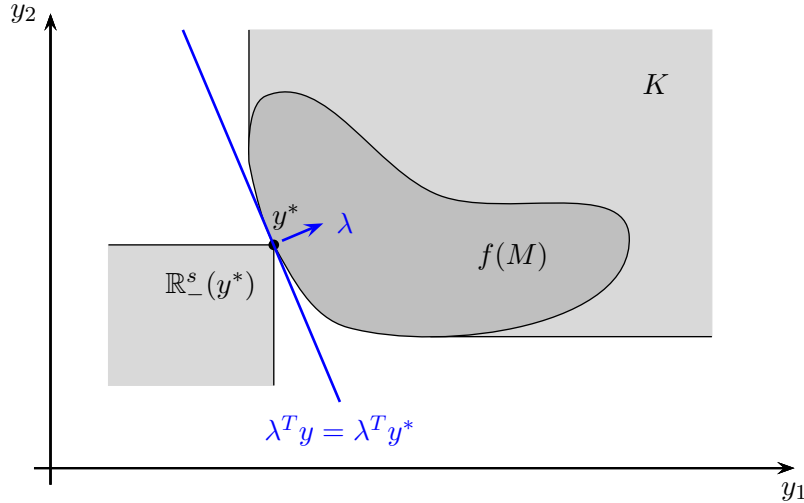
$$\lambda^T y \leq \lambda^T y^* \quad \text{pro všechna } y \in \mathbb{R}_-^s(y^*), \quad (1.5)$$

$$\lambda^T y \geq \lambda^T y^* \quad \text{pro všechna } y \in K. \quad (1.6)$$

Z první vlastnosti (1.5) vyplývá, že $\lambda \not\geq 0$. To snadno nahlédneme dosazením vektorů $y := y^* - e_i \in \mathbb{R}_-^s(y^*)$, kde e_i je i -tý jednotkový vektor, do nerovnosti (1.5) postupně pro $i = 1, \dots, s$. Dostaneme $\lambda^T(y^* - e_i) \leq \lambda^T y^*$, z čehož $\lambda^T e_i \geq 0$, neboli $\lambda_i \geq 0$.

Z druhé vlastnosti (1.6) vyplyne optimalita. Pro každé $x \in M$ je $f(x) \in K$ a podle (1.6) je $\lambda^T f(x) \geq \lambda^T f(x^*)$. Tudíž $x^* \in M_{opt}(\lambda)$. \square

Inkluze z věty 1.13 obecně jako rovnost neplatí (najděte protipříklad). Za určitých předpokladů na jednoznačnost optimálního řešení skalarizace ale rovnost nastává.



Obrázek 1.4: (důkaz věty 1.13) Oddělující nadrovina $\lambda^T y = \lambda^T y^*$ množin $\mathbb{R}_-^s(y^*)$ a K .

Věta 1.14. *Bud' $\lambda \in \bar{S}$ a necht' platí $M_{opt}(\lambda) = \{x^*\}$. Pak $x^* \in \mathcal{E}$.*

Důkaz. Sporem necht' $x^* \notin \mathcal{E}$, tedy existuje $x \in M$ takové, že $f(x) \not\leq f(x^*)$. Přenásobením λ^T dostaneme $\lambda^T f(x) \leq \lambda^T f(x^*)$, což je spor s tím, že x^* je jednoznačné optimum skalarizace. \square

Jednoznačné optimum se nabývá například pro optimalizační úlohy s účelovou funkcí, která je ryze konvexní. *Ryze konvexní* je funkce $\varphi: \mathbb{R}^n \rightarrow \mathbb{R}$, která splňuje podmínku

$$\varphi(\alpha_1 x_1 + \alpha_2 x_2) < \alpha_1 \varphi(x_1) + \alpha_2 \varphi(x_2)$$

pro všechny konvexní kombinace s $x_1 \neq x_2$ a $\alpha_1, \alpha_2 > 0$, $\alpha_1 + \alpha_2 = 1$. Poznamenejme, že lineární funkce není ryze konvexní, ale například funkce e^x nebo x^2 ryze konvexní jsou.

Důsledek 1.15. *Necht' f_1, \dots, f_s jsou ryze konvexní. Pak $\mathcal{E} = \cup_{\lambda \in \bar{S}} M_{opt}(\lambda)$.*

Ve větě 1.13 jsme ukázali, že každé eficientní řešení je optimem nějaké skalarizace s nezápornými vahami. Nyní odvodíme podobný výsledek pro vlastní eficientci: každé vlastní eficientní řešení je optimem nějaké skalarizace s kladnými vahami.

Věta 1.16 (Geoffrion [1968]). *Platí $\mathcal{E}^v = \cup_{\lambda \in S} M_{opt}(\lambda)$.*

Důkaz. Inkluzi „ \supseteq “ máme z věty 1.10, takže se zaměříme pouze na tu druhou. Bud' $x^* \in \mathcal{E}^v$. Tedy existuje $\beta \geq 0$ takové, že pro každé $x \in M$ a každé $i \in \{1, \dots, s\}$, pro které $f_i(x) < f_i(x^*)$ existuje k takové, že $f_k(x) > f_k(x^*)$ a navíc $f_i(x^*) - f_i(x) \leq \beta(f_k(x) - f_k(x^*))$, neboli

$$f_i(x^*) + \beta f_k(x^*) \leq f_i(x) + \beta f_k(x). \quad (1.7)$$

Bud' $i \in \{1, \dots, s\}$ libovolné pevné a definujme vektorovou funkci

$$F^i(x) := \begin{pmatrix} f_i(x) + \beta f_1(x) \\ \vdots \\ f_i(x) + \beta f_{i-1}(x) \\ f_i(x) \\ f_i(x) + \beta f_{i+1}(x) \\ \vdots \\ f_i(x) + \beta f_s(x) \end{pmatrix}.$$

Platí, že pro žádné $x \in M$ není splněna nerovnost

$$F^i(x) < F^i(x^*). \quad (1.8)$$

Pokud by totiž platila, tak i -tá nerovnost $f_i(x) < f_i(x^*)$ implikuje podle vlastní eficeie vztah (1.7), který je v rozporu s k -tou nerovnicí v (1.7).

Definujeme $y^i := F^i(x^*)$ a množinu

$$K^i := \{y \in \mathbb{R}^s; y \geq F^i(x) \text{ pro nějaké } x \in M\}.$$

Tato množina je konvexní a pro použití věty o oddělitelnost potřebujeme ještě ukázat

$$\text{int } \mathbb{R}_-^s(y^i) \cap \text{int } K^i = \emptyset.$$

Pro spor předpokládejme, že existuje $y \in \text{int } \mathbb{R}_-^s(y^i) \cap \text{int } K^i$. Protože $y \in \text{int } K^i$, tak existuje $x \in M$ takové, že $F^i(x) \leq y$. Z vlastnosti $y \in \text{int } \mathbb{R}_-^s(y^i)$ zase vyplývá $y < y^i$. Dohromady máme $F^i(x) \leq y < y^i = F^i(x^*)$, což je spor s (1.8).

Množiny $\mathbb{R}_-^s(y^i)$ a K^i tedy můžeme oddělit nadrovinou, a ta má popis $(\lambda^i)^T y = (\lambda^i)^T y^i$, kde λ^i je její normála znormovaná $\sum_{j=1}^s |\lambda_j^i| = 1$. Bez újmy na obecnost necht' $\mathbb{R}_-^s(y^i)$ leží v záporném a K^i v kladném poloprostoru, tedy

$$\begin{aligned} (\lambda^i)^T y &\leq (\lambda^i)^T y^i && \text{pro všechna } y \in \mathbb{R}_-^s(y^i), \\ (\lambda^i)^T y &\geq (\lambda^i)^T y^i && \text{pro všechna } y \in K^i. \end{aligned}$$

Analogicky jako v důkazu věty 1.13 vyplývá z první vlastnosti, že $\lambda^i \in \bar{S}$. Z druhé vlastnosti pak plyne, že pro každé $x \in M$ je $(\lambda^i)^T F^i(x) \geq (\lambda^i)^T F^i(x^*)$, neboli

$$\sum_{j=1}^s \lambda_j^i f_j(x) + \sum_{j \neq i} \beta \lambda_j^i f_j(x) \geq \sum_{j=1}^s \lambda_j^i f_j(x^*) + \sum_{j \neq i} \beta \lambda_j^i f_j(x^*).$$

Protože $\sum_{j=1}^s \lambda_j^i = 1$, nerovnost se zjednoduší na

$$f_i(x) + \sum_{j \neq i} \beta \lambda_j^i f_j(x) \geq f_i(x^*) + \sum_{j \neq i} \beta \lambda_j^i f_j(x^*).$$

Sečtením všech těchto nerovnic pro $i = 1, \dots, s$ získáme

$$\sum_{i=1}^s f_i(x) + \sum_{i,j,j \neq i} \beta \lambda_j^i f_j(x) \geq \sum_{i=1}^s f_i(x^*) + \sum_{i,j,j \neq i} \beta \lambda_j^i f_j(x^*).$$

Přejmenováním indexů

$$\sum_{i=1}^s f_i(x) + \sum_{i,j,j \neq i} \beta \lambda_i^j f_i(x) \geq \sum_{i=1}^s f_i(x^*) + \sum_{i,j,j \neq i} \beta \lambda_i^j f_i(x^*)$$

a úpravou máme

$$\sum_{i=1}^s \left(1 + \sum_{j \neq i} \beta \lambda_i^j\right) f_i(x) \geq \sum_{i=1}^s \left(1 + \sum_{j \neq i} \beta \lambda_i^j\right) f_i(x^*).$$

Pokud označíme $\lambda_i^* := 1 + \sum_{j \neq i} \beta \lambda_i^j > 0$ pro $i = 1, \dots, s$, tak nerovnost má tvar $(\lambda^*)^T f(x) \geq (\lambda^*)^T f(x^*)$. To znamená, že $x^* \in M_{\text{opt}}(\lambda^*)$, kde $\lambda^* \in S$. \square

Shrnutí: Ukázali jsme, že pro každou konvexní úlohu jest

$$\cup_{\lambda \in S} M_{\text{opt}}(\lambda) = \mathcal{E}^v \subseteq \mathcal{E} \subseteq \cup_{\lambda \in \bar{S}} M_{\text{opt}}(\lambda). \quad (1.9)$$

Pro úlohu s ryze konvexními účelovými funkcemi navíc platí

$$\cup_{\lambda \in S} M_{\text{opt}}(\lambda) = \mathcal{E}^v \subseteq \mathcal{E} = \cup_{\lambda \in \bar{S}} M_{\text{opt}}(\lambda).$$

Podmínky optimality

Připomeňme nejprve obecné podmínky optimality Fritze Johna pro jednokriteriální úlohu

$$\min \varphi(x) \quad \text{za podm.} \quad g(x) \leq 0. \quad (1.10)$$

Zde $\varphi: \mathbb{R}^n \rightarrow \mathbb{R}$ a $g: \mathbb{R}^n \rightarrow \mathbb{R}^m$.

Věta 1.17 (Fritz John, 1948, nutné podmínky optimality). *Bud' x^* lokální minimum úlohy (1.10). Pak existuje $\mu \in \mathbb{R}$ a $\nu \in \mathbb{R}^m$ takové, že*

$$\begin{aligned} \mu &\geq 0, \quad \nu \geq 0, \quad (\mu, \nu) \neq 0, \\ \mu \nabla \varphi(x^*) + \nu^T \nabla g(x^*) &= 0, \\ \nu^T g(x^*) &= 0. \end{aligned}$$

Připomeňme, že podmínka $\nu^T g(x^*) = 0$ se nazývá podmínka komplementarity, protože ekvivalentně říká, že pro každé $i = 1, \dots, m$ je $\nu_i = 0$ nebo $g_i(x^*) = 0$.

Zobecnění podmínek na vícekritériální konvexní úlohu

$$\min f(x) \quad \text{za podm.} \quad g(x) \leq 0. \quad (1.12)$$

kde $f: \mathbb{R}^n \rightarrow \mathbb{R}^s$ a $g: \mathbb{R}^n \rightarrow \mathbb{R}^m$, je snadné pomocí věty 1.13.

Věta 1.18. *Bud' x^* eficientní řešení konvexní úlohy (1.12). Pak existuje $\lambda \in \mathbb{R}^s$ a $\nu \in \mathbb{R}^m$ takové, že*

$$\begin{aligned} \lambda &\geq 0, \quad \nu \geq 0, \quad (\lambda, \nu) \neq 0, \\ \lambda^T \nabla f(x^*) + \nu^T \nabla g(x^*) &= 0, \\ \nu^T g(x^*) &= 0. \end{aligned}$$

Důkaz. Z věty 1.13 víme, že eficientní řešení x^* je optimem úlohy

$$\min \mu^T f(x) \quad \text{za podm.} \quad g(x) \leq 0$$

pro určité $\mu \in \overline{S}$. Aplikací věty Fritze Johna dostaneme výsledné znění. \square

Karush-Kuhn-Tuckerovy (KKT) podmínky pro jednokriteriální úlohu (1.10) připomeneme v následující variantě se Slaterovou podmínkou (existence vnitřního bodu $x^0: g(x^0) < 0$). KKT podmínky ilustruje obrázek 1.5. Geometricky KKT podmínky říkají, že záporný gradient účelové funkce $-\nabla \varphi(x^*)$ leží v nezáporném kuželi, generovaném gradienty $\nabla g_i(x^*)$ aktivních podmínek (tj. těch i , pro které $g_i(x^*) = 0$). Pro neaktivní podmínky (tj. $g_i(x^*) < 0$) je $\nu_i = 0$, čímž dosáhneme splnění poslední z podmínek – podmínky komplementarity $\nu^T g(x^*) = 0$.

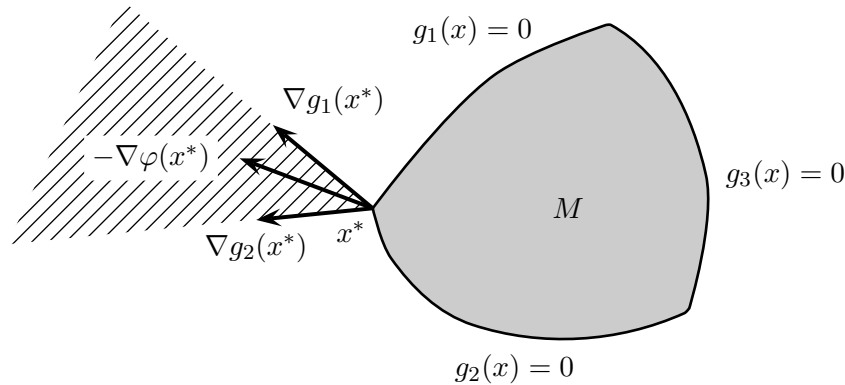
Věta 1.19 (KKT nutné podmínky optimality). *Bud' x^* lokální minimum úlohy (1.10). Necht' funkce $g_j(x)$, $j \in \{k; g_k(x^*) = 0\}$, jsou konvexní a necht' existuje vektor x^0 takový, že $g(x^0) < 0$. Pak existuje $\nu \in \mathbb{R}^m$ takové, že platí podmínky*

$$\begin{aligned} \nu &\geq 0, \\ \nabla \varphi(x^*) + \nu^T \nabla g(x^*) &= 0, \\ \nu^T g(x^*) &= 0. \end{aligned}$$

Zobecnění KKT podmínek na vícekritériální případ je opět přímočaré. Poznamenejme, že už v původním článku Kuhn and Tucker [1951] autoři v zásadě uvažují vícekritériální úlohu.

Věta 1.20. *Bud' x^* eficientní řešení konvexní úlohy (1.12) a necht' $g(x^0) < 0$ pro určitý vektor x^0 . Pak existuje $\lambda \in \mathbb{R}^s$ a $\nu \in \mathbb{R}^m$ takové, že*

$$\begin{aligned} \lambda &\in \overline{S}, \quad \nu \geq 0, \\ \lambda^T \nabla f(x^*) + \nu^T \nabla g(x^*) &= 0, \\ \nu^T g(x^*) &= 0. \end{aligned}$$



Obrázek 1.5: KKT podmínky: záporný gradient účelové funkce $-\nabla\varphi(x^*)$ leží v nezáporném kuželi, generovaném gradienty $\nabla g_1(x^*)$ a $\nabla g_2(x^*)$ aktivních podmínek.

Důkaz. Podle věty 1.13 eficientní řešení x^* je optimum úlohy

$$\min \lambda^T f(x) \quad \text{za podm. } g(x) \leq 0$$

pro určité $\lambda \in \bar{S}$. Zbytek plyne aplikací věty o KKT podmínkách. □

Postačující KKT podmínky mají následující podobu.

Věta 1.21 (KKT postačující podmínky optimality). *Uvažujme konvexní úlohu (1.10). Pokud pro daný přípustný bod x^* platí KKT podmínky*

$$\begin{aligned} \nu &\geq 0, \\ \nabla\varphi(x^*) + \nu^T \nabla g(x^*) &= 0, \\ \nu^T g(x^*) &= 0. \end{aligned}$$

pak x^ je optimální řešení.*

Zobecnění pro vícekriteriální úlohu:

Věta 1.22. *Uvažujme konvexní úlohu (1.12). Pokud pro daný přípustný bod x^* platí KKT podmínky*

$$\begin{aligned} \lambda \in S, \nu &\geq 0, \\ \lambda^T \nabla f(x^*) + \nu^T \nabla g(x^*) &= 0, \\ \nu^T g(x^*) &= 0. \end{aligned}$$

pak x^ je vlastní eficientní řešení úlohy (1.12).*

Důkaz. Podle věty 1.21 o KKT podmínkách je x^* optimálním řešením úlohy

$$\min \lambda^T f(x) \quad \text{za podm. } g(x) \leq 0,$$

a podle věty 1.10 je pak vlastním eficientním řešením úlohy (1.12). □

1.3 Vícekriteriální lineární programování

Úlohou *vícekriteriálního lineárního programování* rozumíme úlohu

$$\min Cx \quad \text{za podm. } x \in M, \tag{1.18}$$

kde $C \in \mathbb{R}^{s \times n}$. Množina přípustných řešení M je konvexní polyedr, popsán například jako

$$M = \{x \in \mathbb{R}^n; Ax = b, x \geq 0\},$$

kde $A \in \mathbb{R}^{m \times n}$ a $b \in \mathbb{R}^m$.

Nejprve ukážeme užitečnou charakterizaci eficientních řešení. Není těžké nahlédnout, že podobné tvrzení bude platit, i když účelovou funkci $e^T y$ v (1.19) změníme na $\lambda^T y$, kde $\lambda > 0$ je libovolné.

Věta 1.23 (Charnes & Cooper, 1961, Ecker & Kouada, 1975). *Buď $x^0 \in M$ a uvažujme úlohu lineárního programování*

$$\max e^T y \quad \text{za podm. } x \in M, Cx + y = Cx^0, y \geq 0. \quad (1.19)$$

Pak platí:

- (1) $x^0 \in \mathcal{E}$ právě tehdy, když úloha (1.19) má optimální hodnotu 0; optimální řešení je pak $(x^0, 0)$,
- (2) je-li (x^*, y^*) optimální řešení (1.19), pak $x^* \in \mathcal{E}$.

Důkaz.

- (1) „ \Rightarrow “ Pro spor necht existuje přípustné řešení (x^*, y^*) , splňující $e^T y^* > 0$. Pak z rovnosti $Cx^* + y^* = Cx^0$ plyne $Cx^* \not\leq Cx^0$, což je spor s eficiencí x^0 .
„ \Leftarrow “ Pro spor necht x^0 není eficientní, tj. existuje $x \in M$ takové, že $Cx \not\leq Cx^0$. Definujme-li $y := Cx^0 - Cx$, pak (x, y) je přípustné řešení úlohy (1.19) a navíc $e^T y > 0$. To je spor s předpokladem, že optimální hodnota je 0.
- (2) Pro spor necht x^* není eficientní, tj. existuje $x \in M$ takové, že $Cx \not\leq Cx^*$. Definujme-li $y := Cx^0 - Cx$, pak (x, y) je přípustné řešení úlohy (1.19) a platí $y = Cx^0 - Cx \not\leq Cx^0 - Cx^* = y^*$, tedy $e^T y > e^T y^*$. To je spor s optimalitou (x^*, y^*) . \square

V naší postupně budované charakterizaci eficientních řešení pomocí skalarizačních úloh postoupíme o další krok tím, že do ní zahrneme lineární případ.

Věta 1.24 (Isermann, 1974). *Platí $\cup_{\lambda \in S} M_{opt}(\lambda) = \mathcal{E}$.*

Důkaz. Inkluzi „ \subseteq “ máme z věty 1.9, proto se zaměříme na tu druhou. Buď $x^0 \in \mathcal{E}$, pak podle věty 1.23 má úloha (1.19), tj.

$$\max e^T y \quad \text{za podm. } Ax = b, Cx + y = Cx^0, x, y \geq 0.$$

optimální řešení $(x^0, 0)$. Uvažujme duální úlohu

$$\min b^T u + (Cx^0)^T v \quad \text{za podm. } A^T u + C^T v \geq 0, v \geq e.$$

Protože má primární úloha vždy optimální řešení, má duální úloha optimální řešení (u^*, v^*) a platí rovnost optimálních hodnot $b^T u^* + (Cx^0)^T v^* = 0$.

Zřejmě zůstane u^* optimálním řešením duální úlohy i když zafixujeme $v := v^*$ a úloha bude mít tvar

$$\min b^T u \quad \text{za podm. } A^T u \geq -C^T v^*.$$

Protože má tato úloha optimum, bude jej mít i k ní duální úloha

$$\max -(C^T v^*)^T x \quad \text{za podm. } Ax = b, x \geq 0.$$

Navíc optimální hodnoty se rovnají a mají hodnotu $b^T u^* = -(Cx^0)^T v^* = -(C^T v^*)^T x^0$. Tím pádem je x^0 přípustné řešení této úlohy, které nabývá optimální hodnoty, tedy je optimem. Proto $x^0 \in M_{opt}(\lambda)$ pro $\lambda = v^* \geq e$. \square

Pro úlohy vícekriteriálního lineárního programování můžeme tedy řetízek inkluzí (1.9) ještě utáhnout a dostáváme vztah

$$\cup_{\lambda \in S} M_{opt}(\lambda) = \mathcal{E}^v = \mathcal{E} \subseteq \cup_{\lambda \in \bar{S}} M_{opt}(\lambda).$$

Inkluze napravo se obecně jako rovnost nabývat nemusí.

Příklad 1.25. Uvažujme dvoukriteriální úlohu

$$\min (x_1, x_2) \quad \text{za podm. } x_1, x_2 \geq 0.$$

Zřejmě $\mathcal{E} = \{(0, 0)\}$, ale $\cup_{\lambda \in \bar{S}} M_{opt}(\lambda)$ tvoří celou hranici nezáporného kvadrantu. Například pro $\lambda = (1, 0)$ je $M_{opt}(\lambda) = \{(0, t); t \geq 0\}$, což představuje nezápornou část osy x_2 . \square

Isermannova věta ještě řekne něco více o úloze (1.19).

Věta 1.26. *Úloha (1.19) neomezená právě tehdy, když $\mathcal{E} = \emptyset$.*

Důkaz. „ \Rightarrow “ Pro spor předpokládejme, že $x^1 \in \mathcal{E}$, tedy podle věty 1.24 existuje $\lambda > 0$ takové, že

$$\lambda^T C x^1 \leq \lambda^T C x \quad \forall x \in M. \quad (1.20)$$

Dále, z neomezenosti (1.19) plyne, že pro každé $\alpha \in \mathbb{R}$ existuje $x \in M$ a příslušné y takové, že $Cx + y = Cx^0$ a $e^T y > \alpha$. Označíme-li $\lambda_{\min} := \min_i \lambda_i > 0$, pak dostáváme

$$\lambda^T y \geq \lambda_{\min} e^T y > \lambda_{\min} \alpha.$$

Substitucí $y = Cx^0 - Cx$ pak máme

$$\lambda^T (Cx^0 - Cx) > \lambda_{\min} \alpha,$$

neboli

$$\lambda^T Cx < \lambda^T Cx^0 - \lambda_{\min} \alpha.$$

To je spor s (1.20), říkající, že $\lambda^T Cx$ je zdola omezené.

„ \Leftarrow “ Je-li pro spor úloha (1.19) omezená, pak (protože je přípustná) musí mít optimum. Podle věty 1.23(2) je toto optimální řešení eficientním. \square

Poznámka 1.27. Věta 1.23 (s větou 1.26) poskytuje tzv. *fail-safe* způsob nalezení eficientního řešení. Pokud $\mathcal{E} \neq \emptyset$, tak vždy eficientní řešení najdeme. Na druhou stranu, skalarizací pomocí váženého součtu sice také dostaneme eficientní řešení, ale může se stát, že při konkrétní volbě vah je skalarizace neomezená (přestože $\mathcal{E} \neq \emptyset$). Tento nedostatek lze napravit tak, že uvažujeme pouze vhodné výběry vah, aby skalarizace nebyla neomezená. To jest, duální úloha musí být přípustná, neboli soustava

$$A^T y \leq C^T \lambda, \quad \lambda \geq e$$

musí být řešitelná.

Jak vypadá množina přípustných řešení \mathcal{E} geometricky?

Věta 1.28. *Množina eficientních řešení \mathcal{E} je uzavřená, souvislá a tvoří ji sjednocení několika na sebe navazujících stěn polyedru M .*

Důkaz. Víme z Isermannovy věty, že $\cup_{\lambda \in S} M_{opt}(\lambda) = \mathcal{E}$. Protože $M_{opt}(\lambda)$ je uzavřenou stěnou M , je \mathcal{E} sjednocením několika uzavřených stěn.

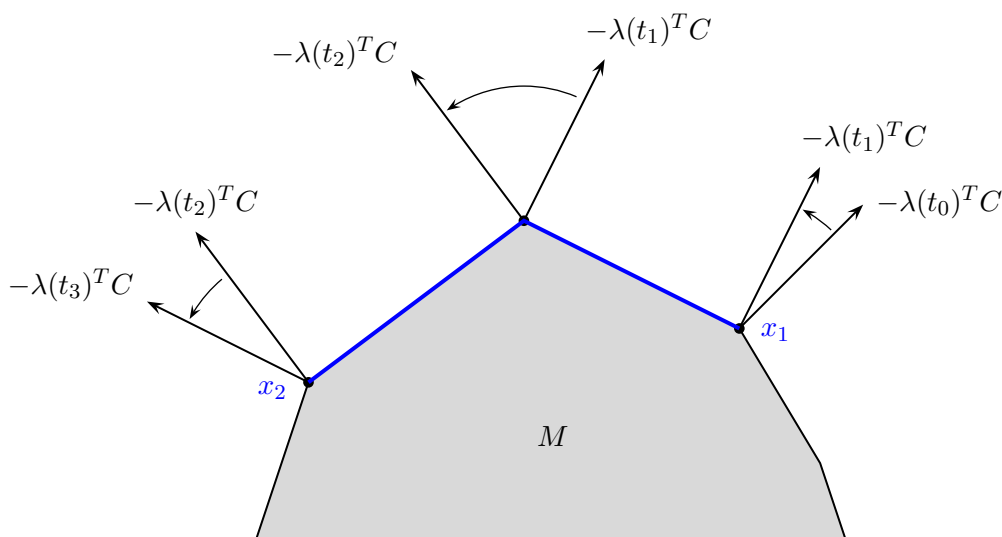
Zbývá ukázat, že tyto stěny na sebe navazují. Uvažme proto $x_1, x_2 \in \mathcal{E}$. Jím odpovídají $\lambda_1, \lambda_2 \in S$ takové, že $x_1 \in M_{opt}(\lambda_1)$ a $x_2 \in M_{opt}(\lambda_2)$. Uvažujme dále úsečku

$$\lambda(t) := \lambda_1 + t(\lambda_2 - \lambda_1) \in S, \quad t \in [0, 1].$$

Z teorie parametrického programování (viz poznámka 1.29) plyne $M_{opt}(\lambda) \neq \emptyset$ pro všechna $t \in [0, 1]$. Interval $[0, 1]$ se navíc rozpadne na takzvané obory stability

$$[0, 1] = \cup_{i=1}^k [t_{i-1}, t_i], \quad t_0 = 0, \quad t_k = 1.$$

Každý obor stability $[t_{i-1}, t_i]$ přísluší jedné bázi, která je optimální pro všechna $t \in [t_{i-1}, t_i]$. Navíc sousední obory stability přísluší sousedním bázím, tudíž jim odpovídající vrcholy na sebe navazují skrze hranu. Ty nám dají po částech lineární křivku v \mathcal{E} spojující x_1 a x_2 ; viz obrázek 1.6. \square



Obrázek 1.6: (důkaz věty 1.28) Body x_1, x_2 spojené křivkou (modře) v \mathcal{E} . Vrcholy na křivce jsou optimální řešení skalarizace pro příslušný interval $[t_{i-1}, t_i]$. Hrany na křivce jsou optimální pro hodnoty t_i .

Geometrie \mathcal{E} z pohledu parametrického programování

Díky Isermannově větě 1.24 lze eficientní řešení vyjádřit ekvivalentně pomocí skalarizací s kladnými vahami. Uvažujeme-li váhy λ jako parametry, můžeme strukturu množiny eficientních řešení \mathcal{E} nahlédnout z pohledu parametrického programování [Grygarová, 1994; Nožička et al., 1974].

Poznámka 1.29 (Parametrické programování a obory stability). Uvažujme nejprve úlohu lineárního programování, pro geometrickou názornost jako maximalizační

$$\max \mu^T x \quad \text{za podm. } Ax = b, x \geq 0, \quad (1.21)$$

kde $\mu \in \mathbb{R}^n$ je vektor parametrů. Předpokládejme, že je přípustná. Otázka zní, pro které hodnoty μ má optimum (tedy je omezená)?

Označme

$$\mathcal{A} := \{\mu \in \mathbb{R}^n; (1.21) \text{ má optimum}\}.$$

Je známo, že množina neomezených směrů \mathcal{R} polyedru M je popsána omezeními $Ax = 0, x \geq 0$ a představuje konvexní polyedrický kužel. Jsou-li $h_i, i \in I$, vektory jeho hran, pak \mathcal{A} představuje polární kužel k \mathcal{R} a je popsán

$$\mathcal{A} = \{\mu \in \mathbb{R}^n; h_i^T \mu \leq 0, i \in I\}.$$

Má-li úloha (1.21) optimum, pak se nabyde v aspoň jednom z jeho vrcholů; označme je $x_k, k \in K$. Pak lze psát

$$\mathcal{A} = \cup_{k \in K} \mathcal{A}_k,$$

kde

$$\mathcal{A}_k := \{\mu \in \mathbb{R}^n; (1.21) \text{ má optimum v } x_k\}.$$

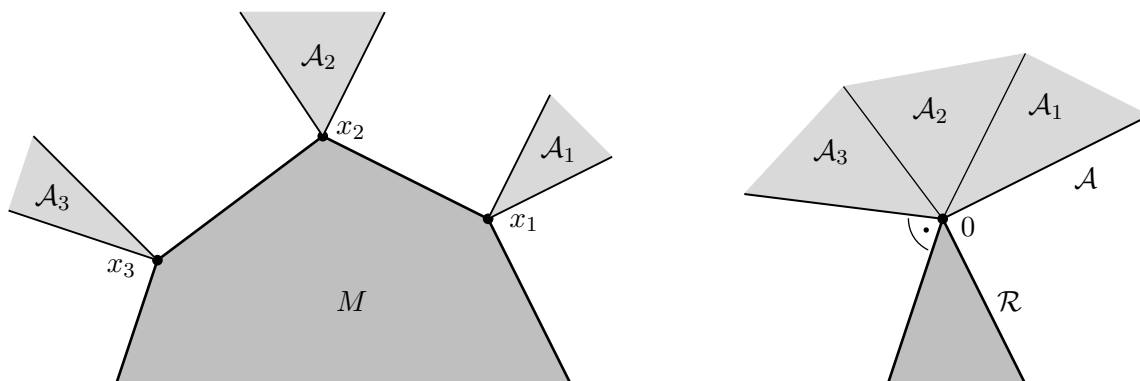
Označíme-li jako $h_i, i \in I(x_k)$, vektory ve směru hran, vedoucích z vrcholu x_k , pak platí

$$\mathcal{A}_k = \{\mu \in \mathbb{R}^n; h_i^T \mu \leq 0, i \in I(x_k)\},$$

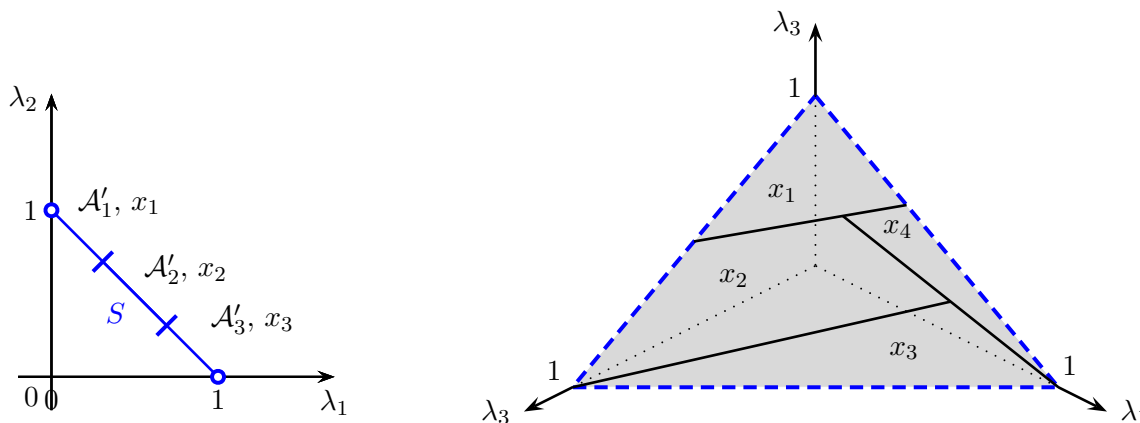
viz obrázek 1.7.

Vrátíme ze zpět ke skalární úloze vícekriteriálního programování (1.18) substitucí $\mu := -C^T \lambda$. Tudíž množina

$$\mathcal{A}' = \{\lambda \in S; h_i^T C^T \lambda \geq 0, i \in I\}$$



Obrázek 1.7: (poznámka 1.29) Napravo kužel \mathcal{R} neomezených směrů polyedru M a duální kužel \mathcal{A} rozdělený na jednotlivé pod-kužele \mathcal{A}_k . Kužel \mathcal{A}_k představuje množinu vektorů účelové funkce, pro které se x_k nabyde jako optimum (vlevo).



Obrázek 1.8: (poznámka 1.30) Obory stability množiny vah S pro $s = 2$ a pro $s = 3$.

popisuje množinu vah $\lambda \in S$, pro které skalarizace vede na omezenou úlohu, a množina

$$\mathcal{A}'_k = \{\lambda \in S; h_i^T C^T \lambda \geq 0, i \in I(x_k)\}$$

popisuje množinu vah $\lambda \in S$, pro které skalarizace vede na úlohu s optimem v x_k .

Tento postup je možno také použít k nalezení množiny \mathcal{E} . Existují i jiné postupy, například vícekriteriální simplexová metoda [Ehrgott, 2005].

Poznámka 1.30 (Vizualizace vah a oborů stability). Normovaná množina vah $\{\lambda \in \mathbb{R}^s; \lambda > 0, e^T \lambda = 1\}$ představuje v prostoru \mathbb{R}^s takzvaný simplex (bez hranice), což je konvexní polyedr s právě $s + 1$ vrcholy. Pro $s = 2$ má tvar úsečky a obory stability \mathcal{A}'_k z poznámky 1.29 představují pod-úsečky. Pro $s = 3$ má S tvar trojúhelníku a obory stability \mathcal{A}'_k jsou pak mnohoúhelníky v tomto trojúhelníku. Hrany dotyku dvou mnohoúhelníků odpovídají situaci, kdy je optimální celá hrana, spojující odpovídající dva vrcholy. Podobně bod dotyku tří mnohoúhelníků odpovídá situaci, kdy je optimální celá stěna, určená odpovídajícími třemi vrcholy. Obory stability se pak typicky znázorňují jak je uvedeno na obrázku 1.8.

Kapitola 2

Přístupy k řešení

Tato kapitola je věnována různým metodám na řešení úloh vícekritériálního programování. V zásadě lze podle [Hwang and Masud, 1979] rozdělit metody do čtyř kategorií:

- I. Informace od uživatele žádné nemáme.
- II. Informace od uživatele (důležitost jednotlivých funkcí atp.) dostaneme před formulací problému.
- III. Informace od uživatele dostáváme během výpočtu, jedná se tedy o algoritmy dialogu.
- IV. Informace od uživatele dostaneme až po skončení výpočtu.

Pokud není řečeno jinak, uvažujeme standardní úlohu (1.1), tj.

$$\min f(x) \quad \text{za podm. } x \in M.$$

2.1 Metody typu I.

Tyto metody jsou vytvořeny pro situaci, kdy od uživatele nemáme žádné dodatečné informace.

Aby byly jednotlivé účelové funkce porovnatelné, používá se v lineárním vícekritériálním programování $\min_{x \in M} Cx$ tzv. škálování [Steuer, 1986, sekce 8.4]. Toto škálování přenásobí normálové vektory kritérií tak, aby měli normu 1, čili i -tý řádek C_{i*} matice C nahradíme řádkem $\frac{1}{\|C_{i*}\|}C_{i*}$. Typicky se používá eukleidovská nebo maximová norma.

2.1.1 Metoda globální cílové funkce

Základní myšlenkou této metody je najít řešení, které je co nejbližší ideální hodnotě. Spočítej ideální hodnoty

$$F_i := \min_{x \in M} f_i(x), \quad i = 1, \dots, s.$$

Pokud tedy existuje ideální řešení $x^I \in M$, tak $f(x^I) = F$.

Metoda globální cílové funkce pak vede na úlohu

$$\min \|f(x) - F\| \quad \text{za podm. } x \in M.$$

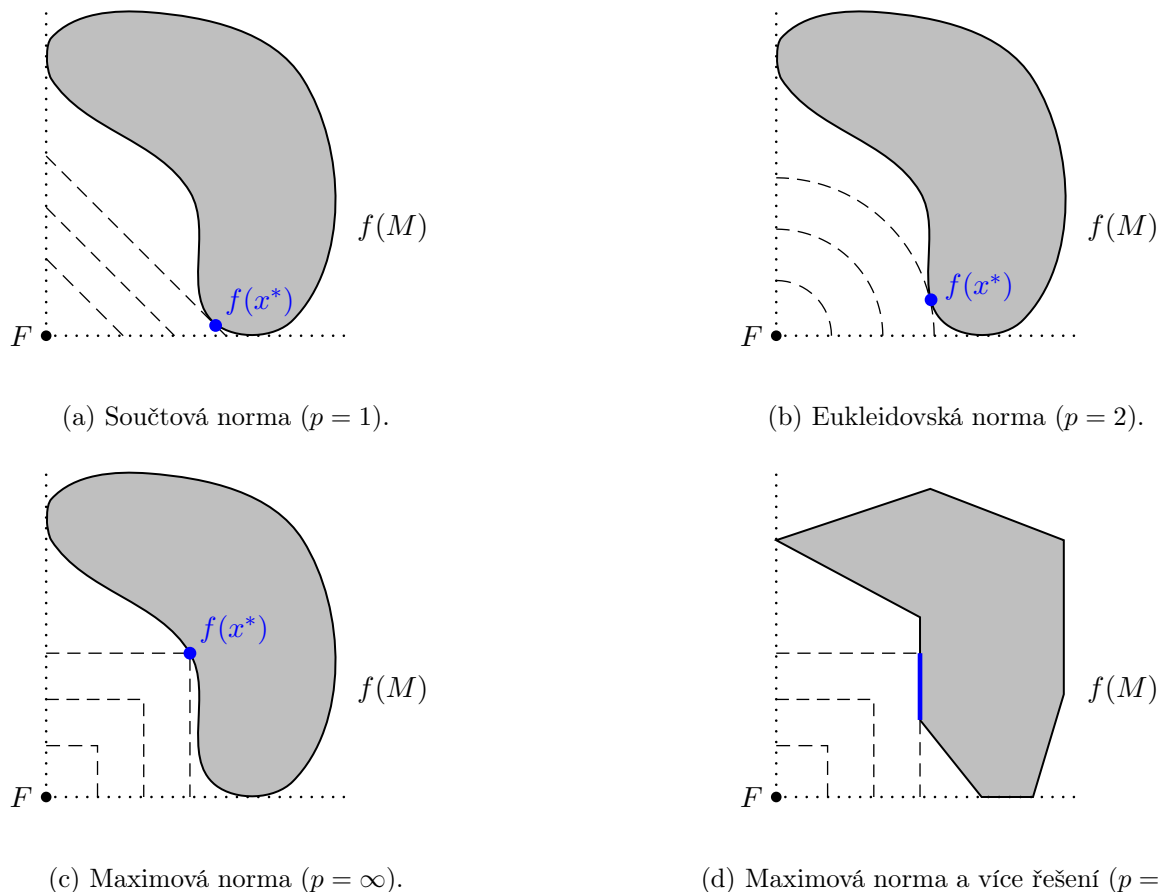
Aby proměnné byly bezrozměrné a nezáleželo tedy na škálování jednotek, uvažuje se často varianta

$$\min \left\| \left(\frac{f_1(x) - F_1}{F_1}, \dots, \frac{f_s(x) - F_s}{F_s} \right) \right\| \quad \text{za podm. } x \in M. \quad (2.1)$$

Zde předpokládejme, že $F > 0$.

Za normu se nejčastěji volí p -norma, která je definovaná

$$\|x\|_p := \left(\sum_{i=1}^n |x_i|^p \right)^{1/p}$$



Obrázek 2.1: Geometrický pohled na metodu globální cílové funkce jako na nalezení nejbližšího řešení k ideálnímu bodu F pro různou volbu p -normy.

pro $p \geq 1$. Nejčastější je volba $p \in \{1, 2, \infty\}$, přičemž hodnotou $p = \infty$ se rozumí limitní přechod, který dá

$$\|x\|_{\infty} := \max_{i=1, \dots, n} |x_i|.$$

Protože p -tá odmocnina je rostoucí funkce, stačí pro nalezení optimálního řešení uvažovat p -tou mocninou z normy, čímž se (2.1) zjednoduší na

$$\min \sum_{i=1}^s \left(\frac{f_i(x) - F_i}{F_i} \right)^p \quad \text{za podm. } x \in M. \quad (2.2)$$

Geometrický pohled na metodu globální cílové funkce je ilustrován na obrázku 2.1 pro různé volby norem.

Volba $p = 1$. V tomto případě má (2.2) tvar

$$\min \sum_{i=1}^s \frac{f_i(x) - F_i}{F_i} \quad \text{za podm. } x \in M.$$

Je-li původní úloha vícekriteriálního programování lineární, pak i tato úloha je lineárním programem. Podobně pokud je původní úloha konvexní, pak opět dostáváme konvexní optimalizační úlohu.

Volba $p = 2$. Nyní má (2.1) tvar

$$\min \sum_{i=1}^s \left(\frac{f_i(x) - F_i}{F_i} \right)^2 \quad \text{za podm. } x \in M.$$

Je-li původní úloha vícekriteriálního programování lineární, pak tato úloha už lineární není, ale aspoň bude konvexní kvadratickou úlohou. Pokud je původní úloha konvexní, pak opět dostáváme konvexní optimalizační úlohu, protože druhá mocnina nezáporné konvexní funkce je konvexní.

Volba $p = \infty$. Zde má (2.2) tvar

$$\min \max_{i=1, \dots, s} \frac{f_i(x) - F_i}{F_i} \quad \text{za podm. } x \in M.$$

Tuto úlohu můžeme přepsat to jednodušší podoby

$$\min y \quad \text{za podm. } x \in M, \quad f_i(x) - F_i \leq F_i \cdot y, \quad i = 1, \dots, s.$$

Je-li původní úloha vícekriteriálního programování lineární, pak i tato úloha je lineární. Pokud je původní úloha konvexní, pak opět dostáváme konvexní úlohu.

Po řešení metodou globální cílové funkce přirozeně požadujeme, aby bylo eficientní. To za celkem obecných předpokladů platí.

Věta 2.1. Řešení úlohy (2.2) je eficientní pokud

- (1) $1 \leq p < \infty$, nebo
- (2) $p = \infty$ a řešení (2.2) je jednoznačné.

Důkaz. Pro spor předpokládejme, že řešení x^* úlohy (2.2) není eficientní. Tudíž existuje $x \in M$ takové, že $f(x) \not\leq f(x^*)$. To znamená, že pro všechna $i = 1, \dots, s$ platí

$$\frac{f_i(x) - F_i}{F_i} \leq \frac{f_i(x^*) - F_i}{F_i}$$

a aspoň pro jedno i platí ostrá nerovnost. V závislosti na p budeme dále odvozovat z těchto nerovností.

- (1) Nyní odvodíme

$$\left(\frac{f_i(x) - F_i}{F_i} \right)^p \leq \left(\frac{f_i(x^*) - F_i}{F_i} \right)^p,$$

a dále

$$\sum_{i=1}^s \left(\frac{f_i(x) - F_i}{F_i} \right)^p < \sum_{i=1}^s \left(\frac{f_i(x^*) - F_i}{F_i} \right)^p,$$

což je spor s tím, že x^* je optimum (2.2).

- (2) Zde odvodíme

$$\max_{i=1, \dots, s} \frac{f_i(x) - F_i}{F_i} \leq \max_{i=1, \dots, s} \frac{f_i(x^*) - F_i}{F_i},$$

což je opět spor s tím, že x^* je jednoznačné optimum (2.2). □

V případě $p = \infty$ je zapotřebí jednoznačnost a řešení (2.2). Jak ukazuje obrázek 2.1d, v opačném případě nemusí být řešení eficientní.

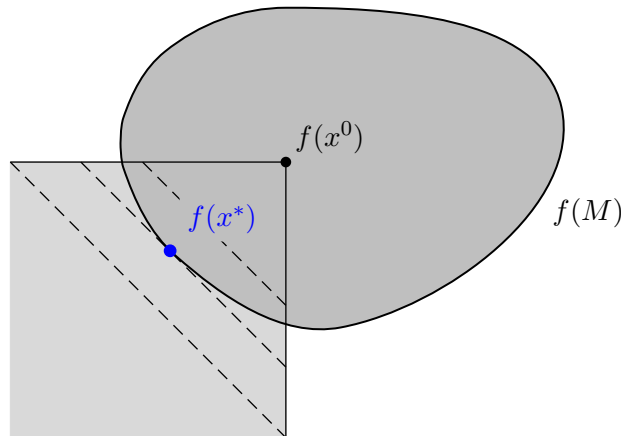
Příklad 2.2 (Lineární regrese). Podívejme se na lineární regresi z pohledu vícekriteriální optimalizace. Uvažujme přeuročenou soustavu rovnic $Ax = b$, kde $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$. Tato soustava typicky nemá řešení, tudíž hledáme přibližné řešení. Přirozené je hledat takový vektor x , pro který je nejmenší rozdíl mezi levou a pravou stranou, to vede na vícekriteriální úlohu

$$\min \|Ax - b\| \quad \text{za podm. } x \in \mathbb{R}^n,$$

kteřá má m účelových funkcí. Jak bude řešit tuto úlohu metoda globální cílové funkce? Ideální bod je $F = 0$, což vede na úlohu

$$\min \|Ax - b\|_p \quad \text{za podm. } x \in \mathbb{R}^n.$$

To je standardní postup řešení úlohy lineární regrese. Pro $p = 2$ se postup redukuje na metodu nejmenších čtverců, která, jak známo, má dobrou statistickou motivaci. Pokud hodnoty ve vektoru b jsou zatíženy nezávisle s chybou s normálním rozdělením, pak řešení naší úlohy je nejlepší nestranný odhad a také nejvěrohodnější odhad. I další volby $p \in \{1, \infty\}$ mají své opodstatnění, například v robustní regresi nebo pokud chyby mají jiné rozdělení (při Laplaceově rozdělení je nejvěrohodnější odhad optimum úlohy s volbou $p = 1$). □



Obrázek 2.2: Ilustrace Bensonovy metody: hledáme nejvzdálenější bod od $f(x^0)$ v kuželu dominujících směrů.

2.1.2 Bensonova metoda

Tato metoda využívá optimalizační úlohu

$$\max e^T y \quad \text{za podm. } x \in M, y = f(x^0) - f(x), y \geq 0, \quad (2.3)$$

kde $x^0 \in M$ je dané přípustné řešení. Hledáme tedy v množině $f(M)$ nejvzdálenější bod k bodu $f(x^0)$, a to v kuželu dominujících směrů k $f(x^0)$ a za použití součtové normy, viz obrázek 2.2.

Je snadné nahlédnout, že pro vícekriteriální lineární úlohu má tato úloha tvar lineárního programu. Pokud je původní úloha konvexní, pak úloha (2.3) již být nemusí díky podmínce $y = f(x^0) - f(x)$. Nicméně, konvexitu lze zachovat tím, že úlohu přepíšeme na ekvivalentní tvar

$$\max e^T y \quad \text{za podm. } x \in M, y \leq f(x^0) - f(x), y \geq 0.$$

Úloha (2.3) v zásadě zobecňuje úlohu (1.19) z vícekriteriálního lineárního programování a má i podobné vlastnosti. Otestuje, zda je daný bod x^0 eficientní, najde eficientní řešení a za celkem obecných předpokladů dokáže i rozhodnout o prázdnotě množiny eficientních řešení.

Věta 2.3. Platí $x^0 \in \mathcal{E}$ právě tehdy, když optimální hodnota úlohy (2.3) je 0.

Důkaz. „ \Rightarrow “ Sporem necht' je optimální hodnota kladná, tedy existuje přípustné řešení (x, y) takové, že $e^T y > 0$. To ale znamená, že $f(x^0) - f(x) = y \geq 0$, z čehož $f(x) \leq f(x^0)$, spor.

„ \Leftarrow “ Sporem necht' $x^0 \notin \mathcal{E}$, tedy existuje $x \in M$ takové, že $f(x) \leq f(x^0)$. Definuj $y := f(x^0) - f(x)$, pak (x, y) je přípustné řešení pro (2.3), ale s větší kriteriální hodnotou $e^T y > 0$, spor. \square

Věta 2.4. Bud' (x^*, y^*) optimální řešení úlohy (2.3). Pak $x^* \in \mathcal{E}$.

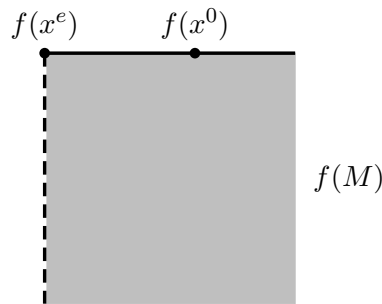
Důkaz. Sporem necht' $x^* \notin \mathcal{E}$, tedy existuje $x \in M$ takové, že $f(x) \leq f(x^*)$. Definuj $y := f(x^0) - f(x) \geq f(x^0) - f(x^*) = y^* \geq 0$, pak

$$e^T y = e^T (f(x^0) - f(x)) > e^T (f(x^0) - f(x^*)) = e^T y^*.$$

To je spor s optimalitou (x^*, y^*) . \square

Prázdnot množiny \mathcal{E} lze z úlohy (2.3) rozpoznat jen za určitých předpokladů. Jak ukazuje obrázek 2.3, uzavřenost množiny $f(M)$ je podstatná pro neexistenci eficientního řešení.

Věta 2.5 (Benson, 1978). Uvažujme úlohu konvexního vícekriteriálního programování a necht' (2.3) je neomezená. Pak



Obrázek 2.3: Ilustrace Bensonovy metody: přestože je úloha (2.3) neomezená, eficientní řešení existuje. Na obrázku je označeno jako x^e .

- (1) $\mathcal{E}^v = \emptyset$,
- (2) $\mathcal{E} = \emptyset$, je-li $f(M)$ uzavřená.

Důkaz. Dokážeme jen první část, pro druhou viz např. Ehrgott [2005]. Sporem necht' existuje $x^* \in \mathcal{E}^v$. Podle věty 1.16 je x^* optimum úlohy $\min_{x \in M} \lambda^T f(x)$ pro určité $\lambda > 0$. Definujme $\lambda_{\min} := \min_i \lambda_i > 0$.

Z neomezenosti úlohy (2.3) plyne, že pro každé $\alpha > 0$ existuje $x \in M$ takové, že $e^T(f(x^0) - f(x)) > \alpha$. Z toho odvodíme

$$\lambda^T(f(x^0) - f(x)) \geq \lambda_{\min} e^T(f(x^0) - f(x)) > \lambda_{\min} \alpha.$$

Po dosazení $\alpha := \lambda_{\min}^{-1} \lambda^T(f(x^0) - f(x^*))$ do této nerovnosti dostaneme

$$\lambda^T(f(x^0) - f(x)) > \lambda^T(f(x^0) - f(x^*)),$$

z čehož $\lambda^T f(x) < \lambda^T f(x^*)$. To je spor s $x^* \in M_{opt}(\lambda)$. □

Poznámka 2.6. Úloha (2.3) lze ekvivalentně vyjádřit jako

$$\max \|y\|_1 \quad \text{za podm. } x \in M, y = f(x^0) - f(x), y \geq 0.$$

V principu můžeme uvažovat i jiné normy než jen součtovou, ale výsledná úloha bude v kategorii konkávní optimalizace (maximalizujeme konvexní funkci) a nemusí tudíž být efektivně řešitelná. Jediná schůdná varianta je ještě ta s maximovou normou

$$\max_{i=1, \dots, s} \{ \max y_i \quad \text{za podm. } x \in M, y = f(x^0) - f(x), y \geq 0 \},$$

ale i ta vyžaduje řešit s pomocných optimalizačních úloh.

2.2 Metody typu II.

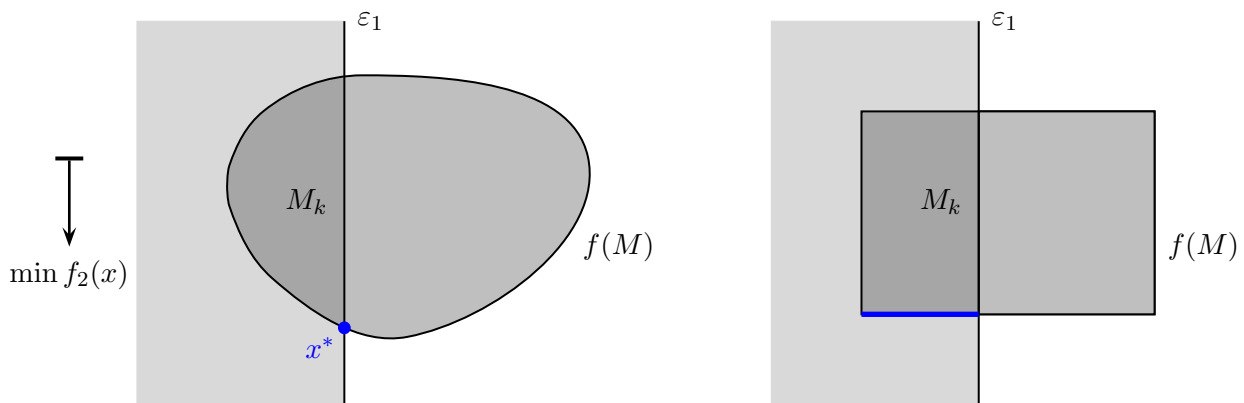
Zde rozebíráme přístupy k řešení úlohy vícekriteriální optimalizace $\min_{x \in M} f(x)$ pro situace, kdy máme předem od uživatele nějakou dodatečnou informaci. Ta se může týkat preferencí pro kritéria, nastavení vah kritériím atp.

2.2.1 Vážený součet

Uvažujme skalarizaci

$$\min \lambda^T f(x) \quad \text{za podm. } x \in M,$$

kde $\lambda \geq 0$ jsou konkrétní váhy nastavené uživatelem. Víme, že optimální řešení je vždy vlastní eficientní řešení pokud jsou váhy kladné (věta 1.10) nebo pokud je optimální řešení jednoznačné (věta 1.14).



Obrázek 2.4: Ilustrace metody ε -omezení pro $k = 2$. Vlevo jednoznačné řešení, které je eficientní. Napravo více optimálních řešení (modře), z nichž pouze levý kraj úsečky je eficientní.

2.2.2 Metoda ε -omezení

Metoda ε -omezení (Haimes, Lasdon, Wismer, 1971) je nejvýznačnější metodou kategorie metod typu II. Mějme zadané hodnoty $\varepsilon_1, \dots, \varepsilon_s$, což jsou nejvyšší přijatelné hodnoty pro jednotlivá kritéria. Pro předem vybrané $k \in \{1, \dots, s\}$ pak řešíme úlohu

$$\min f_k(x) \quad \text{za podm. } x \in M_k, \quad (2.4)$$

kde

$$M_k = \{x \in M; f_i(x) \leq \varepsilon_i, \forall i \neq k\},$$

viz obrázek 2.4.

Pro úlohu vícekritériální konvexní optimalizace dostaneme konvexní problém a podobně úloha vícekritériální lineární optimalizace se redukuje na lineární programování. Tudíž úloha zůstává stejného typu.

Věta 2.7. *Buď x^* jednoznačné řešení úlohy (2.4). Pak $x^* \in \mathcal{E}$.*

Důkaz. Sporem necht' existuje $x \in M$ takové, že $f(x) \not\leq f(x^*)$. Pak $x \in M_k$ a navíc $f_k(x) \leq f_k(x^*)$, což je spor s jednoznačnou optimalitou x^* . \square

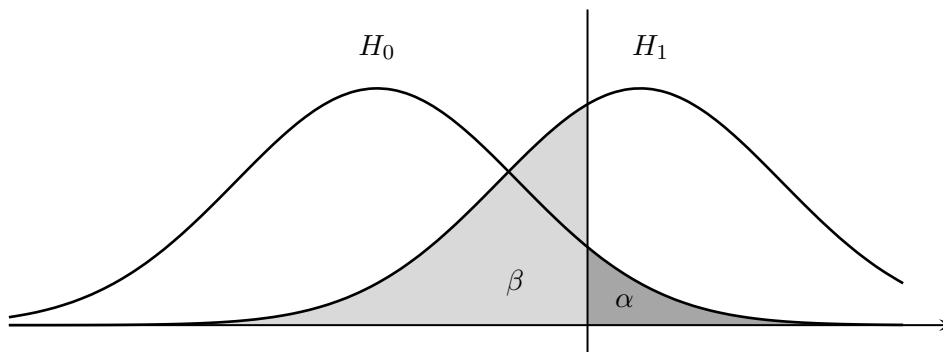
Věta 2.8. *Buď $x^* \in M$. Pak $x^* \in \mathcal{E}$ právě tehdy, když existuje $\varepsilon \in \mathbb{R}^s$ takové, že x^* je optimálním řešením úlohy (2.4) pro každé $k \in \{1, \dots, s\}$.*

Důkaz. „ \Rightarrow “ Polož $\varepsilon := f(x^*)$ a buď $k \in \{1, \dots, s\}$. Pro spor předpokládejme, že x^* není optimem úlohy (2.4). Tudíž existuje $x \in M_k$ takové, že $f_k(x) < f_k(x^*)$. Protože ale pro všechna $i \neq k$ platí $f_i(x) \leq \varepsilon_i = f_i(x^*)$, tak dostáváme $f(x) \not\leq f(x^*)$, spor.

„ \Leftarrow “ Sporem necht' $x^* \notin \mathcal{E}$. Pak existuje $x \in M$ takové, že $f(x) \not\leq f(x^*)$. To znamená, že $f(x) \leq f(x^*)$ a existuje index k takový, že $f_k(x) < f_k(x^*)$. Pak ale $x \in M_k$ a dá lepší hodnotu účelové funkce úlohy (2.4), což je spor s optimalitou x^* . \square

Příklad 2.9 (Testování statistických hypotéz). Častá statistická úloha je testování hypotézy H_0 oproti alternativě H_1 . Objevuje se v různých oborech – v zemědělství můžeme testovat zda nové hnojivo má průkazný vliv na výnos, v medicíně zda nový lék snižuje hladinu cholesterolu pod určitou mez, či v průmyslu zda životnost žárovek je alespoň určitá doba.

Při testování mohou nastat dva typy chyb. Chyba prvního druhu (pravděpodobnost α) je ta, že hypotéza platí, ale my ji zamítneme. Chyba druhého druhu (pravděpodobnost β) je naopak ta, že hypotéza neplatí, ale my ji nezamítneme. Chceme minimalizovat obě chyby, ale protože se chyby navzájem vylučují, je potřeba na to použít metody vícekritériální optimalizace. Konkrétně se používá metoda ε -omezení s tím, že se stanoví, aby chyba prvního druhu byla například nanejvýš $\alpha = 5\%$ a chceme minimalizovat β . To pak vede na určitou hraniční hodnotu dané statistiky, pod kterou hypotézu nezamítáme a nad kterou ji už zamítneme (ve statistické hantýrce „zamítáme na hladině α “), viz obrázek 2.5. \square



Obrázek 2.5: (Příklad 2.9) Testování hypotézy H_0 oproti alternativě H_1 , chyba prvního druhu (α) a druhého druhu (β).

Příklad 2.10 (Problém výběru portfolia). Jedná se o učebnicový příklad použití konvexního kvadratického programování. Průkopníkem v této oblasti byl Harry Markowitz, nositel Nobelovy ceny za ekonomii z roku 1990, i když jeho výsledek je z už roku 1952.

Základní úloha zní: Máme kapitál K a můžeme investovat to n investic, investice i vynese c_i jednotek peněz. Problém nalezení optimálního portfolia investic vede na triviální úlohu lineárního programování

$$\max c^T x \quad \text{za podm.} \quad e^T x = K, \quad x \geq 0.$$

Výnos investic ale nebývá známý dopředu a modeluje se jako náhodná veličina. Nechť náhodný vektor c má střední hodnotu $\tilde{c} := E c$ a kovarianční matici $\Sigma := \text{cov } c = E (c - \tilde{c})(c - \tilde{c})^T$, která je pozitivně semidefinitní (to se nahlédne snadno, neboť pro každé x je $x^T \Sigma x = E x^T (c - \tilde{c})(c - \tilde{c})^T x = E (x^T (c - \tilde{c}))^2 \geq 0$). Pak pro reálný vektor $x \in \mathbb{R}^n$ máme, že střední hodnota $c^T x$ je $E(c^T x) = \tilde{c}^T x$ a rozptyl $c^T x$ je $\text{var}(c^T x) = x^T \Sigma x$.

Chceme maximalizovat střední hodnotu výnosu, ale zároveň minimalizovat risk. To vede na vícekritériální úlohu

$$\max (\tilde{c}^T x, -x^T \Sigma x) \quad \text{za podm.} \quad e^T x = K, \quad x \geq 0.$$

Metoda ε -omezení pak redukuje problém na úlohu

$$\max \tilde{c}^T x \quad \text{za podm.} \quad x^T \Sigma x \leq \alpha, \quad e^T x = K, \quad x \geq 0,$$

kde $\alpha > 0$ je parametr averze vůči riziku zadaný uživatelem. Konservativnější uživatelé volí nižší hodnotu (vede spíš na investice do dluhopisů), zatímco vyšší hodnotu volí odvážnější uživatelé (vede spíš na investice do akcií). \square

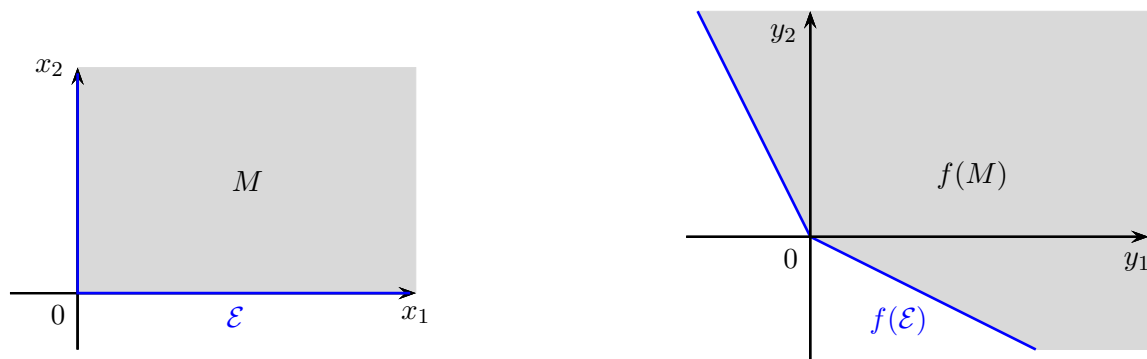
2.2.3 Lexikografická metoda

Zatímco dosavadní metody byly založené na redukcí vícekritériálního problému na jednokritériální, nyní se setkáme s metodou založenou na jiném principu. Základní myšlenka je zavést na prostoru \mathbb{R}^s úplné uspořádání. Konkrétně použijeme lexikografické uspořádání. Pro vektory $a, b \in \mathbb{R}^s$ říkáme, že a je lexikograficky menší než b pokud pro nějaké k platí

$$a_1 = b_1, \dots, a_k = b_k, \quad a_{k+1} < b_{k+1}.$$

Úlohu $\min_{x \in M} f(x)$ pak lexikografickou metodou řešíme tak, že hledáme lexikografické minimum vektorové funkce $f(x)$ na množině M . Zkráceně,

$$\text{lex min } f(x) \quad \text{za podm.} \quad x \in M.$$



Obrázek 2.6: (Příklad 2.12) Eficientních řešení je nekonečně mnoho, ale lexikografická metoda nevydává řešení, protože úloha je neomezená.

Prakticky můžeme lexikografické minimum najít iterativním procesem, kde v k -té iteraci minimalizujeme k -té kritérium na množině optimálních hodnot v předchozích kritériích. Konkrétně, definuj $k := 1$, $M_1 := M$ a řeš úlohu

$$\min f_k(x) \text{ za podm. } x \in M_k,$$

Je-li optimum x^k jednoznačné, končíme. V opačném případě definujeme

$$\begin{aligned} M_{k+1} &:= \{x \in M_k; f_k(x) = f_k(x^k)\}, \\ k &:= k + 1 \end{aligned}$$

a postup opakujeme.

Věta 2.11. *Optimální řešení x^* je eficientní.*

Důkaz. Sporem necht' existuje $x \in M$ takové, že $f(x) \not\leq f(x^*)$. Pak $f(x)$ je lexikograficky menší než $f(x^*)$, spor. \square

Příklad 2.12. Věta sice zaručuje, že optimální řešení je eficientní, ale může se stát, že $\mathcal{E} \neq \emptyset$ a přesto lexikografická metoda nemá optimální řešení. Uvažujme například úlohu

$$\min \begin{pmatrix} -1 & 2 \\ 2 & -1 \end{pmatrix} x \text{ za podm. } x \geq 0.$$

Eficientních řešení je nekonečně mnoho, jsou to nezáporné části souřadných os. Lexikografická metoda ale nevydává řešení, protože první (i druhá) účelová funkce je neomezená na množině přípustných řešení, viz obrázek 2.6. \square

Je vidět, že metoda je velmi citlivá na pořadí účelových funkcí, kdy první z nich přikládá neomezeně velkou váhu a na ostatních závisí jen okrajově. Proto se metoda používá jen ve speciálních případech nebo v kombinaci s jinými přístupy, viz sekce 2.2.5 o cílovém programování.

Příklad 2.13 (Dvě fáze v lineárním programování). Lexikografickou metodou je v zásadě dvoufázová simplexová metoda v lineárním programování. V první fázi se hledá počáteční přípustné báze řešení a ve druhé dochází k samotnému hledání minima procházením sousedních vrcholů se zmenšující se hodnotou účelové funkce. Formálněji, uvažujme úlohu lineárního programování

$$\min c^T x \text{ za podm. } Ax \leq b, x \geq 0.$$

Pro první fázi zavedeme novou proměnnou $y \geq 0$ a s velkou kladnou konstantou K transformujeme omezení na systém

$$Ax \leq b + K \cdot ey, x, y \geq 0.$$

V tomto systému snadno vidíme přípustné bázecké řešení: je to bod $x = 0$, $y = -\frac{1}{K} \min\{b_1, \dots, b_m, 0\}$.

Obě fáze můžeme sloučit do jedné tak, že zároveň hledáme přípustný bod a minimalizujeme účelovou funkci. To můžeme vyjádřit jako dvou-kriteriální úlohu lineárního programování

$$\min (y, c^T x) \text{ za podm. } Ax \leq b + K \cdot ey, \quad x, y \geq 0.$$

Lexikografická metoda pak odpovídá dvěma fázím simplexové metody: nejprve hledáme počáteční přípustné řešení minimalizací proměnné y ; každý vektor s nulovou y -ovou složkou odpovídá přípustnému řešení původní úlohy. V rámci druhého kriteriálu minimalizujeme původní účelovou funkci. To první kritérium je fundamentální, takže minimalizujeme pouze ta řešení, která jsou přípustná. \square

S lexikografickou metodou se lze setkat i v project managementu. Uvažujme síť aktivit spolu s časem potřebným na jejich zpracování a kauzálními vztahy mezi nimi. Metoda PERT pak slouží k nalezení nejkratšího času potřebného na dokončení projektu. Optimálních rozvrhů jednotlivých aktivit ale může být mnoho, čili má smysl uvažovat i sekundární kriteriála na časovou vyváženost aktivit atp.

2.2.4 Goal attainment

Tato metoda (F.W. Gembicki, 1974) je podobná metodě globální cílové funkce (sekce 2.1.1). V určité normě chceme dosáhnout účelovými funkcemi jistého cíle $G \in \mathbb{R}^s$. Cílový vektor G je daný uživatelem a tento postup vede na úlohu

$$\min \|f(x) - G\| \text{ za podm. } x \in M.$$

Zde se nejčastěji používá maximová norma. Pak úloha

$$\min \|f(x) - G\|_\infty \text{ za podm. } x \in M$$

se dá vyjádřit jako

$$\min y \text{ za podm. } x \in M, \quad f_i(x) - G_i \leq y, \quad i = 1, \dots, s.$$

Zde je možná i varianta, kdy uživatel navíc přiřadí váhy jednotlivým kritériím, což vede na úlohu

$$\min \|(\lambda_1(f_1(x) - G_1), \dots, \lambda_s(f_s(x) - G_s))\| \text{ za podm. } x \in M,$$

kde $\lambda > 0$ je daný vektor vah. V maximové normě pak máme ekvivalentní vyjádření

$$\min y \text{ za podm. } x \in M, \quad \lambda_i(f_i(x) - G_i) \leq y, \quad i = 1, \dots, s.$$

Poznámka 2.14. Pokud $G_i \leq \min_{x \in M} f_i(x)$ (to znamená, že pokud cílové hodnoty jsou menší než ideální bod F), potom řešení metodou *goal attainment* má stejné vlastnosti jako řešení metodou globální cílové funkce. V opačném případě řešení nemusí být eficientní.

2.2.5 Cílové programování

Cílové programování (Charnes & Cooper, 1961) je velmi populární metodou pro svoji variabilitu (v zásadě to není konkrétní metoda, ale spíš metodologie), na druhou stranu nemá tak pěkné teoretické vlastnosti.

Podobně jako v metodě *goal attainment* (sekce 2.2.4) uvažujeme nějaký cílový vektor $G \in \mathbb{R}^s$. Tento vektor stále může být nedosažitelný naráz, ale jednotlivé složky dosažitelné být mohou. Uvažme například model optimalizace výroby, ve kterém maximalizujeme zisk (tedy G_i nemusí být dosažitelná), ale také minimalizujeme přesčasy nebo chceme naplnit co možná přesně požadavky výroby (zde G_i může být dosažitelná a chceme ji víceméně naplnit). Z výše uvedeného proto hodnota $f(x) - G$ nemusí být nezáporná a má smysl si definovat její kladné a záporné části. Tedy rozložíme

$$f_i(x) - G_i = d_i^+ - d_i^-, \quad d_i^+, d_i^- \geq 0. \quad (2.5)$$

Nyní převedeme vícekriteriální úlohu $\min_{x \in M} f(x)$ na úlohu

$$\min (d_1^+, \dots, d_s^+, d_1^-, \dots, d_s^-) \text{ za podm. } x \in M, \quad (2.5)$$

kteřá je překvapivě také vícekriteriální, dokonce s dvojnásobným počtem kritérií. Na tuto úlohu můžeme aplikovat některé z předcházejících method, například vážený součet, lexikografickou metodu nebo jejich kombinaci:

$$\begin{aligned} &\min \sum_{i=1}^s (d_i^+ + d_i^-) \text{ za podm. } x \in M, \quad (2.5), \\ &\text{lex min } (d_1^+, \dots, d_s^+, d_1^-, \dots, d_s^-) \text{ za podm. } x \in M, \quad (2.5), \\ &\text{lex min } (\sum_{i=1}^s d_i^+, d_1^-, \dots, d_s^-) \text{ za podm. } x \in M, \quad (2.5). \end{aligned}$$

Můžeme minimalizovat maximální odchylku

$$\min \alpha \text{ za podm. } d_i^+, d_i^- \leq \alpha, x \in M, \quad (2.5).$$

Tento přístup tedy umožňuje širokou variabilitu postupů a zvolit tak ten nejlepší pro daný problém. Na druhou stranu, je obtížné říct teoretické vlastnosti řešení – obecně nemusí být eficientní, protože bod G může ležet ve vnitřku množiny $f(M)$.

Příklad 2.15 (Příklady převzané a upravené z Winston [2004]).

- Reklamní agentura chce rozvrhnout televizní reklamu pro První univerzitní pivovar. Cílem je, aby reklamu vidělo aspoň 0.8 milionu mužů, 0.1 milionu žen a 0.5 milionu nízkopříjmových lidí. Podmínky jsou omezení na rozpočet.
- Uzenářství VEG-AN vyrábí párky obsahující mj. hovězí maso, vepřové maso, sóju a vodu. Chce najít optimální složení uzeniny tak, aby byly splněny cíle: párek obsahuje aspoň 15% bílkovin, 8% tuku a stojí maximálně 9.90 Kč za 100 g.
- Fakulta antisociálních věd Kazimírovy Univerzity otevírá nový program v aplikované matematice. Cílem je, aby jej navštěvovalo celkově alespoň 100 studentů, alespoň 20 cizinců, alespoň 1 žena, ale maximálně 10 nerdů. \square

2.3 Metody typu III.

Metody tohoto typu jsou iterativní a interaktivní, kdy během komunikuje uživatel s počítačem. Na základě aktuální spokojenosti uživatele s dosaženými hodnotami účelových funkcí a jeho preferencí dochází k opakovanému přepočtu řešení dokud uživatel není spokojen s řešením. Viz například Steuer [1986].

Z podstaty většiny algoritmů nedokážeme u výsledného řešení garantovat, že je eficientní.

2.3.1 Metoda Geoffrion–Dyer–Feinberg

Tato metoda [Geoffrion et al., 1972] uvažuje konvexní úlohu vícekriteriálního programování $\min_{x \in M} f(x)$, kde účelové funkce jsou diferencovatelné a M je kompaktní množina. Dále metoda uvažuje, že existuje funkce užítka účelových funkcí, ale neumíme ji explicitně vyjádřit. Nicméně předpokládáme, že funkce užítka má tvar váženého součtu

$$U(f(x)) = w^T f(x), \quad w > 0,$$

kde váhy w neznáme. Tudíž naše úloha pak je

$$\min_{x \in M} U(f(x)) = \min_{x \in M} w^T f(x).$$

Buď x^k přípustné řešení v k -tém kroku. Pokud s ním uživatel není spokojen, tak vybere index i a pro každé $j \neq i$ určí že pohoršení účelové hodnoty $f_i(x^k)$ o Δf_i je rovnocenné s polepšením $f_j(x^k)$ o hodnotu Δf_j (se zachováním ostatních). Nyní z rovnosti

$$w^T f(x^k) = w^T f(x^k) + w_i \Delta f_i - w_j \Delta f_j$$

odvodíme $w_i \Delta f_i = w_j \Delta f_j$, z čehož jednoduše můžeme stanovit váhy pro k -tou iteraci jako

$$w_\ell := \frac{1}{\Delta f_\ell}, \quad \ell = 1, \dots, s.$$

Nyní jsme v zásadě hotovi, na úlohu $\min_{x \in M} w^T f(x)$ můžeme aplikovat libovolný řešič konvexní optimalizace. Tato metoda byla vyvinuta speciálně v souvislosti s gradientním algoritmem Frank–Wolfe (1956), proto si pro názornost ukážeme použití tohoto algoritmu. V k -té iteraci máme přípustné řešení x^k a provedeme dva kroky:

1. Nejprve najdeme směr největšího poklesu vyřešením pomocné úlohy

$$\min \nabla U(f(x^k))^T y \quad \text{za podm. } y \in M.$$

Tato úloha najde bod y^k takový, že směr klesání $y^k - x^k$ je nejslibnější, to jest, derivace v tomto směru je nejmenší. Zohledňujeme tedy nejenom strmost klesání, ale i vzdálenost ke kraji množiny M , neboť y^k je typicky na hranici M . Pro náš tvar užitečné funkce má pomocná úloha podobu konvexního programu

$$\min w^T \nabla f(x^k)^T y \quad \text{za podm. } y \in M.$$

2. V druhém kroku řešíme jednorozměrnou úlohu, kde hledáme minimum na polopřímce ve směru $y^k - x^k$,

$$\min U(f(x^k + t(y^k - x^k))) \quad \text{za podm. } t \in [0, 1].$$

Speciálně, pro naši úlohu dostaneme

$$\min w^T f(x^k + t(y^k - x^k)) \quad \text{za podm. } t \in [0, 1].$$

2.3.2 Metoda STEM

Metoda STEM (step method) [Benayoun et al., 1971] byla vyvinuta pro lineární vícekritériální programování, ale můžeme ji pojmut obecněji a uvažovat úlohu

$$\min_{x \in M} f(x) = \min_{x \in M} Cx$$

Nejprve nalezneme optima jednotlivých kritérií, tedy označme

$$F_i = \min_{x \in M} f_i(x)$$

a necht' se nabyde pro optimální řešení x^i . Předpokládejme $F > 0$. Dále označme jako

$$F_i^{\max} = \max_{j=1, \dots, s} f_i(x^j)$$

největší hodnotu i -tého kritéria na množině individuálních minim x^1, \dots, x^s . (Nejmenší hodnota je zřejmě F_i .) Pomocí těchto hodnot definujeme váhy

$$\lambda_i := \frac{F_i^{\max} - F_i}{F_i \|C_{i*}\|}$$

Metoda pak v zásadě postupuje algoritmem „Goal attainment“ (sekce 2.2.4) s volbou $G := F$, tedy řeší úlohu

$$\min y \quad \text{za podm. } x \in M, \lambda_i (f_i(x) - F_i) \leq y, \quad i = 1, \dots, s.$$

Označme x^0, y^0 optimální řešení. Uživatel nyní rozdělí kritéria do několika skupin. V indexové množině I jsou ty kritéria, ve kterých jsme dosáhli dostatečné hodnoty, tedy

$$I := \{i = 1, \dots, s; f_i(x^0) \text{ je vyhovující}\}.$$

Dále uživatel definuje podmnožinu $J \subset I$ těch kritérií, které se mohou i zhoršit v naději, že tím zlepšíme kritéria z $\{1, \dots, s\} \setminus I$. Pro každé $j \in J$ navíc uživatel stanoví hodnotu Δf_j o kterou se může hodnota j -tého kritéria pohoršit. Nyní přeformulujeme optimalizační úlohu a řešíme

$$\begin{aligned} \min y \quad \text{za podm.} \quad & x \in M, \\ & \lambda_i(f_i(x) - F_i) \leq y, \quad i \notin I, \\ & f_i(x) \leq f_i(x^0), \quad i \in I \setminus J, \\ & f_i(x) \leq f_i(x^0) + \Delta f_i, \quad i \in J. \end{aligned}$$

Celý postup opakujeme dokud není uživatel spokojen.

2.3.3 Algoritmy dialogu

Zde uživatel ještě více zasahuje do výpočtu a de facto řídí průběh algoritmu [Guddat and Wendler, 1984]. Uvažujeme úlohu lineárního vícekritériálního

$$\min_{x \in M} Cx$$

a jeho skalarizaci

$$\min_{x \in M} \lambda^T Cx \quad P(\lambda)$$

pro váhy $\lambda > 0$.

Průběh algoritmu: Uživatel zadá počáteční váhy $\lambda^0, \lambda^1 > 0$. Vyřešíme nejprve úlohu $P(\lambda^0)$ a potom uvažujeme úlohu jednoparametrické optimalizace

$$\min_{x \in M} (\lambda^0 + t(\lambda^1 - \lambda^0))^T Cx,$$

kde $t \in [0, 1]$ je parametr. Z teorie parametrického programování víme, že optimální hodnota v závislosti na t je konkávní a po částech lineární. Lineární části odpovídají tzv. oborům stability, což jsou podintervaly množiny parametrů $[0, 1]$ na kterých zůstává optimální řešení stejné. Algoritmus postupně vypočítává jednotlivé obory stability a jim příslušející optima a hodnoty kritérií a nechává je uživateli k posouzení. V závislosti na uživatelském rozhodnutí buďto pokračujeme ve směru $\lambda^1 - \lambda^0$ a generování dalších řešení, nebo uživatel změní směr. To může učinit změnou koncového vektoru λ^1 nebo počátečního vektoru λ^0 .

Dále se uživatel může vracet ke starším vektorům vah a různě kombinovat směry parametrizace.

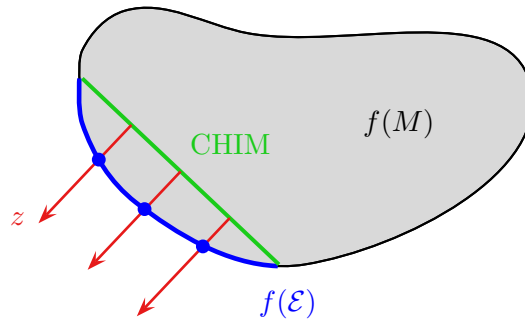
2.4 Metody typu IV.

V této skupině jsou metody, které dostávají informaci od uživatele až po skončení výpočtu. Prakticky to znamená, že musíme najít konečný výběr z eficientních řešení a nechat uživateli ať si z nich vybere to, které mu nejvíce vyhovuje, viz sekce 2.4.1.

Do této kategorie spadají i metody na sestavení celé množiny eficientních řešení \mathcal{E} , což je reálné ponejvíce ve vícekritériálním lineárním programování. Související problém je i najít vhodnou aproximaci množiny \mathcal{E} a jejího obrazu $f(\mathcal{E})$, toho se dotkneme v sekci 2.4.2.

2.4.1 Normal boundary intersection aneb jak najít vzorek

Vzorek bodů z \mathcal{E} . První myšlenka, která se nabízí, je využít skalarizace $\min_{x \in M} \lambda^T f(x)$, protože pro kladné váhy dostaneme vždy (vlastní) eficientní řešení. Volbou různých vah λ tak vygenerujeme vzorek eficientních řešení. Tento přímočarý přístup má ale i velkou nevýhodu. Vzhledem k citlivosti vůči škálování účelových funkcí snadno dostaneme shluky eficientních řešení, i když váhy λ vybíráme rovnoměrně. Takže není vždy snadné touto metodou najít aspoň trochu rovnoměrně rozprostřený vzorek eficientních řešení.



Obrázek 2.7: Ilustrace metody Normal boundary intersection.

Normal boundary intersection. Metoda Normal boundary intersection [Das and Dennis, 1998] slouží k nalezení co nejrovnoměrnějšího výběru eficientních řešení. V prvním kroku najde optimální řešení jednotlivých kritérií. Buď x^i optimum úlohy $\min_{x \in M} f_i(x)$, $i = 1, \dots, s$. Označme si konvexní obal jejich obrazů jako¹⁾

$$\text{CHIM} := \text{conv}\{f(x^1), \dots, f(x^s)\} = \{\sum_{i=1}^s \alpha_i f(x^i); \sum_{i=1}^s \alpha_i = 1, \alpha_i \geq 0\}.$$

Pokud nenastane degenerace, tak geometricky tato množina tvoří simplex, to jest konvexní polyedr dimenze $s - 1$ o s vrcholech.

Nyní metoda najde rovnoměrně rozdělený vzorek bodů na CHIM, v každém bodě ve směru normály najde nejvzdálenější bod na hranici $f(M)$, viz obrázek 2.7.

Výběr bodů z CHIM. Pro parametr $p \in \mathbb{N}$ rozdělíme interval $[0, 1]$ na p dílků. Hodnoty $\alpha_1, \dots, \alpha_s$ pak obsahují určitý počet (i nulový) dílků tak, aby celkový součet byl 1. Počet těchto možností je $\binom{p+s-1}{s-1}$, což můžeme nahlédnout tak, že si představíme sekvenci p dílků a $s - 1$ přepážek oddělujících hodnoty $\alpha_1, \dots, \alpha_s$. Počet možných uspořádání je stejný jako kombinací jak vybrat $(s - 1)$ -prvkovou množinu z $(p + s - 1)$ -prvkové.

Normála k CHIM. Normálu z k nadrovně obsahující simplex CHIM najdeme snadno jako vektor kolmý na jeho hrany, tedy jako nenulové řešení soustavy lineárních rovnic

$$(f(x^i) - f(x^1))z = 0, \quad i = 2, \dots, s.$$

Přestože je normála snadno spočitatelná, v principu můžeme použít i jiný směr, protože rovnoměrnost výběru zůstane víceméně zachována. Jako alternativní směr je možné uvažovat o vektoru tzv. kvazi-normály $-\sum_{i=1}^s f(x^i)$.

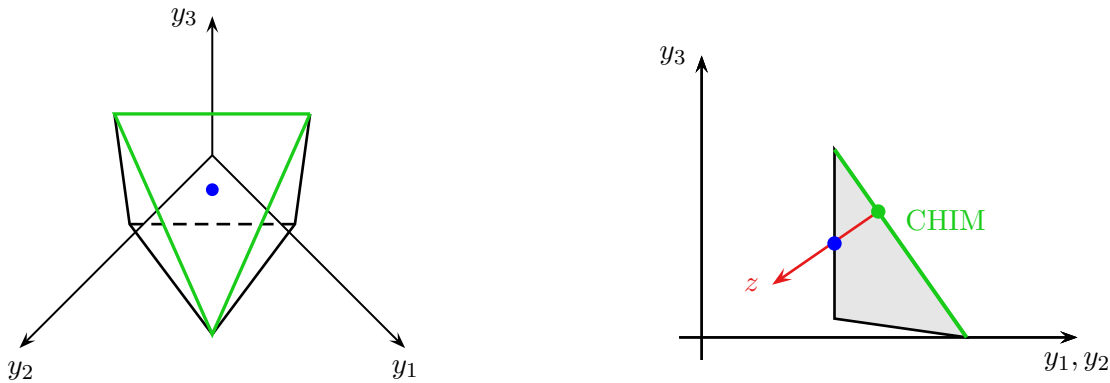
Nalezení průsečíku. Buď $y \in \text{CHIM}$ a z normála k CHIM. Nalezení nejvzdálenějšího bodu na hranici $f(M)$ ve směru z od bodu y lze vyjádřit jako optimalizační úlohu

$$\max t \quad \text{za podm.} \quad y + tz = f(x), \quad x \in M. \quad (2.6)$$

Pro vícekritériální lineární programování je úloha opět lineární, ale pro nelineární (byť konvexní) kritéria už nedostaneme konvexní úlohu díky rovnici $y + tz = f(x)$.

Problémy. Tato metoda má jisté slabiny: Jsou-li jednotlivé úlohy $\min_{x \in M} f_i(x)$ výpočetně složité, potom můžeme dostat pouze lokální optima a tím pádem CHIM není určen správně a výsledný vzorek není z celé Paretoovské hranice. Úloha (2.6) může mít nesouvislou množinu přípustných řešení a proto i nalezení globálního optima je obtížné. Toto jsou však obecné problémy, když zadaná úloha je výpočetně složitá (např. nekonvexní). Význačná slabina metody je však, že výsledný průsečík nemusí být na Paretoovské hranici.

¹⁾CHIM = Convex Hull of Individual Minima.



Obrázek 2.8: Ilustrace případu, kdy metoda Normal boundary intersection nevydává eficientní řešení. Nalevo pohled kolmo k CHIM, napravo pohled z boku (kolmo k rovině osy y_3 a středu os y_1, y_2).

Příklad 2.16. Tento příklad ukazuje, že výsledný průsečík nemusí ležet na Paretoovské hranici ani v případě vícekritériálního lineárního programování. Uvažujme příklad, kdy $f(M)$ je tvořeno konvexním obalem bodů

$$(1, 1, 0), (1, 0, 1), (0, 1, 1), (0.9, 0.1, 0.1), (0.1, 0.9, 0.1),$$

viz obrázek 2.8. Zde CHIM je trojúhelník tvořený prvními třemi body a normála k němu je $z = -(1, 1, 1)$. Uvažujme bod $y := \frac{2}{3}(1, 1, 1) \in \text{CHIM}$. Pak polopřímka $y + tz$, $t \geq 0$, opustí polyhedr $f(M)$ v bodě $\frac{1}{2}(1, 1, 1)$, který je však dominován bodem $(0.5, 0.5, 0.1)$, ležícím ve středu hrany posledních dvou vrcholů. \square

2.4.2 Vnitřní a vnější aproximace

Většinou aproximujeme množinu $f(\mathcal{E})$, protože to poskytuje hrubou informaci o Paretoovské hranici a tudíž také o dosažitelných hodnotách jednotlivých kritérií.

Narozdíl od běžného významu vnitřní a vnější aproximace, kdy se jimi myslí podmnožina a nadmnožina, se ve vícekritériálním programování uvažuje koncept dominance. Proto říkáme, že množina

- $A^I \subseteq \mathbb{R}^s$ je vnitřní aproximace $f(\mathcal{E})$ pokud každé eficientní řešení dominuje nějakému bodu $z \in A^I$, to jest pro každé $y \in f(\mathcal{E})$ existuje $z \in A^I$ takové, že $y \not\leq z$;
- $A^O \subseteq \mathbb{R}^s$ je vnější aproximace $f(\mathcal{E})$ pokud každé eficientní řešení je dominováno nějakým bodem $z \in A^O$, to jest pro každé $y \in f(\mathcal{E})$ existuje $z \in A^O$ takové, že $z \leq y$.

Takováto vnitřní a vnější aproximace pak dává i klasickou aproximaci ve smyslu inkluze, protože

$$f(\mathcal{E}) \subseteq (A^I + \mathbb{R}_-^s) \cap (A^O + \mathbb{R}_+^s).$$

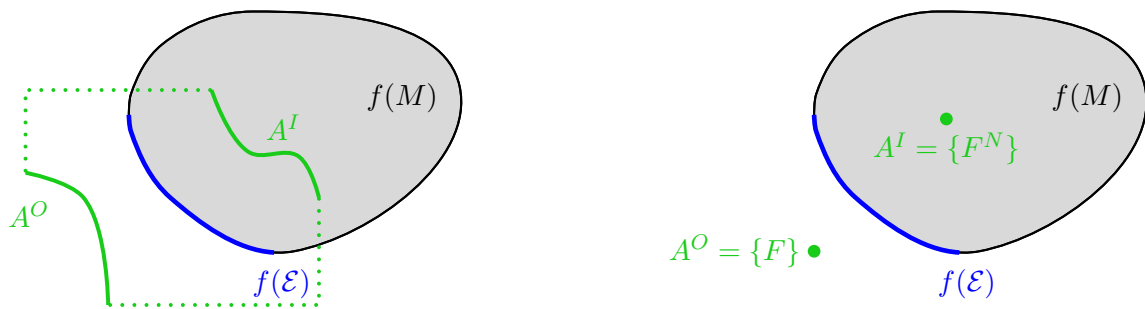
Jako vnější aproximace může sloužit ideální bod F , definovaný po složkách

$$F_i = \min_{x \in M} f_i(x), \quad s = 1, \dots, s.$$

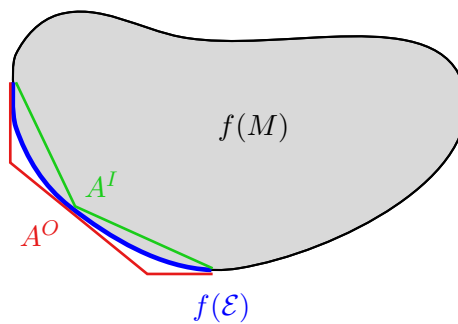
Je to jednoduchá jednobodová vnější aproximace, ale nemusí být těsná. Viz obrázek 2.9.

Jako protějšek ideálního bodu slouží u vnitřní aproximace tzv. nadir F^N (podnožník, tj. opak zenitu – nadhlavníku), definovaný

$$F_i^N = \max_{x \in \mathcal{E}} f_i(x), \quad s = 1, \dots, s.$$



Obrázek 2.9: Vnitřní a vnější aproximace Paretovské hranice $f(\mathcal{E})$ a její zapouzdření do množiny $(A^I + \mathbb{R}_-^s) \cap (A^O + \mathbb{R}_+^s)$, znázorněno zeleně. Napravo aproximace jednobodovými množinami A^I, A^O .



Obrázek 2.10: Vnitřní a vnější aproximace Paretovské hranice $f(\mathcal{E})$ pomocí po částech lineárních funkcí a pro případ dvou kritérií (tj. $s = 2$).

Aproximace přímkami pro $s = 2$

V konvexních úlohách se dvěma kritérii můžeme poměrně snadno sestrojít libovolně těsnou vnitřní a vnější aproximaci pomocí přímek, viz obrázek 2.10.

Nejprve sestrojíme konečný výběr bodů $y_0, y_1, \dots, y_s \in f(\mathcal{E})$, například vyřešením skalarizací

$$\min_{x \in M} \lambda f_1(x) + (1 - \lambda) f_2(x)$$

pro $\lambda \in \{0, \frac{1}{p}, \frac{2}{p}, \dots, 1\}$. Pospojujeme-li po sobě jdoucí body úsečkami, dostaneme vnitřní aproximaci. Pro vnější aproximaci pak sestrojíme tečné přímky v bodech y_0, y_1, \dots, y_s a části vymezené průsečíky dávají vnější aproximaci.

Poznámka 2.17. Množinu eficientních řešení můžeme aproximovat libovolně přesně pomocí intervalových metod. Tyto metody jsou založené na principu branch & bound a prohledávají přípustný prostor rigorózně, tedy se zárukou, že žádné eficientní řešení nevynecháme. Viz např. Araya et al. [2021]; Kubica [2019].

Kapitola 3

Vícekriteriální kombinatorické programování

V této kapitole se zabýváme kombinatorickými úlohami s více kritérii. Protože úloha kombinatorické optimalizace lze ekvivalentně vyjádřit pomocí celočíselného programování, můžeme na vícekriteriální případ nahlížet i jako na vícekriteriální celočíselné programování. Z nekonvexní podstaty této úlohy je patrné, že mnohé vlastnosti konvexní vícekriteriální optimalizace zde platit nebudou, například ne každé eficientní řešení získáme váženým součtem kritérií a vyřešením příslušné skalarizace. Nicméně, vždy máme zaručeno, že skalarizace s kladnými váhami povede na eficientní řešení (věta 1.9), stejně jako lexikografická metoda (sekce 2.2.3). Také nemá smysl vyšetřovat zvlášť vlastní eficientci, protože tento pojem splývá s eficientci.

Z diskrétní (a konečné) podstaty je eficientních řešení konečný počet. Má proto smysl nalézt všechna, nicméně může jich být exponenciálně mnoho vzhledem k dimenzi problému. Typicky také související rozhodovací problémy jsou NP-těžké i pro dvě kritéria: „Existuje řešení, které v obou kritériích má předem dané hodnoty?“

3.1 Nejkratší cesta s více kritérii

Uvažujme orientovaný graf $G = (V, E)$, kde V je množina n vrcholů a E množina m hran. Máme dán počáteční vrchol $z \in V$ a koncový vrchol $t \in V$. Dále máme vektorovou funkci vah $w: E \rightarrow \mathbb{R}^s$. Předpokládejme, že váhy jsou nezáporné (i když některé algoritmy si dovedou poradit i se zápornými váhami). Cílem je najít nejkratší cestu z vrcholu z do vrcholu t , přičemž délku měříme součtem vah na použitých hranách.

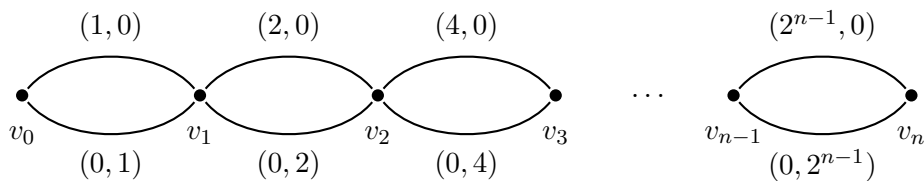
Toto je zjednodušený model, praktické úlohy mají rozličnější typy kritérií než jenom součet vah. Můžeme minimalizovat nejdelsí hranu, případně mít kritéria, která nejsou separovatelná. Na druhou stranu, minimalizovat počet použitých hran lze jednoduše za použití vah o hodnotě 1. Podobně v tomto duchu jsou kritéria minimalizace vzdálenosti, času atp.

Přesto i takovýto model a jeho vlastnosti odráží vlastnosti obecnějších modelů. Například, že eficientních cest může být exponenciálně mnoho a všechny jsou navzájem neporovnatelné (optimálních cest může být exponenciálně mnoho i v jednokriteriálním případě, třeba když je účelová funkce nulová, ale optimální řešení nejsou neporovnatelná).

Příklad 3.1 (Hansen [1980]). Uvažme graf s vrcholy v_0, v_1, \dots, v_n . Mezi vrcholy v_i, v_{i+1} vedou dvě hrany s váhami $(2^i, 0)$ a $(0, 2^i)$, viz obrázek 3.1. Každá cesta od v_0 do v_n má délku $(a, 2^n - 1 - a)$, kde $a \in \{0, 1, \dots, 2^n - 1\}$, a i naopak každá takováto délka je dosažitelná pro určité $a \in \{0, 1, \dots, 2^n - 1\}$. Tyto délky jsou navzájem neporovnatelné, tedy odpovídající cesty jsou eficientní. Tudíž eficientní je každá z 2^n cest. \square

Pro jednokriteriální úlohu existuje řada efektivních metod na nalezení nejkratší cesty. K nalezení všech eficientních cest při více kritériích můžeme tyto metody celkem snadno adaptovat. Jako v jednokriteriálním případě jsou založeny na základním pozorování.

Tvrzení 3.2. *Každá podcesta eficientní cesty je opět eficientní.*



Obrázek 3.1: (Příklad 3.1) Graf s 2^n eficientními cestami mezi v_0 a v_n .

Vektorový Dijkstrův algoritmus [Martins, 1984]. Pro každý vrchol $v \in V$ uvažujeme dvě množiny vektorů: $D(v)$ označuje množinu dočasných délek cest od z do v , a $T(v)$ označuje množinu trvalých (nejkratších, nedominovaných) délek cest od z do v . Na začátku nastav

$$\begin{aligned} D(z) &:= \{0\}, \\ D(v) &:= \emptyset, \quad \forall v \neq z \\ T(v) &:= \emptyset, \quad \forall v \in V. \end{aligned}$$

Hlavní cyklus probíhá dokud platí $D(v) \neq \emptyset$ pro nějaké $v \in V$. V rámci cyklu vybereme nějaké eficientní minimum z množiny $\cup_{v \in V} D(v)$, typicky $\text{lex min } \cup_{v \in V} D(v)$. Nechť je jeho hodnota h a nabyde se pro vrchol $u \in V$. Nyní odebereme h z množiny $D(u)$ a přidáme do $T(u)$:

$$\begin{aligned} D(u) &:= D(u) \setminus \{h\}, \\ T(u) &:= T(u) \cup \{h\}, \end{aligned}$$

a aktualizujeme všechny dočasné délky cest, vedoucí přes u k jeho sousedům

$$D(v) := D(v) \cup \{h + w(u, v)\}, \quad (u, v) \in E.$$

Nakonec je potřeba z množiny $D(v)$ vyhodit neeficientní vektory, to jest ty, kterým dominuje nový vektor $h + w(u, v)$, anebo naopak vyhodit nový vektor, pokud mu dominuje některý ze starých.

Na rozdíl od klasického Dijkstrova algoritmu nestačí skočit když narazíme na vrchol t , ale musíme doběhnout až do konce, dokud existuje dočasná cesta.

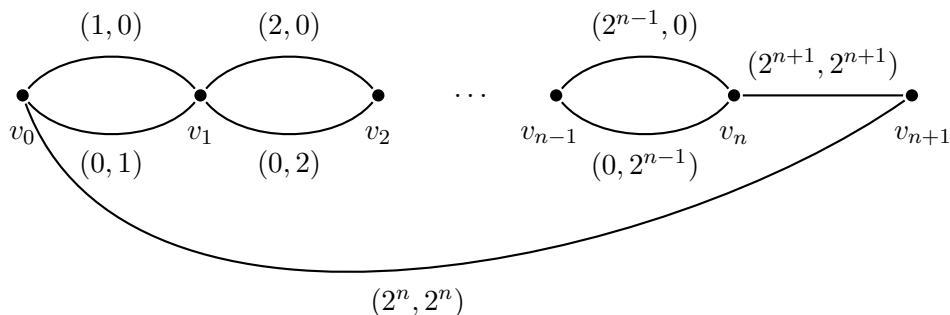
Věta 3.3. *Algoritmus najde všechny eficientní cesty z vrcholu z do všech ostatních.*

Důkaz. Myšlenka důkazu je založena na dvou pozorováních:

1) Hodnota $c \in T(v)$ určuje délku eficientní cesty od z do v kdykoli v průběhu algoritmu. Nahlédneme to sporem. Hodnota c odpovídá nějaké cestě od $z = v_1, \dots, v_k = v$ a zřejmě odpovídající částečné délky jsou v příslušných množinách $T(v_1), \dots, T(v_k)$. Pokud cesta v_1, \dots, v_k není eficientní, pak jí dominuje jiná eficientní cesta. Vrcholy na této cestě jsme ale museli v průběhu algoritmu zpracovat, označit jako trvalé a c odstranit jako dominované. Spor.

2) Najdeme všechny cesty. Pro spor nechť nenajdeme eficientní cestu $z = v_1, \dots, v_k$, a nechť délky do v_1, \dots, v_i , kde $i < k$, jsou označeny jako trvalé. Pak ale i hranu (v_i, v_{i+1}) musíme zpracovat, a protože je nedominovaná, tak i časem zařadit do trvalých. Spor. \square

Poznámka 3.4 (Ještě ke složitosti). Algoritmus na rozdíl od jednokriteriální verze už není polynomiální. Ani být nemůže, protože víme, že eficientních cest může být exponenciálně mnoho. Zajímavé je, že algoritmus není polynomiální ani vzhledem k n a počtu eficientních cest. Stačí uvažovat příklad 3.1 s tím, že přidáme navíc vrchol v_{n+1} a váhy dle obrázku 3.2. Pak je nejkratší cesta z v_0 do v_{n+1} jediná, a to přímá hrana mezi těmito vrcholy. Nicméně algoritmus najde všechny eficientní cesty z v_0 do všech ostatních vrcholů, tedy i 2^n cest z v_0 do v_n .

Obrázek 3.2: (Poznámka 3.4) Graf s 2^n eficientními cestami mezi v_0 a v_n .

Podobným způsobem můžeme adaptovat na vícekritériální případ Bellmanův–Fordův algoritmus, viz Ehrgott [2005]. Ten funguje i pro záporné váhy, ale v grafu nesmí existovat záporný cyklus; toto jsme nicméně schopni detekovat.

Najít nějakou eficientní cestu je jednoduché pomocí váženého součtu s kladnými vahami. Jinak je řada otázek složitá [Serafini, 1987]. Pro $s = 2$ už je NP-úplný rozhodovací problém: „Pro dané $k_1, k_2 \in \mathbb{N}$, existuje cesta délky nanejvýš (k_1, k_2) ?“ Problém je i $\#P$ -úplný, tj. určit počet eficientních cest je také složitě.

Věta 3.5 (Serafini [1987]). *Následující rozhodovací problém pro případ dvou kritérií je NP-úplný: Pro dané $k_1, k_2 \in \mathbb{N}$, existuje cesta délky nanejvýš (k_1, k_2) ?*

Důkaz. Problém je zřejmě ve třídě NP a NP-těžkost ukážeme redukcí z problému batohu

$$\max c^T x \quad \text{za podm.} \quad a^T x \leq b, \quad x \in \{0, 1\}^n,$$

kde $a, c > 0$. Rozhodovací verze problému batohu zní: Pro dané $k \in \mathbb{N}$, existuje $x \in \{0, 1\}^n$ takový, že $a^T x \leq b$ a $c^T x \geq k$? Druhou podmínku přepíšeme do tvaru $Kn - c^T x \leq Kn - k$. Jako v příkladu 3.1 zkonstruujeme graf s vrcholy $z = v_0, v_1, \dots, v_n = t$. Mezi vrcholy v_{i-1}, v_i vedou dvě hrany s váhami $(a_i, K - c_i)$ a $(0, K)$, kde K je dost velká konstanta, například $K := e^T c$. Potom v grafu existuje cesta délky nanejvýš $(b, Kn - k)$ právě tehdy, když je splněn problém batohu. \square

3.2 Minimální kostra s více kritérii

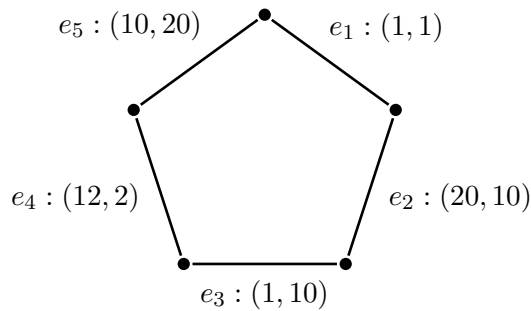
Uvažujme neorientovaný graf $G = (V, E)$, kde V je množina n vrcholů a E množina m hran. Máme vektorovou funkci vah $w: E \rightarrow \mathbb{R}^s$. Velikost kostry grafu je daná součtem vah na jejích hranách. Cílem je najít kostry s nejmenší (nedominovanou) velikostí.

Podobně jako v předchozí sekci může být navzájem neporovnatelných eficientních řešení exponenciálně mnoho.

Příklad 3.6. Buď $G = K_n$ úplný graf a uvažujme dvoukritériální funkci $w(e_i) = (2^{i-1}, 2^m - 2^{i-1})$, kde $m = \binom{n}{2}$ je počet hran. Každá kostra se skládá z $n - 1$ hran a má velikost (a, b) , kde $a + b = (n - 1)2^m$ je konstantní. Tudíž každá kostra je eficientní. Navíc z definice váhové funkce jsou velikosti různých koster navzájem různé. Počet koster je n^{n-2} , čili i tolik je eficientních řešení. \square

Podobná je i otázka složitosti. I pro dvě kritéria je NP-těžké rozhodnout, zda existuje kostra o velikosti maximálně k_1 a k_2 v prvním a druhém kritériu [Camerini et al., 1983].

K nalezení všech eficientních koster snadno adaptujeme základní metody na hledání minimálních koster.



Obrázek 3.3: (Příklad 3.8) Minimální kostra se dvěma kritérii.

Vektorový Jarníkův (Primův) algoritmus. Symbol \mathcal{M} bude značit množinu postupně vznikajících (eficientních) koster. Základní myšlenka je přidávat postupně nejmenší hrany tak, aby vznikal zvětšující se souvislý strom. Na začátku volíme všechny nedominované hrany

$$\mathcal{M} := \arg \min_{e \in E} w(e)$$

Nyní provedeme $n - 2$ iterací. V každé iteraci zvětšíme velikost každé zárodečné kostry T o jedna přidáním nedominované hrany $e = (i, j)$, jejíž jeden konec je v T a druhý mimo:

$$\mathcal{M} := \{T \cup \{e\}; T \in \mathcal{M}, w(e) \in \min\{w(i, j); i \in V(T), j \notin V(T)\}\},$$

kde $V(T)$ značí množinu vrcholů obsažených v grafu T .

Na konec z množiny \mathcal{M} odstraníme neeficientní kostry. To znamená, že pro každé dva podgrafy $T, T' \in \mathcal{M}$ takové, že $V(T) = V(T')$, porovnáme jejich váhy a pokud je jeden z nich dominován tím druhým, tak ho odstraníme. Pokud $V(T) \neq V(T')$, tak tento postup nesmíme použít, viz příklad 3.8.

Věta 3.7. *Algoritmus najde všechny eficientní kostry.*

Důkaz. Důkaz je podobný důkazu jednokriteriální verze. Výsledné kostry budou eficientní, protože ty neeficientní jsme odstranili na konci algoritmu. Zbývá ukázat, že najdeme všechny eficientní kostry. Předpokládejme pro spor, že eficientní kostru K nenajdeme. Uvažujme první takovou iteraci algoritmu, kdy na jejím konci žádný strom z množiny \mathcal{M} není částí kostry K . Na začátku této iterace takový strom T existuje. Protože jsme ho nezvětšili o navazující hranu e z kostry K , znamená to, že tato hrana je dominována jinou hranou e' . Přidáním e' do kostry K a odebráním vhodné navazující hrany e dostaneme jinou kostru, která dominuje kostře T . \square

Příklad 3.8. Uvažme graf G ve tvaru kružnice s pěti vrcholy a váhami dle obrázku 3.3. Postupujme podle algoritmu:

1. V prvním kroku vybereme $\mathcal{M} := \{\{e_1\}\}$.
2. V druhém kroku upravíme $\mathcal{M} := \{\{e_1, e_2\}, \{e_1, e_5\}\}$.
3. Ve třetím kroku upravíme $\mathcal{M} := \{\{e_1, e_2, e_3\}, \{e_1, e_5, e_4\}\}$. V množině \mathcal{M} ponecháme oba podgrafy, i když váha prvního je $(22, 21)$ a druhého $(23, 23)$. Podgrafy totiž nesdílejí stejné vrcholy.
4. Ve čtvrtém kroku v množině \mathcal{M} budou všechny kostry grafu kromě $G \setminus \{e_1\}$. Odstraněním dominovaných koster zůstanou dvě eficientní kostry $G \setminus \{e_2\}$ a $G \setminus \{e_5\}$.

Celkem tedy existují v grafu G dvě eficientní kostry $G \setminus \{e_2\}$ a $G \setminus \{e_5\}$. \square

3.3 Problém batohu s více kritérii

Tento problém má podobné vlastnosti jako předchozí. Uvedeme pro informatiky zajímavou aplikaci.

Příklad 3.9 (Web access problem). Mějme n informačních zdrojů, cena za použití i -tého z nich je c_i , přičemž čas na zpracování je t_i a pravděpodobnost nalezení informace p_i . Chceme rozhodnout jakých maximálně m zdrojů zvolit, abychom minimalizovali cenu, čas a pravděpodobnost nenalezení informace. Pokud binární proměnná x_i bude reprezentovat zda zvolíme i -tý zdroj či ne, pak úloha je 3-kritériální problém batohu:

$$\min (c^T x, \max_i t_i x_i, \prod_i (1 - p_i x_i)) \quad \text{za podm. } e^T x \leq m, x \in \{0, 1\}^n. \quad \square$$

Kapitola 4

Další témata

4.1 Kuželová dominance

Doposud jsme porovnávali řešení z hlediska dominance tak, že vektor $y_1 \in \mathbb{R}^s$ dominuje vektoru $y_2 \in \mathbb{R}^s$ pokud $y_1 \leq y_2$ a $y_1 \neq y_2$. Pohled na tuto dominanci lze zobecnit.

Buď $\mathcal{K} \subseteq \mathbb{R}^s$ konvexní kužel, tedy platí dvě podmínky:

- (1) pro každé $\alpha \geq 0$ a $x \in \mathcal{K}$ je $\alpha x \in \mathcal{K}$,
- (2) pro každé $x, y \in \mathcal{K}$ je $x + y \in \mathcal{K}$.

Dále se často předpokládá, že kužel je bodový (neobsahuje přímku) a má plnou dimenzi.

Definice 4.1. Vektor y_1 *dominuje při kuželi* \mathcal{K} vektoru y_2 pokud $y_1 - y_2 \in \mathcal{K}$ a $y_1 \neq y_2$.

Často se uvažují polyedrické kužele, které jsou popsány

$$\mathcal{K} = \{y \in \mathbb{R}^s; Cy \leq 0\},$$

kde $C \in \mathbb{R}^{m \times s}$ je daná matice. Potom vektor y_1 dominuje při vektoru y_2 pokud $C(y_1 - y_2) \not\leq 0$. Klasická dominance je tudíž polyedrická kuželová dominance s maticí $C = I_s$.

Kuželová dominance umožňuje větší flexibilitu na poli eficientních řešení. Pokud kužel zmenšíme, množina nedominovaných řešení se nezmenší a naopak. Viz obrázek 4.1.

Pro kuželovou dominanci již nestačí uvažovat kladné vážené součty, ale skalarizace má trochu jiný tvar. K tomu je potřeba zavést pojem polárního kužele.

Definice 4.2. Buď $\mathcal{K} \subseteq \mathbb{R}^s$ kužel. Pak *polární kužel* ke \mathcal{K} je kužel

$$\mathcal{K}^* = \{y \in \mathbb{R}^n; y^T x \leq 0 \forall x \in \mathcal{K}\}.$$

Čím větší je kužel, tím menší je polární a naopak. Pro skalarizaci pak polární kužele vstoupí do hry následujícím způsobem.

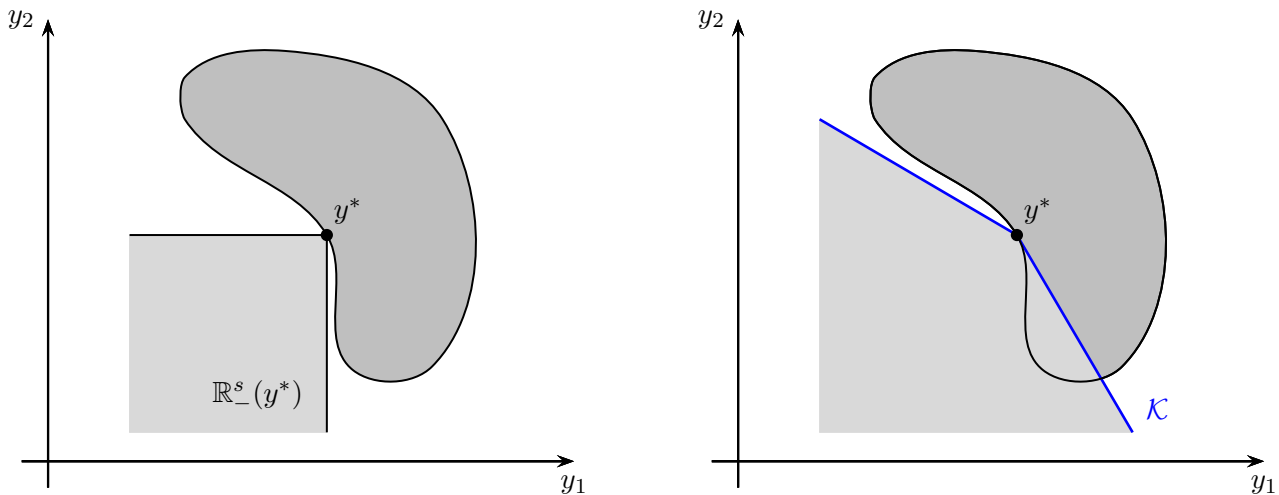
Věta 4.3.

- (1) Buď $\lambda \in \mathcal{K}^*$. Je-li řešení skalarizace $\min_{x \in M} \lambda^T f(x)$ jednoznačné, pak je kuželově nedominovaným řešením.
- (2) Buď $\lambda \in \text{int } \mathcal{K}^*$. Pak řešení skalarizace $\min_{x \in M} \lambda^T f(x)$ je kuželově nedominovaným řešením.

Důkaz. Dokážeme jen první část. Pro spor předpokládejme, že jednoznačné optimální řešení x^* skalarizace je dominováno vektorem $x \in M$. To znamená, že $0 \neq f(x) - f(x^*) \in \mathcal{K}$. Z definice polárního kužele pak $\lambda^T (f(x) - f(x^*)) \leq 0$, neboli $\lambda^T f(x) \leq \lambda^T f(x^*)$, což je spor s jednoznačností a optimalitou x^* . \square

Podotkněme, že pro klasickou dominanci první část věty dává přesně znění věty 1.14, a druhá část přesně větu 1.9.

Nyní se pozastavme u polárních kuželů pro polyedrické kužele. Platí následující charakterizace.



Obrázek 4.1: Kuželová dominance pro různé kužele. Bod y^* je eficientní v klasickém pojetí (nalevo), ale neeficientní pro kužel \mathcal{K} (napravo).

Věta 4.4. *Je-li*

$$\mathcal{K} = \{y \in \mathbb{R}^s; Cy \leq 0\},$$

pak

$$\mathcal{K}^* = \{C^T x \in \mathbb{R}^s; x \geq 0\}.$$

Důkaz. „ \supseteq “ Buď $y \in \mathcal{K}$ a necht' z splňuje $z = C^T x$, $x \geq 0$. Pak $y^T z = y^T C^T x = (Cy)^T x \leq 0$.

„ \subseteq “ Buď $z \in \mathcal{K}^*$, tj. platí

$$Cy \leq 0 \Rightarrow y^T z \leq 0.$$

Podle Farkasova lemmatu ekvivalentně existuje x , které je řešením soustavy $C^T x = z$, $x \geq 0$. □

Je-li speciálně C regulární matice, pak polární kužel lze jednoduše popsat v nerovnicovém tvaru.

Věta 4.5. *Je-li*

$$\mathcal{K} = \{y \in \mathbb{R}^s; Cy \leq 0\},$$

kde C je regulární, pak

$$\mathcal{K}^* = \{z \in \mathbb{R}^s; C^{-T} z \geq 0\}.$$

Důkaz. Upravíme

$$\begin{aligned} \mathcal{K}^* &= \{z \in \mathbb{R}^s; z = C^T x, x \geq 0\} \\ &= \{z \in \mathbb{R}^s; x = C^{-T} z \geq 0\}. \end{aligned}$$

□

Poznámka 4.6. Polární kužele hrají důležitou roli i v klasickém jednokriteriálním lineárním programování. Kdy je daný bod x^* optimálním řešením úlohy $\max_{x \in M} c^T x$? Je to právě tehdy, když vektor c leží v polárním kuželi k tečnému kuželi množiny M v bodu x^* . Jinými slovy, když c je nezápornou kombinací normál z aktivních podmínek.

Poznámka 4.7. Víme již, že klasická eficeience se shoduje s kuželovou při použití kužele \mathbb{R}^s_- . Zajímavý je kuželový pohled na vlastní eficeience [Henig, 1982]. Platí totiž, zhruba řečeno, že řešení je vlastní eficientní právě tehdy, když je kuželově eficientní pro nějaký konvexní uzavřený kužel \mathcal{K} splňující $\text{int } \mathcal{K} \supseteq \mathbb{R}^s_- \setminus \{0\}$.

4.2 Užítková funkce

Přestože vícekritériální úloha $\min_{x \in M} f(x)$ obsahuje s kritérií, teorie užítkové funkce předpokládá, že existuje jakási globální užítková funkce $U: \mathbb{R}^s \rightarrow \mathbb{R}$ uživatele, která agreguje všechna kritéria dohromady. Pak by stačilo vyřešit jednokritériální optimalizační úlohu (pro jednotnost uvažujeme úlohu jako minimalizační, i když by bylo logické spíš užítkovou funkci maximalizovat):

$$\min U(f(x)) \quad \text{za podm. } x \in M. \quad (4.1)$$

Problém však spočívá v tom, že užítkovou funkci neznáme a těžko se aproximuje. Skalarizace váženými součty lze z tohoto úhlu nahlížet jako na užítkovou funkci $U(y) = \lambda^T y$, viz též přístup ze sekce 2.3.1.

Pokud užítková funkce splňuje určité předpoklady, pak (4.1) dává eficientní řešení.

Věta 4.8. *Bud' $U: \mathbb{R}^s \rightarrow \mathbb{R}$ rostoucí v každé složce. Pak každé optimální řešení úlohy (4.1) je eficientní.*

Důkaz. Sporem nechť optimum x^* je dominováno vektorem x , tj. $f(x) \not\leq f(x^*)$. Protože U je rostoucí, $U(f(x)) < U(f(x^*))$, což je spor s optimalitou x^* . \square

Příklad 4.9. Na množině $y \in [1, 10]^2$ mohou užítkové funkce vypadat například jako:

- $U(y) = y_2^2 y_2^3$,
- $U(y) = \log(y_1) + \log(y_2)$,
- $U(y) = \frac{\sqrt{y_1}}{11 - y_2}$. \square

4.3 Intervalové vícekritériální programování

Uvažujme úlohu lineárního vícekritériálního programování

$$\min Cx \quad \text{za podm. } x \in M, \quad (4.2)$$

kde $M = \{x \in \mathbb{R}^n; Ax \leq b\}$. Dále předpokládejme, že matice C není známa přesně, ale může se pohybovat v rozmezí daným intervalovou maticí $C = [\underline{C}, \overline{C}]$. To znamená, že každá složka c_{ij} může nabývat libovolnou hodnotu z c_{ij} nezávisle na ostatních složkách.

Co se pak rozumí (eficientním) řešením takové intervalové úlohy? Přirozeně rozlišujeme dva koncepty.

Definice 4.10. Vektor $x^* \in M$ je *potenciálně eficientní*, pokud je eficientním řešením úlohy (4.2) pro alespoň jedno $C \in \mathcal{C}$. Vektor $x^* \in M$ je *nutně eficientní*, pokud je eficientním řešením úlohy (4.2) pro všechna $C \in \mathcal{C}$.

Označme jako \mathcal{E}_P množinu všech potenciálně eficientních řešení a jako \mathcal{E}_N množinu všech nutně eficientních řešení. Dále označme jako $\mathcal{E}(C)$ množinu eficientních řešení úlohy (4.2) pro dané C . Potom můžeme psát

$$\mathcal{E}_P = \bigcup_{C \in \mathcal{C}} \mathcal{E}(C), \quad \mathcal{E}_N = \bigcap_{C \in \mathcal{C}} \mathcal{E}(C).$$

V následujících dvou sekcích se podrobněji podíváme na tyto dva koncepty řešení.

4.3.1 Potenciálně eficientní řešení

Věta 4.11 (Inuiguchi and Sakawa [1996]). *Testování, zda dané řešení $x^* \in M$ je potenciálně eficientní, je polynomiální problém.*

Důkaz. Označme jako $I = \{i; A_{i*}x^* = b_i\}$ množinu aktivních podmínek a jako A_I, b_I restriktce matic A, b na řádky indexované I . Vektor x^* je optimálním řešením lineárního programu $\min_{x \in M} c^T x$ pokud existuje přípustné řešení duálního systému $-c = A^T v, v \geq 0$ a platí podmínky optimality $v^T(b - Ax^*) = 0$. Z těchto podmínek dostaneme, že $v_i = 0$ pro $i \notin I$, což vede pouze na testování přípustnosti systému

$$-c = A_I^T v, \quad v \geq 0.$$

Eficientní řešení úlohy (4.2) jsou právě optimální řešení skalarizací vzniklých váženým součtem s kladnými váhami (věta 1.24). Tudíž x^* je efficientní pro (4.2) pokud je soustava

$$-C^T \lambda = A_I^T v, \quad v \geq 0, \quad \lambda \geq e$$

řešitelná. Protože C může nabývat jakoukoli hodnotu z \mathbf{C} , tak $C^T \lambda$ může nabývat jakoukoli hodnotu z $[\underline{C}^T \lambda, \overline{C}^T \lambda]$. Tedy x^* je potenciálně efficientní pokud je soustava

$$\underline{C}^T \lambda \leq -A_I^T v \leq \overline{C}^T \lambda, \quad v \geq 0, \quad \lambda \geq e \quad (4.3)$$

řešitelná. □

Pokud je řešení $x^* \in M$ je potenciálně efficientní, pak snadno najdeme i matici $C \in \mathbf{C}$, pro kterou je řešení x^* efficientní, tj. $x^* \in \mathcal{E}(C)$. Stačí vzít jakékoli řešení v^*, λ^* soustavy (4.3) a dopočítat matici $C \in \mathbf{C}$ tak, aby $-A_I^T v^* = C^T \lambda^*$; typicky existuje nekonečně mnoho takových matic C .

Věta 4.12 (Hladík [2017]). *Rozhodovací problém $\mathcal{E}_P \neq \emptyset$ je polynomiální. Pokud $\mathcal{E}_P \neq \emptyset$, pak i nalezení bodu $x^* \in \mathcal{E}_P$ je polynomiální problém.*

Důkaz. Předpokládejme, že $M \neq \emptyset$, což se ověří snadno. Buď $C \in \mathbf{C}$ a nejprve budeme hledat vektor vah $\lambda \geq e$ takový, aby skalarizace $\min_{x \in M} \lambda^T C x$ měla optimální řešení. To ekvivalentně znamená, že duální úloha má přípustné řešení, tedy soustava

$$-\lambda^T C = A^T v, \quad v \geq 0, \quad \lambda \geq e$$

proměnných v, λ má řešení. Protože matice C se pohybuje v rozmezí $\mathbf{C} = [\underline{C}, \overline{C}]$, vektor $\lambda^T C$ tak může nabýt jakékoli hodnoty v intervalu $[\lambda^T \underline{C}, \lambda^T \overline{C}]$. To nás vede na soustavu

$$\lambda^T \underline{C} \leq -A^T v \leq \lambda^T \overline{C}, \quad v \geq 0, \quad \lambda \geq e.$$

Pokud soustava nemá řešení, tak $\mathcal{E}_P = \emptyset$. Naopak, je-li v, λ řešení této soustavy, pak úloha $\min_{x \in M} \lambda^T C x$ má optimální řešení, a to navíc náleží do \mathcal{E}_P . Tento postup tedy zároveň dává i návod jak bod $x^* \in \mathcal{E}_P$ efektivně najít. □

Věta 4.13. *Množinu \mathcal{E}_P tvoří sjednocení několika na sebe navazujících stěn.*

Důkaz. Pro každé $C \in \mathbf{C}$ je množina efficientních řešení $\mathcal{E}(C)$ úlohy (4.2) tvořena sjednocením několika na sebe navazujících stěn (věta 1.28). Tudíž i \mathcal{E}_P tvoří sjednocení několika stěn. Souvislost \mathcal{E}_P se ukáže podobně jako ve větě 1.28. Pro každé $x_1, x_2 \in \mathcal{E}_P$ existuje $\lambda_1, \lambda_2 > 0$ a $C_1, C_2 \in \mathbf{C}$ takové, že x_i je optimum $\min_{x \in M} \lambda_i^T C_i x$, $i = 1, 2$. Postupnou změnou normálového vektoru $\lambda_1^T C_1$ na vektor $\lambda_2^T C_2$ a za použití teorie parametrického programování dostanu souvislost \mathcal{E}_P . □

V klasickém vícekritériálním programování máme dominanci reprezentovanou relací $Cy \preceq 0$. Má tato relace analogii v intervalovém programování? Ano, a je jí relace $C_c y + C_\Delta |y| \preceq 0$, kde

$$C_c = \frac{1}{2}(\underline{C} + \overline{C}), \quad C_\Delta = \frac{1}{2}(\overline{C} - \underline{C})$$

je matice středů a poloměrů intervalové matice \mathbf{C} . Je zajímavé, že relace $C_c y + C_\Delta |y| \preceq 0$ popisuje konvexní polyedrický kužel.

Lemma 4.14 (Rohn and Kreslová [1994]). *Následující je ekvivalentní*

- (1) $\forall C \in \mathbf{C} \exists x \in \mathbb{R}^n : Cx \leq d$,
- (2) $\exists x^1, x^2 \in \mathbb{R}^n : \overline{C}x^1 - \underline{C}x^2 \leq d, x^1, x^2 \geq 0$,
- (3) $\exists x \in \mathbb{R}^n \forall C \in \mathbf{C} : Cx \leq d$,
- (4) $\exists x \in \mathbb{R}^n : C_c x + C_\Delta |x| \leq d$.

Důkaz. „(1) \Rightarrow (2)“ Je-li (2) neřešitelné, potom podle Farkasova lemmatu je řešitelný systém

$$\overline{C}^T y \geq 0, \quad -\underline{C}^T y \geq 0, \quad d^T y \leq -1, \quad y \geq 0,$$

neboli

$$\overline{C}^T y \geq 0 \geq \underline{C}^T y, \quad d^T y \leq -1, \quad y \geq 0.$$

Tím pádem existuje $C \in \mathcal{C}$ taková, že

$$C^T y = 0, \quad d^T y \leq -1, \quad y \geq 0.$$

Podle Farkasova lemmatu opět je soustava $Cx \leq d$ neřešitelná.

„(2) \Rightarrow (3)“ Definujme $x^* := x^1 - x^2$. Pro libovolné $C \in \mathcal{C}$ pak je

$$Cx^* = C(x^1 - x^2) = Cx^1 - Cx^2 \leq \overline{C}x^1 - \underline{C}x^2 \leq d.$$

„(3) \Rightarrow (1)“ Zřejmé.

„(3) \Leftrightarrow (4)“ Platí proto, že největší hodnota výrazu Cx přes všechna možná $C \in \mathcal{C}$ je $C_c x + C_\Delta |x|$. To nahlédneme tak, že pro každé $C \in \mathcal{C}$ je

$$Cx = C_c x + (C - C_c)x \leq C_c x + |C - C_c| \cdot |x| \leq C_c x + C_\Delta |x|.$$

Tato horní mez se pak nabyde pro matici $C \in \mathcal{C}$ definovanou $c_{ij} = \overline{c}_{ij}$ pokud $x_j \geq 0$ a $c_{ij} = \underline{c}_{ij}$ jinak. \square

Věta 4.15. *Následující je ekvivalentní pro bod $x^* \in M$:*

- (1) $\forall C \in \mathcal{C} \exists x \in M : C(x - x^*) \not\leq 0$, (t_j , $x^* \notin \mathcal{E}_P$),
- (2) $\exists x \in M \forall C \in \mathcal{C} : C(x - x^*) \not\leq 0$,
- (3) $\exists x \in M : C_c(x - x^*) + C_\Delta |x - x^*| \not\leq 0$.

Důkaz. Vyplývá modifikací lemmatu 4.14. \square

4.3.2 Nutně eficientní řešení

Nutně eficientní řešení mají z řady pohledů komplementární vlastnosti oproti potenciálně eficientním. Množinu všech nutně eficientních řešení je tvořena sjednocením stěn M (protože se dá vyjádřit jako průnik eficientních množin všech realizací (4.2)), ale nemusí být souvislá.

Příklad 4.16. Buď $M = \{x \in \mathbb{R}^2; \|x\|_1 \leq 1\}$ a

$$C = \begin{pmatrix} 1 & [-\varepsilon, \varepsilon] \\ -1 & [-\varepsilon, \varepsilon] \end{pmatrix},$$

kde $\varepsilon > 0$ je malé. Pro realizaci $C = \begin{pmatrix} 1 & \varepsilon \\ -1 & \varepsilon \end{pmatrix}$ je množinou eficientních řešení sjednocení hran $(-1, 0) - (0, -1)$ a $(0, -1) - (1, 0)$. Naopak pro realizaci $C = \begin{pmatrix} 1 & -\varepsilon \\ -1 & -\varepsilon \end{pmatrix}$ je množinou eficientních řešení sjednocení hran $(-1, 0) - (0, 1)$ a $(0, 1) - (1, 0)$. Při určitých realizacích jako je například $C = \begin{pmatrix} 1 & 0 \\ -1 & 0 \end{pmatrix}$ je eficientní celé M . Jiný případ nenastane, a proto průnikem těchto množin dostaneme dva izolované body

$$\mathcal{E}_N = \{(-1, 0)\} \cup \{(1, 0)\}. \quad \square$$

Následující tvrzení charakterizuje, kdy je dané řešení nutně eficientní.

Věta 4.17 (Bitran [1980]). *Řešení $x^* \in M$ je nutně eficientní právě tehdy, když je eficientní pro (4.2) s $C := C_c - C_\Delta \text{diag}(s)$ pro všechna $s \in \{\pm 1\}^n$.*

Důkaz. Jedna implikace je triviální a druhou dokážeme sporem. Necht' $x^* \in M$ není nutně eficientní. Pak existuje $C \in \mathcal{C}$ a $x \in M$, které mu dominuje, to jest $C(x - x^*) \not\leq 0$.

$$0 \not\leq C(x - x^*) \geq C_c(x - x^*) - C_\Delta |x - x^*| = (C_c - C_\Delta \text{diag}(s))(x - x^*),$$

kde $s := \text{sgn}(x - x^*)$. To je ale spor s eficientí x^* vůči $C = C_c - C_\Delta \text{diag}(s)$. \square

Charakterizace vyžaduje 2^n testů klasické efciencie a je tedy exponenciální vzhledem k počtu proměnných, nikoli k počtu kritérií. Obecně lze stěží získat rychlejší test, protože rozhodování o nutné efcienci je výpočetně složité.

Věta 4.18 (Hladík [2012]). *Testování zda $x^* \in \mathcal{E}_N$ je co-NP-těžké i při jednom kritériu.*

Důkaz. Uvažujme NP-těžký SET-PARTITIONING problém: Dána čísla $a_1, \dots, a_n \in \mathbb{Z}$ a otázka zní, zda existují $x_1, \dots, x_n \in \{\pm 1\}$ takové, že $\sum_{i=1}^n a_i x_i = 0$. Úlohu můžeme ekvivalentně vyjádřit jako řešitelnost intervalové soustavy

$$a^T x = 0, \quad -e \leq x \leq e, \quad c^T x \leq -n$$

pro nějaké $c \in [-e, e]$. Právě poslední nerovnice vynutí podmínku $x \in \{\pm 1\}^n$. Protože koeficienty a_1, \dots, a_n jsou celočíselné, tak existuje $\varepsilon > 0$ polynomiální velikosti tak, že soustava je ekvivalentní co do řešitelnosti soustavě

$$a^T x = 0, \quad -e \leq x \leq e, \quad c^T x \leq -n + 2\varepsilon. \quad (4.4)$$

Intuitivně: pokud $|x| \approx e$, tak aby byla splněná rovnice $a^T x = 0$, musí platit $|x| = e$.

Nyní ukážeme, že soustava (4.4) je ekvivalentní soustavě

$$a^T x = 0, \quad -ez \leq x \leq ez, \quad c^T x + (n - 2\varepsilon)z \leq -\varepsilon. \quad (4.5)$$

Pokud (4.4) je řešitelná, pak existuje řešení splňující $|x| = e$, a tudíž x spolu se $z = 1$ je řešením (4.5). Naopak předpokládejme, že x, z je řešením (4.5). Protože $(x, z) \neq (0, 0)$, musí $z > 0$. Substitucí $x := \frac{1}{z}x$ získá (4.5) tvar

$$a^T x = 0, \quad -e \leq x \leq e, \quad c^T x \leq -n + 2\varepsilon - \varepsilon/z \leq -n + 2\varepsilon.$$

Tím pádem x je řešením (4.4).

Neřešitelnost soustavy (4.5) pro všechna $c \in [-e, e]$ je ekvivalentní nutné efcienci bodu $x^* = 0$ pro lineární program s podmínkami

$$a^T x \leq 0, \quad -a^T x \leq 0, \quad -ez \leq x \leq ez$$

a jednou účelovou funkcí $c^T x + (n - 2\varepsilon)z$, kde $c \in \mathbf{c} = [-e, e]$. □

Důkaz lze trochu zjednodušit, pokud budeme uvažovat dvě kritéria, avšak s intervaly pouze v jednom z nich.

Věta 4.19. *Testování zda $x^* \in \mathcal{E}_N$ je co-NP-těžké i při dvou kritériích.*

Důkaz. Uvažujme NP-těžký SET-PARTITIONING problém: Dána čísla $a_1, \dots, a_n \in \mathbb{Z}$ a otázka zní, zda existují $x_1, \dots, x_n \in \{\pm 1\}$ takové, že $\sum_{i=1}^n a_i x_i = 0$. Úlohu můžeme ekvivalentně vyjádřit jako řešitelnost intervalové soustavy

$$a^T x = 0, \quad -e \leq x \leq e, \quad c^T x \leq -n$$

pro nějaké $c \in [-e, e]$. Řešitelnost této soustavy je ekvivalentní řešitelnosti intervalové soustavy

$$a^T x = 0, \quad -ez \leq x \leq ez, \quad c^T x \leq -nz, \quad 0 < z \leq 1. \quad (4.6)$$

Nyní tvrdíme, že neřešitelnost soustavy (4.6) pro všechna $c \in [-e, e]$ je ekvivalentní nutné efcienci bodu $(x^*, z^*) = (0, 0)$ pro lineární program s podmínkami

$$a^T x \leq 0, \quad -a^T x \leq 0, \quad -ez \leq x \leq ez, \quad z \leq 1.$$

a se dvěma účelovými funkcemi $f_1(x, z) = -z$ a $f_2(x, z) = c^T x + nz$, kde $c \in \mathbf{c} = [-e, e]$. Pokud má soustava (4.6) řešení (x, z) , pak

$$f_1(x, z) < 0 = f_1(x^*, z^*), \quad f_2(x, z) \leq 0 = f_2(x^*, z^*).$$

Naopak, pokud přípustný bod (x, z) dominuje bodu (x^*, z^*) pro určité $c \in \mathbf{c}$, pak nutně $z > 0$, tudíž $f_2(x, z) \leq f_2(x^*, z^*) = 0$ a $f_1(x, z) < f_1(x^*, z^*) = 0$. Proto (x, z) vyhovuje soustavě (4.6). □

Kužel dominujících směrů nutné eficiency je podobný jako pro potenciální eficiency, ale tvoří sice polyedrický, ale obecně nekonvexní, kužel.

Věta 4.20. *Bod $x^* \in M$ je nutně eficientní právě tehdy, když neexistuje $x \in M$ takové, že*

$$C_c(x - x^*) - C_\Delta|x - x^*| \not\leq 0.$$

Důkaz. Platí $x^* \notin \mathcal{E}_N$ právě tehdy, když existuje $C \in \mathbf{C}$ a $x \in M$ takové, že $C(x - x^*) \not\leq 0$. Nejmenší hodnota výrazu $C(x - x^*)$ přes všechny možné $C \in \mathbf{C}$ je $C_c(x - x^*) - C_\Delta|x - x^*|$, a to ve všech složkách. Tudíž dostáváme podmínku $C_c(x - x^*) - C_\Delta|x - x^*| \not\leq 0$. \square

Kapitola 5

Vícekriteriální rozhodování

Ve vícekriteriálním rozhodování rovněž porovnááme různá přípustná řešení z hlediska více kritérií, ale modely, kterými se zabývá, již nespádají do klasického rámce vícekriteriální optimalizace.

5.1 DEA

Metoda *DEA* (data envelopment analysis) byla vyvinuta v 70-tých letech na vyhodnocení vzdělávacích programů amerických veřejných škol, a ukázala se i jako užitečný nástroj pro porovnávání dalších institucí (škol, nemocnic atp.), obchodních řetězců, bank atd. Těmto porovnávaným objektům se v oboru říká DMU (decision making unit). Název DEA i DMU pochází od Charnes et al. [1978], viz též knihy Cooper et al. [2007]; Zhu [2016].

Tradičně se výkonnost či účinnost měří podílem

$$\frac{\text{výstup}}{\text{vstup}},$$

kdy poměříme množství výstupu (výroba, ...) ku množství vstupu (spotřeba, ...). Podíl tedy dává hodnotu v intervalu $[0, 1]$, kdy 0 je nejhorší (všechno se spotřebuje a nic nevyrobí) a 1 nejlepší (100% účinnost). V některých ekonometrických modelech může mít smysl uvažovat účinnost větší než 1, ale například fyzikální účinnost stroje jako poměr energie vydané/vykonané a energie dodané je vždy menší než 1.

Uvažujme situaci, kdy máme více vstupů i výstupů. Například při hodnocení nemocnic lze jako vstup uvažovat finanční příjem, počet doktorů a sester, a jako výstup například počet pacientů, vyléčených pacientů či scientometrický údaj vědecké činnosti.¹⁾ Relativní výkonnost k -tého subjektu můžeme měřit opět podobným podílem, kdy vezmeme vážený součet vstupů ve jmenovateli a vážený součet výstupů v čitateli. Matematicky

$$\frac{\sum_{i=1}^m u_i y_i^k}{\sum_{j=1}^n v_j x_j^k},$$

kde $u_j, v_j \geq 0$ jsou váhy, x_1^k, \dots, x_n^k měří vstupy a y_1^k, \dots, y_m^k výstupy k -tého subjektu (DMU).

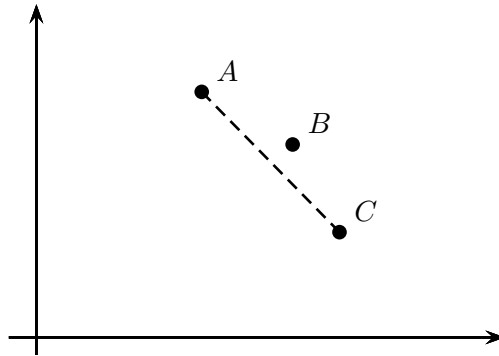
Určit váhy může být obtížné. Metoda DEA je založena na tom, že najde váhy, při kterých má k -tý subjekt nejlepší hodnotu relativní výkonnosti. Aby byla úloha dobře definována, při stejných vahách omezíme výkonnosti ostatních subjektů $\ell \neq k$ jedničkou. Matematická formulace problému pak je

$$\max \frac{\sum_{i=1}^m u_i y_i^k}{\sum_{j=1}^n v_j x_j^k}, \quad \text{za podm.} \quad \frac{\sum_{i=1}^m u_i y_i^\ell}{\sum_{j=1}^n v_j x_j^\ell} \leq 1, \quad \forall \ell \neq k, \quad u_i, v_j \geq 0.$$

Protože jsou výrazy pozitivně homogenní v proměnných, můžeme jmenovatel účelové funkce normovat a celou optimalizační úlohu přepsat na lineární program

$$\max \sum_{i=1}^m u_i y_i^k \quad \text{za podm.} \quad \sum_{j=1}^n v_j x_j^k = 1, \quad \sum_{j=1}^m u_j y_j^\ell \leq \sum_{j=1}^n v_j x_j^\ell, \quad \forall \ell \neq k, \quad u_i, v_j \geq 0.$$

¹⁾Rozdělení na vstupy a výstupy není vždy jednoznačné. V tomto příkladu může počet pacientů být považován i jako vstup.



Obrázek 5.1: (Příklad 5.1) DEA se dvěma eficientními subjekty A a C a jedním neeficientním subjektem B.

Maticově zapsáno má lineární program tvar

$$\max y^T u \quad \text{za podm.} \quad x^T v = 1, \quad Y u - X v \leq 0, \quad u, v \geq 0, \quad (5.1)$$

kde $x = (x_1^k, \dots, x_n^k)^T$, $y = (y_1^k, \dots, y_m^k)^T$, $X_{\ell j} = x_j^\ell$, $Y_{\ell i} = y_i^\ell$.

Jaká je interpretace výsledků? Optimální hodnoty těchto úloh nám dají výkonnosti jednotlivých subjektů. Je-li optimální hodnota alespoň 1, pak je daný subjekt eficientní (dost často se uvádí jen eficeience rovna 1). V opačném případě je neeficientní a optimální hodnota měří míru neeficience; DMU má tedy v určitých směrech rezervy na zlepšení výkonnosti.

Do modelu lze ještě přidávat další omezující podmínky, například, že první váha není víc než pětikrát důležitější než druhá váha ($v_1 \leq 5v_2$). Z modelu je také vidět, že subjekt, který striktně dominuje v jednom kritériu, je vždy eficientní, bez ohledu na to jak špatně si vede v ostatních kritériích. Těto situaci jde také předcházet stanovením vhodných omezení na váhy (minimální resp. maximální hodnota).

Příklad 5.1. Uvažujme tři stavební společnosti A, B a C. Stavba objektu

společností A trvá 10 dní a vyžaduje 200 jednotek peněz,
 společností B trvá 13 dní a vyžaduje 160 jednotek peněz,
 společností C trvá 15 dní a vyžaduje 120 jednotek peněz.

Společnost A je zřejmě eficientní, protože je nejlepší v prvním kritériu. Stačí uvažovat vážený součet s dost velkou vahou pro první kritérium, např. $(100, 1)$, a společnost A dosahuje nejnižší hodnoty 1200, v porovnání s hodnotami 1460 a 1620 pro ostatní společnosti. Podobně společnost C je eficientní, protože je nejlepší v druhém kritériu. Výkonnost společnosti B určíme vyřešením odpovídající úlohy. Výstup je zde jediný (výsledná stavba), zatímco vstupem je délka stavby a finanční náklad. Úloha pak má tvar

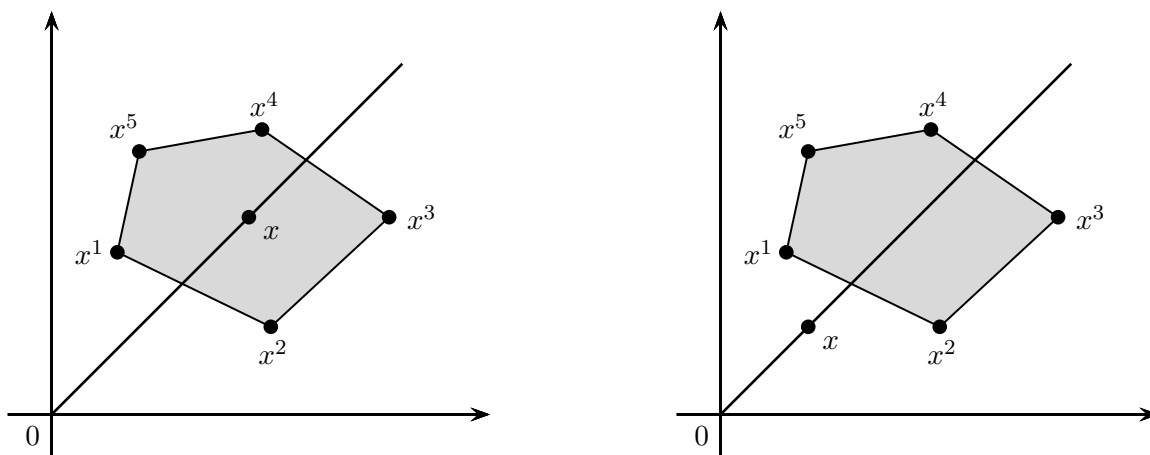
$$\begin{aligned} &\max u_1 \\ &\text{za podm.} \quad 13v_1 + 160v_2 = 1, \quad u_1 \leq 10v_1 + 200v_2, \quad u_1 \leq 15v_1 + 120v_2, \quad u_1, v_1, v_2 \geq 0. \end{aligned}$$

Optimální hodnota je ≈ 0.978 , čili společnost B není zcela eficientní. Výsledné výkonnosti společností A, B a C jsou 1, 0.978 a 1.

Interpretace může být i taková, že společnosti B dominuje konvexní kombinace společností A a C. Obrázek 5.1 znázorňuje graficky výkon jednotlivých společností, eficientní jsou ty vlevo dole, tj. jsou optimem při lineárním kritériu s nezápornými vahami. Jsou to vlastně klasická nedominovaná řešení, pokud množinu bodů rozšíříme na jejich konvexní obal. \square

Příklad nabízí zajímavou geometrickou interpretaci eficientních řešení. Uvažujme duální úlohu k úloze (5.1)

$$\min z \quad \text{za podm.} \quad xz - X^T w \geq 0, \quad Y^T w \geq y, \quad w \geq 0.$$



Obrázek 5.2: Geometrická interpretace eficientních subjektů v DEA. Nalevo neeficientní subjekt (polopřímka protne polytop před bodem x) a napravo eficientní subjekt (polopřímka protne polytop za bodem x).

Předpokládejme, že výstup je jednorozměrná veličina jednotkové velikosti, tedy $y = 1$, $Y = (1, \dots, 1)^T$. Úloha má tvar

$$\min z \quad \text{za podm.} \quad xz - X^T w \geq 0, \quad e^T w \geq 1, \quad w \geq 0,$$

který je ekvivalentní tvaru

$$\min z \quad \text{za podm.} \quad xz - X^T w \geq 0, \quad e^T w = 1, \quad w \geq 0.$$

Úloha hledá průnik polopřímky, vycházející z počátku ve směru x , s konvexním obalem sloupců matice X^T , což jsou vstupy jednotlivých subjektů. Daný subjekt je pak eficientní, pokud je průnik za bodem x nebo v něm. Subjekt je neeficientní v opačném případě. Geometricky tato neeficience znamená, že stejného výstupu dosáhneme s menšími vstupy pomocí konvexní kombinace ostatních subjektů (jako bychom z nich složili virtuální nový subjekt, který bude ve všech vstupech lepší). Viz obrázek 5.2. Pokud tedy v prostoru \mathbb{R}^n nakreslíme body $x^k := (x_1^k, \dots, x_n^k)^T$, $k = 1, \dots, n$, tedy vstupy jednotlivých (ale všech) subjektů, pak eficientní budou ty umístěné vlevo dole a na hranici jejich konvexního obalu.

5.2 AHP

Metoda AHP (Analytic Hierarchy Process, Thomas L. Saaty, 1980) je metoda, která slouží k vytváření vah určitým objektům v rozhodovací úloze, která je komplikovaná sítí vztahů. Metoda spočívá v rozdělení problémů na několik úrovní, na každé úrovni určení vah pomocí porovnávací matice a nakonec agregace vah dohromady. Popíšeme jednotlivé kroky. Více podrobností a aplikací viz Saaty [2008]; Saaty and Vargas [2012], zobecnění na modely ANP (Analytic Network Process) s komplikovanější závislostí viz Saaty and Vargas [2013] nebo novější analýzu konsistence [Mazurek, 2023].

Porovnávací matice

Uvažujme n objektů, kterým chceme přiřadit váhy. Uživatel to nedokáže explicitně, ale umí porovnávat objekty mezi sebou. Vytvoří tedy matici $A \in \mathbb{R}^{n \times n}$, kde a_{ij} určuje, kolikrát je i -tý objekt důležitější než j -tý. Přirozeně platí $a_{ii} = 1$ a $a_{ji} = 1/a_{ij}$, čili stačí zadat $\binom{n}{2}$ hodnot.

Konsistence. V ideálním případě by měla platit vlastnost $a_{ik}a_{kj} = a_{ij}$, tzv. *konsistence*, a matice měla tvar

$$A = \begin{pmatrix} \frac{v_1}{v_1} & \frac{v_1}{v_2} & \cdots & \frac{v_1}{v_n} \\ \vdots & & & \vdots \\ \frac{v_n}{v_1} & \frac{v_n}{v_2} & \cdots & \frac{v_n}{v_n} \end{pmatrix}.$$

Vidíme, že pak váhy tvoří prvky prvního sloupcečku (po znormování) a matice je konsistentní. To platí i naopak – každá konsistentní matice má výše uvedený tvar. Stačí definovat váhy $v_i := a_{i1}$ pro $i = 1, \dots, n$, a pak $a_{ij} = a_{i1}a_{1j} = a_{i1}/a_{j1} = v_i/v_j$, čili skutečně hodnota a_{ij} je podílem odpovídajících vah.

Konsistentní matice má hodnotu 1, čili 0 je $(n-1)$ -násobné vlastní číslo a n je zbývající maximální vlastní číslo a odpovídající vlastní vektor je v .

Určení vah. Prakticky matice A konsistentní skoro nikdy není. Předchozí úvaha nás vede k tomu určit váhy jako vlastní vektor v matice A , odpovídající největšímu vlastnímu číslu; navíc normujeme $e^T v = 1$. Protože $A > 0$, podle Perronovy věty je největší vlastní číslo reálné kladné a odpovídající vlastní vektor také. Navíc žádnému jinému vlastnímu číslu neodpovídá kladný vlastní vektor.

Vlastní vektor se často aproximuje. Nechť s_j značí součet prvků j -tého sloupce matice A , tedy $s = A^T e$. Nyní každý prvek matice vydělíme hodnotou příslušného součtu a výslednou matici označíme A' , tedy $a'_{ij} = a_{ij}/s_j$. Tato matice pak splňuje $e^T A' = e^T$. Aproximované váhy w pak určíme jako průměrné řádkové součty, tedy $w = \frac{1}{n} A' e$. Zřejmě platí $e^T w = 1$, neboť

$$e^T w = \frac{1}{n} e^T (A' e) = \frac{1}{n} (e^T A') e = \frac{1}{n} e^T e = 1.$$

Odpovídající vlastní číslo λ pak odhadneme jako $\tilde{\lambda} := s^T w$, neboť

$$\tilde{\lambda} = s^T w = e^T A w \approx e^T \lambda w = \lambda.$$

Motivace pro tuto aproximaci je, že pro konsistentní matici má aproximovaný vlastní vektor a aproximované vlastní číslo stejnou hodnotu jako skutečný vlastní vektor a vlastní číslo. Důkaz: Protože $s_j = \sum_{k=1}^n v_k/v_j = 1/v_j$, odvodíme

$$w_i = \frac{1}{n} \sum_{j=1}^n a'_{ij} = \frac{1}{n} \sum_{j=1}^n \frac{v_i}{v_j} \cdot \frac{v_j}{1} = v_i,$$

$$s^T w = \sum_{j=1}^n s_j w_j = \sum_{j=1}^n \frac{v_j}{v_j} = n.$$

Míra konsistence. Prakticky je málokdy matice A konsistentní. Abychom věděli, jak moc blízko má matice A ke konsistenci, zavádíme míru konsistence. Nechť λ je spočtené či aproximované největší vlastní číslo matice A . Ideálně by mělo být $\lambda = n$, obecně platí $\lambda \geq n$ (viz věty 5.2–5.3). Definujme tzv. index konsistence jako

$$CI = \frac{\lambda - n}{n - 1}.$$

Čím menší je CI , tím blíže je λ k hodnotě n a tím blíže jsme ke konsistenci. Pokud $CI = 0$, je matice zcela konsistentní. Hodnotou $n-1$ dělíme kvůli normování: protože součet vlastních čísel matice A je roven součtu diagonály, tedy n , tak $CI = \frac{-1}{n-1} \sum_{i=2}^n \lambda_i$ udává průměrnou hodnotu nedominantního vlastního čísla.

Index CI pak porovnáme s tzv. tabulovaným indexem RI (*random consistency index*). Ten byl určen z náhodně generovaných dat a slouží k tomu, abychom měli hodnotu CI s čím srovnat. Pro ilustraci několik jeho hodnot:

n	1	2	3	4	5	6	7	8	9	10
RI	0	0	0.58	0.9	1.12	1.24	1.32	1.41	1.45	1.49

Pokud $CI/RI < 0.1$, pak bereme míru konsistence za přijatelnou, jinak je matice příliš nekonsistentní a vypočtené váhy mohou být velmi zkreslené.

Věta 5.2. Pro aproximované největší vlastní číslo $\tilde{\lambda}$ matice A platí $\tilde{\lambda} \geq n$.

Důkaz. Platí $\tilde{\lambda} = s^T w = \sum_{i=1}^n s_i w_i = \frac{1}{n} \sum_{i,j=1}^n s_i a_{ij}/s_j = \frac{1}{n} \sum_{i,j=1}^n b_{ij}$, kde matici B definujeme jako $b_{ij} := s_i a_{ij}/s_j$. Protože $b_{ii} = 1$ a $b_{ij} + b_{ji} = b_{ij} + 1/b_{ij} \geq 2$, tak platí $\frac{1}{n} \sum_{i,j=1}^n b_{ij} \geq n$. \square

Věta 5.3. Pro největší vlastní číslo λ matice A platí $\lambda \geq n$.

Důkaz. Buď $v > 0$ vlastní vektor příslušný λ a vytvořme diagonální matici D s vektorem v na diagonále. Definujme matici $B := D^{-1}AD$, která je podobná matici A a tedy má stejná vlastní čísla; je rovněž porovnávací maticí. Vlastní vektor matice B příslušný λ bude e , neboť

$$Be = D^{-1}ADe = D^{-1}Av = D^{-1}\lambda v = \lambda e.$$

Tudíž platí $e^T Be = \lambda e^T e = \lambda n$, z čehož $\lambda = \frac{1}{n} \sum_{i,j=1}^n b_{ij}$. Dále postupujeme jako ve větě 5.2. \square

Odhady z vět nahoře se nabývají jako rovnosti právě pro konsistentní matice. Odhady jsou totiž těsné jen když matice B se skládá ze samých jedniček, což implikuje konsistenci matice A .

Poznámka 5.4. Váhy lze z porovnávací matice určit i jinak. Můžeme uvažovat prostor matic s určitou metrikou a k dané matici A najít co nejbližší konsistentní matici. Jiná možnost je nafouknout matici A na intervalovou matici minimálním způsobem tak, aby obsahovala konsistentní. Statistický postup je uvažovat v matici chyby s vhodným rozdělením a hledat nejvěrohodnější řešení. Tento způsob Saaty zkoušel, ale prý nebyl vhodný a nedával dobré výsledky. Zde je dobré poznamenat, že podle původního návrhu metody uživatel vybírá hodnoty matice A v rozmezí $\{1, 2, \dots, 9\}$, což ale nemá matematické zdůvodnění a nutně vede k nekonsistenci.

Příklad 5.5. Porovnávací matici použila také NASA pro návrh orbitální stanice NASA v rámci programu Orion. Astronauti porovnávali 20 atributů, jako například cirkadiánní osvětlení, vyhlídkové okno, umístění toalet, kuchyňky či jiných zařízení. Více viz Gernhardt et al. [2019].

Hierarchie

Nyní předpokládejme, že máme několik úrovní atributů; pro vysvětlení myšlenky uvažujme jen dvě úrovně. V první úrovni z porovnávací matice přiřadíme atributům váhy w_1, \dots, w_n . Dále, k -tý atribut se skládá z n_k podatributů, kterým opět na základě příslušné porovnávací matice přiřadíme váhy w_{k1}, \dots, w_{kn_k} . Nakonec váhy podatributů přenormujeme $w_{kj} \rightarrow w_k w_{kj}$. Nyní je součet vah všech podatributů rovenjedné, neboť

$$\sum_{k=1}^n \sum_{j=1}^{n_k} w_k w_{kj} = \sum_{k=1}^n w_k \sum_{j=1}^{n_k} w_{kj} = \sum_{k=1}^n w_k = 1.$$

Příklad 5.6. Uvažujme problém výběru auta. V první úrovni vybíráme podle atributů: cena, kapacita, bezpečnost. Porovnávací matice vypadá:

$$A = \begin{pmatrix} 1 & 1/3 & 5 \\ 3 & 1 & 7 \\ 1/5 & 1/7 & 1 \end{pmatrix}.$$

Největší vlastní číslo a jemu odpovídající vlastní vektor jsou

$$\lambda = 3.0649, \quad v = (0.2790, 0.6491, 0.0719)^T.$$

Aproximované vlastní číslo a vlastní vektor jsou

$$\tilde{\lambda} = 3.0067, \quad w = (0.2828, 0.6434, 0.0738)^T.$$

Pro ně odpovídající index konsistence je $CI = 0.0484$, tudíž $CI/IR = 0.083 < 0.1$ a matice může být považovaná za přiměřeně konsistentní.

Nyní uvažujme druhou úroveň, kdy cena se skládá z vlastní ceny auta a ze spotřeby a nechtě je ovážíme vektorem $(0.7, 0.3)^T$. Pak výsledné váhy jsou

$$(0.7w_1, 0.3w_1, w_2, w_3)^T = (0.1980, 0.0848, 0.6434, 0.0738)^T.$$

Celkem jsme tedy od uživatele potřebovali 4 porovnání. Bez rozlišení dvou úrovní a uvažování celé čtveřice naráz by porovnání bylo 6. \square

Příklad 5.7. Jiným příkladem je výběr kandidáta na pracovní pozici. Nechtě Adéla, Bob a Cyril jsou kandidáti a výběr provádíme podle kritérií: zkušenost, vzdělání, osobnost a jazyky. Z porovnávací matice určíme váhy kritérií. Komise pak pro každé kritérium porovná kandidáty pomocí matice a určí příslušné váhy. Váhy pak sečteme a dostaneme výsledné ohodnocení kandidátů. \square

Příklad ze života o tom, jak Saatyho syn Michael vybíral školu ze tří možností a podle šesti kritérií:
<https://www.youtube.com/watch?v=ChkBNabdfjo>

Poznámka 5.8. Výhoda AHP spočívá v rozdělení na jednodušší podproblémy. Tím se zmenší počet porovnání vyžadovaný od uživatele a také zmenší možnost obdržení nekonsistentní matice (větší matice je spíše nekonsistentní). Na druhou stranu, metoda má i řadu záporných stránek. Může docházet k tzv. *rank reversal*, kdy přidáním dodatečného atributu se zcela obrátí pořadí vah; to lze teoreticky zneužít k manipulaci výsledků. Výsledné váhy taky mohou porušovat *podmínku koherence*, která přirozeně požaduje, aby $a_{ij} > 1$ právě tehdy, když $v_i > v_j$.

Kapitola 6

Závěr

Co se sem už nevešlo. V klasické, jedno-kriteriální optimalizaci jsou některé úlohy tak velké a složité (nekonvexní účelová funkce i množina přípustných řešení), že je nelze vyřešit do optima v reálném čase. Využívají se tam proto různé typy heuristik a metaheuristik, jako je genetické programování (nebo obecně přírodou inspirované metody), simulované žíhání aj. Jejich výhoda spočívá v tom, že neuváznou v lokálních minimech a v reálném čase vydávají více či méně dobré řešení. Ve vícekriteriální optimalizaci můžeme také narazit na velké složité úlohy a také pro ně může být výhodné adaptovat výše zmíněné heuristiky. U přírodou inspirovaných heuristik typicky dostaneme na výstupu pestrý vzorek „populace“, tedy soubor přípustných řešení, ze kterých si uživatel může vybrat – to je výhodou zejména pro úlohy typu IV.

Výhledy. Vícekriteriální programování je stále se rozvíjející obor. Mezi aktuální trendy výzkumu patří [Miettinen et al., 2008]:

- speciální úlohy, jako je například vícekriteriální semidefinitní či dynamické programování,
- large-scale úlohy, kdy velký rozměr znamená nejenom počet proměnných a omezení, ale může to být i počet kritérií (design visutých mostů aj.),
- nové trendy v optimalizaci obecně – robustní optimalizace, nejistota v datech

Značení

Množiny a čísla

$\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}$	množina přirozených, celých, racionálních a reálných čísel
$\mathbb{R}^s(y^*)$	kužel dominujících bodů s počátkem v y^* , $\mathbb{R}^s(y^*) = \{y \in \mathbb{R}^s; y \leq y^*\}$
\mathbb{R}_-^s	záporný ortant, $\mathbb{R}_-^s = \mathbb{R}_-^s(0) = \{y \in \mathbb{R}^s; y \leq 0\}$
\mathbb{R}_+^s	kladný ortant, $\mathbb{R}_+^s = \{y \in \mathbb{R}^s; y \geq 0\}$
$U + V$	součet množin, $U + V = \{u + v; u \in U, v \in V\}$
$\text{conv}(M)$	konvexní obal množiny M
$\text{int}(M)$	topologický vnitřek množiny M
\mathcal{K}^*	polární kužel ke kuželi \mathcal{K} , viz definice 4.2
S	množina kladných vah, $S = \{\lambda \in \mathbb{R}^s; \lambda > 0\}$
\overline{S}	množina nezáporných vah, $\overline{S} = \{\lambda \in \mathbb{R}^s; \lambda \geq 0\}$
\mathcal{E}	množina eficientních řešení, strana 8
\mathcal{E}^v	množina vlastních eficientních řešení, strana 9
r^+	kladná část reálného čísla, $r^+ = \max(r, 0)$
r^-	záporná část reálného čísla, $r^- = \max(-r, 0)$

Matice a vektory

A^T	transpozice matice A
$A \geq B$	nezápornost matice $A - B$, tj. $a_{ij} \geq b_{ij}$
$A > B$	kladnost matice $A - B$, tj. $a_{ij} > b_{ij}$
$0_n, 0$	nulová matice (všechny složky jsou rovny 0)
I_n, I	jednotková matice (diagonální s jedničkami na diagonále)
e_i	jednotkový vektor, $e_i = (0, \dots, 0, 1, \dots, 0)^T$
e	vektor ze samých jedniček, $e = (1, \dots, 1)^T$
$\ x\ _p$	p -norma vektoru $x \in \mathbb{R}^n$, $\ x\ _p = (\sum_{i=1}^n x_i ^p)^{\frac{1}{p}}$
$\ x\ _2$	eukleidovská norma vektoru $x \in \mathbb{R}^n$, $\ x\ _2 = \sqrt{\sum_{i=1}^n x_i^2}$
$\ x\ _\infty$	maximová norma vektoru $x \in \mathbb{R}^n$, $\ x\ _\infty = \max_{i=1, \dots, n} x_i $

Literatura

- I. Araya, D. Aliquintui, F. Ardiles, and B. Lobo. Nonlinear biobjective optimization: improving the upper envelope using feasible line segments. *J. Glob. Optim.*, 79(2):503–520, 2021.
<https://doi.org/10.1007/s10898-021-00991-7>. 35
- A. Ben-Tal and A. Nemirovski. *Lectures on Modern Convex Optimization. Analysis, Algorithms, and Engineering Applications*. SIAM, Philadelphia, PA, 2001. 7
- R. Benayoun, J. Montgolfier, J. Tergny, and O. Laritchev. Linear programming with multiple objective functions: Step method (STEM). *Math. Program.*, 1(1):366–375, 1971.
<https://doi.org/10.1007/BF01584098>. 31
- G. R. Bitran. Linear multiple objective problems with interval coefficients. *Manage. Sci.*, 26:694–706, 1980.
<https://doi.org/10.1287/mnsc.26.7.694>. 47
- P. Camerini, G. Galbiati, and F. Maffioli. On the complexity of finding multi-constrained spanning trees. *Discrete Appl. Math.*, 5(1):39–50, 1983.
[https://doi.org/10.1016/0166-218X\(83\)90014-8](https://doi.org/10.1016/0166-218X(83)90014-8). 39
- A. Charnes, W. W. Cooper, and E. Rhodes. Measuring the efficiency of decision making units. *Eur. J. Oper. Res.*, 2(6):429–444, 1978.
<https://www.utdallas.edu/~ryoung/phdseminar/CCR1978.pdf>. 51
- W. W. Cooper, L. M. Seiford, and K. Tone. *Data Envelopment Analysis. A Comprehensive Text with Models, Applications, References and DEA-Solver Software*. Springer, New York, 2nd edition, 2007.
<https://doi.org/10.1007/978-0-387-45283-8>. 51
- I. Das and J. E. Dennis. Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems. *SIAM J. Optim.*, 8(3):631–657, 1998.
<https://doi.org/10.1137/S1052623496307510>. 33
- M. Ehrgott. *Multicriteria Optimization*. Springer, Berlin, 2nd edition, 2005.
<https://doi.org/10.1007/3-540-27659-9>. 3, 19, 25, 39
- A. M. Geoffrion. Proper efficiency and the theory of vector maximization. *J. Math. Anal. Appl.*, 22(3):618–630, 1968.
[https://doi.org/10.1016/0022-247X\(68\)90201-1](https://doi.org/10.1016/0022-247X(68)90201-1). 9, 12
- A. M. Geoffrion, J. S. Dyer, and A. Feinberg. An interactive approach for multi-criterion optimization, with an application to the operation of an academic department. *Manage. Sci.*, 19(4-Part-1):357–368, 1972.
<https://doi.org/10.1287/mnsc.19.4.357>. 30
- M. Gernhardt et al. Deep space habitability design guidelines based on the nasa nextstep phase 2 ground test program. Report TP-2020-220505, NASA Johnson Space Center, Hampton, VA, USA, November 2019.
<https://ntrs.nasa.gov/api/citations/20200001427/downloads/20200001427.pdf>. 55

- M. Grafnetter. Dynamic resource balancing in virtualization clusters. Master's thesis, Faculty of Mathematics and Physics, Charles University in Prague, 2011.
<https://is.cuni.cz/webapps/zzp/detail/92291/?lang=en>. 7
- L. Grygarová. *Úvod do parametrického programování*. Karolinum, Praha, 1994. 18
- L. Grygarová. *Základy vícekriteriálního programování*. Karolinum, Praha, 1996. 3
- J. Guddat and K. Wendler. On dialogue-algorithms for linear and nonlinear vector optimization from the point of view of parametric programming. In M. Grauer and A. P. Wierzbicki, editors, *Interactive decision analysis*, volume 229 of *Lecture Notes in Economics and Mathematical Systems*, pages 123–131. Springer, 1984.
https://doi.org/10.1007/978-3-662-00184-4_14. 32
- P. Hansen. Bicriterion path problems. In G. Fandel and T. Gal, editors, *Multiple Criteria Decision Making Theory and Application*, volume 177 of *Lecture Notes in Economics and Mathematical Systems*, pages 109–127. Springer, Berlin, Heidelberg, 1980.
https://doi.org/10.1007/978-3-642-48782-8_9. 37
- M. Henig. Proper efficiency with respect to cones. *J. Optim. Theory Appl.*, 36:387–407, 1982.
<https://doi.org/10.1007/BF00934353>. 44
- M. Hladík. Complexity of necessary efficiency in interval linear programming and multiobjective linear programming. *Optim. Lett.*, 6(5):893–899, 2012.
<http://dx.doi.org/10.1007/s11590-011-0315-1>. 48
- M. Hladík. On relation of possibly efficiency and robust counterparts in interval multiobjective linear programming. In A. Sforza and C. Sterle, editors, *Optimization and Decision Science: Methodologies and Applications*, volume 217 of *Springer Proceedings in Mathematics & Statistics*, pages 335–343. Springer, Cham, 2017.
https://doi.org/10.1007/978-3-319-67308-0_34. 46
- C.-L. Hwang and A. S. M. Masud. *Multiple Objective Decision Making. Methods and Applications*, volume 164 of *LNE*. Springer, Berlin Heidelberg, 1979.
<https://doi.org/10.1007/978-3-642-45511-7>. 21
- M. Inuiguchi and M. Sakawa. Possible and necessary efficiency in possibilistic multiobjective linear programming problems and possible efficiency test. *Fuzzy Sets Syst.*, 78(2):231–241, 1996.
[https://doi.org/10.1016/0165-0114\(95\)00169-7](https://doi.org/10.1016/0165-0114(95)00169-7). 45
- J. Jahn. *Vector Optimization. Theory, Applications, and Extensions*. Springer, Berlin, 2nd edition, 2011.
<https://doi.org/10.1007/978-3-642-17005-8>. 3
- B. J. Kubica. *Interval Methods for Solving Nonlinear Constraint Satisfaction, Optimization and Similar Problems*, volume 766 of *SCI*. Springer, Cham, 2019.
<https://doi.org/10.1007/978-3-030-13795-3>. 35
- H. W. Kuhn and A. W. Tucker. Nonlinear programming. In *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability, 1950*, pages 481–492, Berkeley, 1951. University of California Press. 14
- R. T. Marler and J. S. Arora. Survey of multi-objective optimization methods for engineering. *Struct. Multidisc. Optim.*, 26:369–395, 2004.
<https://doi.org/10.1007/s00158-003-0368-6>. 3
- E. Q. V. Martins. On a multicriteria shortest path problem. *Eur. J. Oper. Res.*, 16(2):236–245, 1984.
[https://doi.org/10.1016/0377-2217\(84\)90077-8](https://doi.org/10.1016/0377-2217(84)90077-8). 38

- J. Mazurek. *Advances in Pairwise Comparisons*. Springer, Cham, 2023.
<https://doi.org/10.1007/978-3-031-23884-0>. 53
- K. Miettinen, K. Deb, J. Jahn, W. Ogryczak, K. Shimoyama, and R. Vetschera. Future challenges. In J. Branke et al., editor, *Multiobjective Optimization*, volume 5252 of *LNCS*, pages 435–461. Springer, Berlin, Heidelberg, 2008.
https://doi.org/10.1007/978-3-540-88908-3_16. 57
- K. M. Miettinen. *Nonlinear Multiobjective Optimization*, volume 12 of *ISOR*. Kluwer, Dordrecht, 1999.
<https://doi.org/10.1007/978-1-4615-5563-6>. 3
- F. Nožička, J. Guddat, H. Hollatz, and B. Bank. *Theorie der Linearen Parametrischen Optimierung*. Akademie-Verlag, Berlin, 1974. 18
- J. Rohn and J. Kreslová. Linear interval inequalities. *Linear Multilinear Algebra*, 38(1-2):79–82, 1994.
<https://doi.org/10.1080/03081089508818341>. 46
- T. L. Saaty. Relative measurement and its generalization in decision making. Why pairwise comparisons are central in mathematics for the measurement of intangible factors. The analytic hierarchy/network process. *RACSAM - Rev. R. Acad. Cien. Serie A. Mat.*, 102(2):251–318, 2008.
<https://doi.org/10.1007/BF03191825>. 53
- T. L. Saaty and L. G. Vargas. *Models, Methods, Concepts & Applications of the Analytic Hierarchy Process*, volume 175 of *ISOR*. Springer, New York, second edition, 2012.
<https://doi.org/10.1007/978-1-4614-3597-6>. 53
- T. L. Saaty and L. G. Vargas. *Decision Making with the Analytic Network Process*, volume 195 of *International Series in Operations Research & Management Science*. Springer, New York, second edition, 2013.
<https://doi.org/10.1007/978-1-4614-7279-7>. 53
- P. Serafini. Some considerations about computational complexity for multi objective combinatorial problems. In J. Jahn and W. Krabs, editors, *Recent Advances and Historical Development of Vector Optimization*, volume 294 of *LNE*, pages 222–232. Springer, Berlin, Heidelberg, 1987.
https://doi.org/10.1007/978-3-642-46618-2_15. 39
- R. E. Steuer. *Multiple Ccriteria Optimization. Theory, Computation, and Application*. Wiley, New York, 1986. 3, 21, 30
- D. The Luc. *Theory of Vector Optimization*, volume 319 of *LNE*. Springer, Berlin, 1989.
<https://doi.org/10.1007/978-3-642-50280-4>. 3
- D. The Luc. *Multiobjective Linear Programming. An Introduction*. Springer, Cham, 2016.
<https://doi.org/10.1007/978-3-319-21091-9>. 3
- W. L. Winston. *Operations Research. Applications and Algorithms*. Thomson Learning, Belmont, USA, fourth ed. edition, 2004. 30
- J. Zhu, editor. *Data Envelopment Analysis. A Handbook of Empirical Studies and Applications*. Springer, New York, 2016.
<https://doi.org/10.1007/978-1-4899-7684-0>. 51