ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Elektrotechnická fakulta

OPTIMÁLNÍ ROZHODOVÁNÍ A ŘÍZENÍ

Jan Štecha

Katedra řídicí techniky

1999

Předmluva

Toto skriptum je určeno posluchačům 4. ročníku oboru technická kybernetika. Je základní studijní pomůckou pro stejnojmenný předmět. Skriptum je zpracováno podrobněji než odpovídá odpřednášené látce, takže některé jeho části mohou být užitečné i posluchačům doktorandského studia.

Optimalizační problémy se vyskytují téměř v každém oboru lidské činnosti a proto znalost metod jejich řešení je velmi důležitá. V programovém souboru MATLAB existuje soubor optimalizačních programů (Optimization toolbox), který je vhodným nástrojem pro řešení konkrétních problémů. Proto nezbytnou součástí teorie, které je věnováno toto skriptum, je použití optimalizačních metod na řešení zadaných problémů. To je náplní cvičení z tohoto předmětu.

Problematika optimalizace je značně rozsáhlá a nové specielní numerické algoritmy optimalizace se stále vyvíjejí. Skriptum neobsahuje optimalizační algoritmy, které bychom mohli s trochou nadsázky nazývat "inteligentní", jako jsou například genetické algoritmy, simulované žíhání, metody Monte-Carlo, statistická teorie optimalizace, různé heuristické algorimy a podobně.

Zájemci o hlubší studium se musí obrátit ke specielní literatuře. Přehled vhodné literatury je uveden v závěru skripta.

Autor děkuje recenzentovi Ing. Antonínu Vaněčkovi, DrSc za řadu podnětných připomínek. Vřelý dík patří též kolegovi doc. Ing. Miroslavu Razímovi, CSc za pečlivé přečtení rukopisu a pomoc při odstranění řady stylistických i věcných nepřesností.

Skriptum je připraveno pomocí programu LaTeX.

Jan Štecha

Obsah

Kapitola 1

Úvod

Optimalizační problémy se vyskytují v každém oboru lidské činnosti. Každodenně řešíme řadu problémů, jak něco provést nejlepším způsobem.

Optimalizační problém vznikne v situaci, kdy je nutno vybrat (zvolit, rozhodnout) nějakou variantu řešení. Je jasné, že se snažíme vybrat tu variantu řešení, která je pro nás v nějakém smyslu nejvýhodnější. Hledáním nejlepší varianty řešení vznikne optimalizační problém, který řešíme různými optimalizačními metodami.

Abychom mohli optimalizační problém matematicky formulovat, je třeba vytvořit matematický model situace - vytvořit systém. Dále je třeba mít možnost porovnat různé varianty řešení a vybrat nejlepší variantu. Je jasné, že porovnávat různé varianty řešení můžeme pouze při simulaci na modelu situace, to v reálné rozhodovací situaci není možné. Optimální řešení jsou ta možná řešení, pro která neexistují řešení lepší.

Je zřejmé, že model reálné situace (reálného objektu) je vždy zjednodušen. Již v této fázi vytváření modelu se mohou objevit potíže, související s tím, že matematicky zpracovatelný model situace nepopisuje věrně realitu a naopak skutečnosti blízký model nebude matematicky zpracovatelný. Také vybírání nejlepší varianty řešení přináší řadu problémů. Pro matematickou formulaci optimalizačního problému volíme kritérium, podle kterého vybíráme nejlepší variantu řešení. Výběr kritéria optima je delikátní otázka v mnoha aplikacích a podléhá často subjektivním požadavkům.

Pro vyřešení reálného optimalizačního problému přes jeho matematický model a kritérium optima je často nutno podle výsledků upřesňovat model i modifikovat kritérium. Jedná se tedy častěji o opakované řešení různých variant optimalizačního problému a jejich ověřování na základě simulace a porovnání s realitou.

Vidíme, že při řešení optimalizačního problému se vyskytuje řada problémů. V teorii optimalizace (teorii optimálního řízení), která řeší optimalizační problém, jde o jistou filozofii, od které se očekává spíše metodika než vzorce a recepty, do kterých lze dosadit, abychom vypočetli řešení, které je po všech stránkách optimální.

My se přechodem od reálného optimalizačního problému k jeho matematickému modelu budeme zabývat málo, a to pouze při řešení konkrétních příkladů. Je dobře si na počátku uvědomit, že tento krok je velmi důležitý a podstatně ovlivňuje využitelnost výsledků. Při vytváření modelu reálného objektu či situace jsou nezbytné důkladné znalosti problematiky, do které optimalizační problém patří. Zde se v plné míře uplatní zkušenost

řešitele získaná při řešení podobných problémů a při přenášení výsledků do reality.

Situaci, při níž je potřeba se rozhodnout, budeme nazývat rozhodovací situace. Existuje mnoho situací, jejichž matematický model je statický - popisuje pouze algebraické vztahy mezi veličinami, které jsou časově neproměnné. Také při modelování složitých dynamických systémů nám často nic jiného nezbude, než zanedbat jejich dynamiku a vytvářet pouze statické modely. Proto se nejprve budeme zabývat řešením statických optimalizačních problémů, u nichž se nevyskytuje čas jako nezávisle proměnná. Řešením těchto problémů se zabývá samostatný vědní obor, který se nazývá matematické programování. Jsou-li vztahy mezi veličinami lineární a kritérium je také lineární, dostaneme problém lineárního programování. Obecná úloha tohoto typu vede na problém nelineárního programování.

Je-li modelem situace dynamický systém, pak optimalizační problém je problémem dynamické optimalizace, který také často nazýváme problémem optimálního řízení. Protože pro statickou úlohu nelineárního programování byla vypracována řada účinných numerických metod, často se snažíme převést úlohu dynamické optimalizace na statický problém. Existují však metody, které řeší problém dynamické optimalizace přímo. Jedná se o variační metody, princip maxima a dynamické programování. Variační metody a princip maxima formulují nutné podmínky, které musí splňovat optimální řešení. Jedná se o řešení soustavy diferenciálních rovnic. Pomocí dynamického programování dostáváme rekurentní vztahy, které jsou vhodné pro numerický výpočet.

Je již nyní zřejmé, že tato publikace tvoří pouze úvod do problematiky optimalizace. Podrobnosti je třeba hledat ve speciální literatuře.

1.1 Optimalizační problémy

S optimalizačními problémy se setkáme téměř všude. Uvedeme nyní několik příkladů, které zahrnují velkou třídu problémů.

1. Alokační problémy
Jedná se o optimální rozdělění zdrojů a určení optimálního výrobního programu. Výrobce má k dispozici výrobní zařízení, která jsou schopna vyrábět n různých druhů výrobků (zboží) z m surovin, jejichž zdroj je omezen. Problémem je rozdělit suroviny na možné výrobky tak, abychom maximalizovali zisk.

Zisk z jednoho výrobku j-tého typu je c_j a výrobce ho vyrábí x_j kusů. Na výrobu j-tého výrobku potřebuje výrobce a_{ij} jednotek suroviny i-tého typu. Při tom má výrobce k dispozici b_i jednotek suroviny i-tého typu. Maximalizace zisku při uvažování omezení zdrojů surovin vede zřejmě na následující úlohu

$$\max\{\sum_{j=1}^{n} c_j x_j \ : \ \sum_{j=1}^{m} a_{ij} x_j \leq b_i, x_j \geq 0\}$$

neboli pomocí odpovídajících vektorů a matic

$$\max\{\mathbf{c}^T\mathbf{x} \ : \ \mathbf{A}\mathbf{x} \leq \mathbf{b}, \mathbf{x} \geq 0\}.$$

2. Problémy plánování
Jedná se o problémy nejvhodnějšího plánování investic do výrobních zařízení, zajištění financování komplexních projektů, případně plánování výroby.

Jako jeden problém z této oblasti si uvedeme problém plánování výroby na určitém časovém horizontu ($t \in [0, T]$). Je známa poptávková funkce $d(t)$, dále jako $r(t)$ označíme velikost produkce za jednotku času, která je z provozních důvodů omezena. Výrobní náklady jsou úměrné velikosti produkce a skladovací náklady rostou úměrně s množstvím neprodaného zboží, které označíme $x(t)$. Je zřejmé, že $x(t)$ je popsáno diferenciální rovnicí

$$\dot{x}(t) = r(t) - d(t), \qquad r(t) \geq 0, \qquad x(t) \geq 0,$$

Kritériem jsou celkové výrobní náklady

$$J = \int_0^T (cr(t) + hx(t))dt,$$

které chceme minimalizovat volbou optimální produkce $r(t)$.

3. Problémy optimálního řízení dynamických systémů
Při řízení naše požadavky vyjádříme volbou kritéria kvality řízení, které zohledňuje odchylky od požadované trajektorie a vynaloženou řídicí energii. Minimalizace zvoleného kritéria při respektování dynamických vlastností systému vede na optimalizační problém. Těmito problémy se budeme zabývat podrobně později.

4. Problémy aproximace
Často chceme aproximovat funkci $f(t)$ na určitém intervalu jinou funkcí $p(t)$ (nejčastěji polynomem) tak, aby chyba aproximace $e(t) = f(t) - p(t)$ byla minimální vzhledem k určitému kritériu na př.

$$\int_a^b e^2(t)dt, \quad \text{nebo} \quad \max_{a \leq t \leq b} |e(t)|, \quad \text{případně} \quad \int_a^b |e(t)|dt.$$

5. Problémy estimace (odhadování)
Problémem je odhad určité veličiny z nepřesných pozorování. Jedná se o specielní třídu aproximačních problémů. Při tom vyžadujeme přesnou formulaci kritéria i vlastností chyb měření.

6. Konfliktní situace - hry
Existuje mnoho situací s protikladnými zájmy účastníků. Modelem takové situace je hra. Typickými problémy v této oblasti z oboru řízení je problém pronásledovaného a pronásledujícího v leteckých soubojích. V alokačních problémech se jedná např. o reklamní kampaň, volební kampaň a podobně.

Při řešení optimalizačního problému v podmínkách neurčitosti je třeba brát v úvahu určité zvláštnosti. Je třeba brát v úvahu **riziko** při formulaci problému a využívat **získávání informace během rozhodovacího procesu**. Ukážeme to názorně v následujících třech příkladech.

Příklad 1.: Chceme rozdělit kapitál na dvě investiční možnosti, které označíme A, B. Investice A nabízí zisk 1.2Kč za jednu investovanou korunu, zatímco z investice B získáme pouze 1.1Kč z jedné investované koruny. V tomto případě (bez neurčitosti) je naše rozhodnutí jasné, investujeme do A s větším ziskem. Uvažujme nyní, že zisk 10% z investice B je jistý, ale zisk 20% z investice A je pouze v průměru. Tak např. získáme 0 Kč s pravděpodobností 4/5 a velký zisk 6 Kč s pravděpodobností pouze 1/5. Střední hodnota zisku z investice A je opět 20%. V reálné situaci bychom pravděpodobně ztrátu veškerého vloženého kapitálu do nejistého zisku z investice A neriskovali a část kapitálu vložili do jistého, ale menšího zisku z investice do B. Lepší je vrabec v hrsti, nežli holub na střeše. Proto je důležité brát v úvahu riziko a správně formulovat optimalizační problém.

Příklad 2.: St. Peterburgský paradox Ještě názornější příklad nutnosti uvažovat riziko plyne z následující hazardní hry. Hráč zaplatí x jednotek, aby se mohl účastnit následující hazardní hry. Jsou prováděny následné vrhy mincí a hráč vyhraje 2^k jednotek, padne-li mu za sebou k-kráte panna před tím, než mu poprvé padne lev.

Uvažujte nejprve bez jakýchkoli výpočtů, kolik byste byli ochotni zaplatit, abyste mohli sehrát tuto hazardní hru, ve které můžete vyhrát značnou částku. Určitě nebudete ochotni vložit do této hazardní hry více než desítky korun (abychom uvažovali konkrétní jednotky). Je to způsobeno tím, že výhra je nejistá, zatímco vložená částka za účast ve hře je ztracena jistě.

Spočtěme nyní, jaká je střední hodnota hry. Padne-li poprvé lev, hra končí, hráč vyhraje $2^0 = 1$ Kč a to se stane s pravděpodobností 1/2. Padne-li poprvé panna a pak lev vyhrajete 2^1 Kč a to se stane s pravděpodobností 1/4. Padne-li dvakrát za sebou panna a pak lev, vyhrajete $2^2 = 4$ Kč a to se stane s pravděpodobností 1/8. Podobně vyhrajeme $x_i = 2^i$ Kč padne-li i-kráte za sebou panna a pak lev, což se stane s pravděpodobností $p_i = 1/(2^{i+1})$. Střední hodnota výhry je tedy

$$m = \sum_{i=0}^{\infty} x_i p_i = \sum_{i=0}^{\infty} \frac{2^i}{2^{i+1}} = \frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \ldots = \infty$$

To znamená, že střední hodnota výhry je ∞ a proto vložená částka za účast ve hře může být libovolně veliká a přesto nemůžeme (ve střední hodnotě) prohrát. Vysoké výhry mají ale malou pravděpodobnost, riziko ztráty je zde veliké a to bereme v reálné situaci v úvahu.

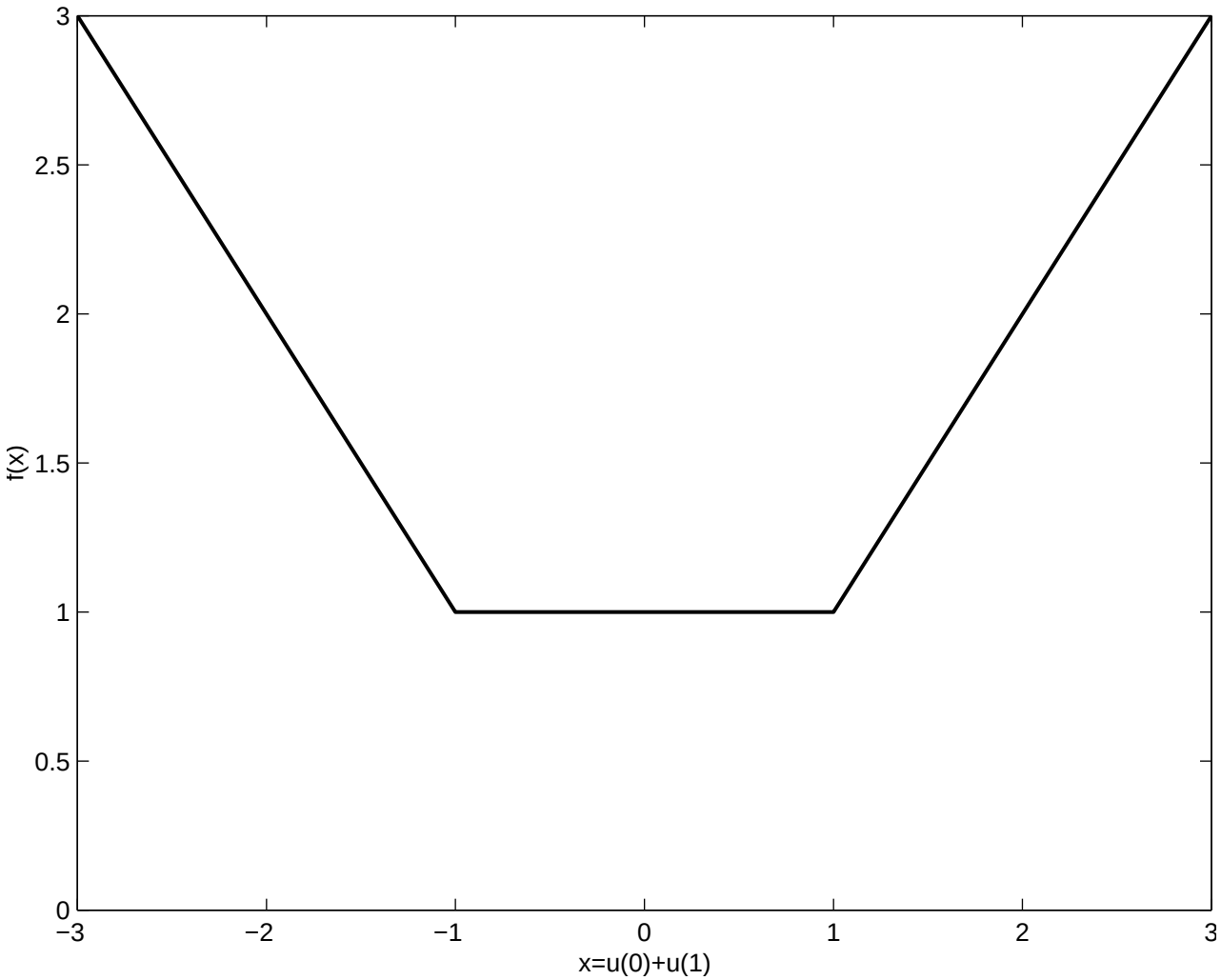
Příklad 3.: Nutnost získávání informací při sekvenčním rozhodování ukážeme na následujícím jednoduchém příkladě. Máme systém popsaný stavovou rovnicí

$$\begin{aligned} x_1(t+1) &= u(t) + v(t) \\ x_2(t+1) &= x_1(t) + u(t) \end{aligned}$$

kde $v(t)$ je náhodná posloupnost nabývající hodnot ± 1, obě s pravděpodobností 0.5. Naším úkolem je nalézt řízení $u(0)$ a $u(1)$ takové, aby střední hodnota $\mathcal{E}\{|x_2(2)|\}$ byla minimální.

Spočtěme tedy nejprve střední hodnotu jednorázovým výpočtem

$$\begin{aligned} \mathcal{E}\{|x_2(2)|\} &= \mathcal{E}\{|x_1(1) + u(1)|\} = \mathcal{E}\{|u(0) + v(0) + u(1)|\} \\ &= 0.5|u(0) + u(1) + 1| + 0.5|u(0) + u(1) - 1| \end{aligned}$$



Obrázek 1.1: Průběh $\mathcal{E}\{|x_2(2)|\} = f(u(0) + u(1))$

Zřejmě platí

$$\min_{u(0),u(1)} \mathcal{E}\{|x_2(2)|\} = 1 \quad \text{pro} \quad u(0) + u(1) \in [-1; +1].$$

Na obr. 1.1 je nakreslen průběh funkce $0.5|\alpha + 1| + 0.5|\alpha - 1|$ kde $\alpha = u(0) + u(1)$.

Pokud ale budeme řídit zpětnovazebně, to znamená získávat informaci během procesu, pak při znalosti $x_1(1)$ bude

$$\min_{u(0),u(1)} \mathcal{E}\{|x_2(2)|\} = \min\{|x_1(1) + u(1)| = 0, \quad \text{pro} \quad u(1) = -x(1)$$

Průběžné získávání informací během sekvenčního rozhodovacího procesu není nutné, není-li žádná nejistota.

Kapitola 2

Nelineární programování - analytické metody

V této kapitole se budeme zabývat řešením optimalizačních úloh, u nichž modelem situace je **statický systém**. Jedná se tedy v podstatě o metody hledání extrémů funkcí více proměnných, které podléhají různým omezením. Tyto problémy můžeme také označovat jako **statické optimalizace**, nebo také úlohy **matematického programování**.

V této kapitole uvedeme nutné i postačující podmínky, které musí splňovat řešení optimalizační úlohy. Výsledkem budou soustavy rovnic či nerovnic, jejichž řešením můžeme vyřešit naši optimalizační úlohu. Proto tento postup, který vede k cíli pouze v jednoduchých případech, nazýváme analytickými metodami řešení úloh matematického programování. Numerickými metodami se budeme zabývat později. Přestože se většina reálných úloh matematického programování řeší numerickými metodami na počítačích, je rozumné znát nutné a postačující podmínky řešení těchto úloh.

V závěru této kapitoly budeme diskutovat problémy vícekriteriální optimalizace. Tyto problémy se blíží reálným problémům, ve kterých se vyskytuje více hodnotících kritérií.

2.1 Klasifikace úloh matematického programování

Základní úloha matematického programování je nalézt extrém (maximum či minimum) skalární funkce $f(\mathbf{x})$, kde hledaný vektor $\mathbf{x}$ je prvkem nějaké množiny $\mathbf{X}$. Množina $\mathbf{X}$ je nejčastěji určena nějakými algebraickými vztahy mezi složkami vektoru $\mathbf{x}$.

Základní úloha matematického programování je tedy

$$\min \{f(\mathbf{x}) \ : \mathbf{x} \in \mathbf{X} \subset E^n\} \tag{2.1}$$

kde E^n je n-rozměrný Eukleidův prostor. Je lhostejné, zda se jedná o maximum či minimum, neboť platí

$$\min \{f(\mathbf{x}) \ : \mathbf{x} \in \mathbf{X}\} = -\max \{-f(\mathbf{x}) \ : \mathbf{x} \in \mathbf{X}\} \tag{2.2}$$

Existenci extrému zajišťuje Weierstrassova věta:

Věta: *Každá spojitá funkce $f(\mathbf{x})$ definovaná na kompaktní (ohraničené a uzavřené) množině $\mathbf{X} \subset E^n$, má na ní maximální i minimální hodnotu.* □

Za uvedených předpokladů existují tedy body $\mathbf{x}_{max}$ a $\mathbf{x}_{min}$, pro které platí

$$\begin{aligned} f(\mathbf{x}_{max}) &= \max_{\mathbf{x}\in X} f(\mathbf{x}) = \sup_{\mathbf{x}\in X} f(\mathbf{x}), \qquad \mathbf{x}_{max} = \arg\max_{\mathbf{x}\in X} f(\mathbf{x}) \\ f(\mathbf{x}_{min}) &= \min_{\mathbf{x}\in X} f(\mathbf{x}) = \inf_{\mathbf{x}\in X} f(\mathbf{x}), \qquad \mathbf{x}_{min} = \arg\min_{\mathbf{x}\in X} f(\mathbf{x}) \end{aligned}$$

Bod extrému funkce (v dalším budeme vždy hledat minimum) budeme značit $\mathbf{x}^*$. Extrém funkce může být buď lokální nebo globální. Uvedeme si příslušné definice

Definice: *Relativní (lokální) minimum.*
Bod $\mathbf{x}^* \in \mathbf{X}$ je bodem relativního *(*lokálního*) minima funkce $f(\mathbf{x})$ na množině $\mathbf{X}$, jestliže existuje $\varepsilon > 0$, že platí $f(\mathbf{x}) \geq f(\mathbf{x}^*)$ pro všechna $\mathbf{x} \in \mathbf{X}$, pro která platí $|\mathbf{x} - \mathbf{x}^*| < \varepsilon$ (bod $\mathbf{x}$ leží v ε-okolí bodu $\mathbf{x}^*$).*
Pokud v ε-okolí platí ostrá nerovnost $f(\mathbf{x}) > f(\mathbf{x}^*)$ pro $\mathbf{x} \neq \mathbf{x}^*$, pak $\mathbf{x}^*$ je bodem ostrého relativního minima funkce f na množině $\mathbf{X}$.

Definice: *Globální minimum.*
Bod $\mathbf{x}^* \in \mathbf{X}$ je bodem globálního *minima funkce $f(\mathbf{x})$ na množině X, jestliže platí $f(\mathbf{x}) \geq f(\mathbf{x}^*)$ pro všechna $\mathbf{x} \in \mathbf{X}$.*
Pokud pro všechna $\mathbf{x} \in \mathbf{X}$ a $\mathbf{x} \neq \mathbf{x}^*$ platí ostrá nerovnost $f(\mathbf{x}) > f(\mathbf{x}^*)$, pak $\mathbf{x}^*$ je bodem ostrého globálního minima funkce f na množině $\mathbf{X}$.

Většina algoritmů nám umožní nalézt pouze lokální extrémy. Globální extrémy nalezneme pouze za určitých předpokladů o konvexnosti problému.

Pokud funkce $f(\mathbf{x})$ je lineární v proměnné $\mathbf{x}$ a množina $\mathbf{X}$ je určena soustavou lineárních nerovnic, pak se jedná o problém *lineárního programování*, pro který jsou vypracovány spolehlivé algoritmy k nalezení globálního maxima či minima.

Pokud omezující množina je $\mathbf{X} = E^n$, nemáme omezení. Jedná se potom o problém určení *volného extrému*. Omezení jsou často určena soustavou rovnic

$$\mathbf{X} = \{\mathbf{x} : \mathbf{x} \in E^n, h_i(\mathbf{x}) = 0, i = 1, 2, \ldots, m\} \tag{2.3}$$

Potom se jedná o problém na vázaný extrém v užším smyslu.

Obecná úloha nelineárního programování (používá se i pojem matematické programování) je úloha, ve které je minimalizovaná funkce $f(\mathbf{x})$ nelineární a omezení jsou určena soustavou rovnic i nerovnic.

$$\mathbf{X} = \{\mathbf{x} : \mathbf{x} \in E^n;\ h_i(\mathbf{x}) = 0, i = 1, 2, \ldots, m;\ g_j(\mathbf{x}) \leq 0, j = 1, 2, \ldots, p\} \tag{2.4}$$

Rozdíl mezi problémy s omezením ve tvaru rovnosti či nerovnosti je pouze formální, neboť každou rovnost $h(\mathbf{x}) = 0$ můžeme nahradit dvěma nerovnicemi: $h(\mathbf{x}) \leq 0$ a $-h(\mathbf{x}) \leq 0$. Obráceně lze každou nerovnici $g_j(\mathbf{x}) \leq 0$ nahradit rovnicí $g_j(\mathbf{x}) + y^2 = 0$, kde y je pomocná proměnná, která se nevyskytuje v kritériu f a nejsou na ni kladena žádná omezení.

Přitom ale množina $\mathbf{X}$, zadaná soustavou rovnic, nemá vnitřní body, to znamená, že v každém okolí bodu $\mathbf{x} \in \mathbf{X}$ jsou body, které do množiny $\mathbf{X}$ nepatří. Naopak množina $\mathbf{X}$ zadaná soustavou nerovnic, může mít vnitřní body. V tom se tyto problémy zásadně liší. Omezení typu rovnosti dostaneme pouze velkým zjednodušením modelu reálné situace. Odchýlíme-li se nepatrně od libovolného přípustného řešení, je v praxi nereálné dostat řešení, které není přípustné.

Klasifikace problémů podle účelové (kriteriální) funkce není důležité, neboť každou účelovou funkci lze převést na lineární účelovou funkci. Platí totiž ekvivalence následujících úloh

$$\begin{aligned} \min\{f(\mathbf{x}) &: \mathbf{x} \in E^n, h_i(\mathbf{x}) = 0, g_j(\mathbf{x}) \leq 0\} \\ \min\{y &: f(\mathbf{x}) - y \leq 0 : \mathbf{x} \in E^n, y \in E^1, h_i(\mathbf{x}) = 0, g_j(\mathbf{x}) \leq 0\} \end{aligned}$$

Někdy požadujeme, aby některé nebo všechny proměnné nabývaly pouze celočíselných hodnot. Problémy tohoto typu nazýváme **celočíselným programováním**. Specielním případem celočíselného programování je **binární programování**, kde proměnné nabývají pouze dvou hodnot, často 0 nebo 1. Zde se těmito metodami nebudeme zabývat.

2.2 Přípustné směry - volný extrém

Mějme tedy optimalizační úlohu $\min\{f(\mathbf{x}) : \mathbf{x} \in \mathbf{X} \subset E^n\}$. Nejprve si budeme definovat přípustný směr.

Definice: *Mějme bod* $\mathbf{x} \in \mathbf{X}$*, pak vektor* $\mathbf{s}$ *je přípustný směr v bodě* $\mathbf{x}$*, jestliže existuje* $\beta > 0$*, že*

$$\mathbf{x} + \alpha\mathbf{s} \in \mathbf{X} \qquad \text{pro všechny} \quad \alpha; \quad 0 \leq \alpha \leq \beta \tag{2.5}$$

2.2.1 Podmínky prvního řádu

Je zřejmé, že pokud je $\mathbf{x}^*$ bodem relativního minima, pak nemůže existovat takový přípustný směr, že v bodě $\mathbf{x} = \mathbf{x}^* + \alpha\mathbf{s}$ nabývá funkce menších hodnot. Musí tedy platit $f(\mathbf{x}) \geq f(\mathbf{x}^*)$. Při aproximaci prvního řádu funkce $f(\mathbf{x})$ v bodě $\mathbf{x}^*$ platí $f(\mathbf{x}) \doteq f(\mathbf{x}^*) + \nabla f(\mathbf{x}^*)(\mathbf{x} - \mathbf{x}^*) = f(\mathbf{x}^*) + \alpha\nabla f(\mathbf{x}^*)\mathbf{s}$. Odtud plyne nutná podmínka minima.

Věta: *Je-li* $\mathbf{x}^*$ *bodem relativního minima funkce* $f(\mathbf{x}) \in C^1$ *($f \in C^1$ znamená, že funkce f má spojité první parciální derivace) na množině* $\mathbf{X}$*, pak pro libovolný vektor* $\mathbf{s}$*, který je přípustným směrem v bodě* $\mathbf{x}^*$ *platí*

$$\nabla f(\mathbf{x}^*)\mathbf{s} \geq 0. \tag{2.6}$$

Poznámka: Derivace skalární funkce $f(\mathbf{x})$ podle vektorového argumentu $\mathbf{x}$ je řádkový vektor, který značíme $\nabla f(\mathbf{x})$. Platí tedy

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \left[\begin{array}{cccc} \frac{\partial f(\mathbf{x})}{\partial x_1} & \frac{\partial f(\mathbf{x})}{\partial x_2} & \cdots & \frac{\partial f(\mathbf{x})}{\partial x_n} \end{array} \right] = \nabla f(\mathbf{x}). \tag{2.7}$$

Gradient skalární funkce $f(\mathbf{x})$ v bodě $\mathbf{x}$ je sloupcový vektor, platí grad $f(\mathbf{x}) = \nabla^T f(\mathbf{x})$. □

Ve vnitřním bodě je přípustný směr libovolný a proto nutné podmínky prvního řádu pro vnitřní bod $\mathbf{x}^*$ množiny $\mathbf{X}$ jsou následující:

Věta: *Pokud* $\mathbf{x}^*$ *je bod relativního minima funkce* $f(\mathbf{x})$ *na množině* $\mathbf{X}$ *a* $\mathbf{x}^*$ *je vnitřní bod množiny* $\mathbf{X}$*, pak*

$$\nabla f(\mathbf{x}^*) = \mathbf{0}. \tag{2.8}$$

Pokud máme úlohu na volný extrém, pak každý bod je vnitřním bodem a předchozí věta platí také. Předchozí tvrzení tvoří nutné podmínky prvního řádu. Tyto podmínky jsme dostali při aproximaci prvního řádu funkce f v bodě $\mathbf{x}^*$.

2.2.2 Podmínky druhého řadu

Podmínky druhého řádu dostaneme při aproximaci druhého řádu funkce f v bodě $\mathbf{x}^*$. Platí

$$\begin{aligned} f(\mathbf{x}) &\doteq f(\mathbf{x}^*) + \nabla f(\mathbf{x}^*)(\mathbf{x}-\mathbf{x}^*) + \frac{1}{2}(\mathbf{x}-\mathbf{x}^*)^T \nabla^2 f(\mathbf{x}^*)(\mathbf{x}-\mathbf{x}^*) \\ &\doteq f(\mathbf{x}^*) + \alpha \nabla f(\mathbf{x}^*)\mathbf{s} + + \frac{1}{2}\alpha^2 \mathbf{s}^T \nabla^2 f(\mathbf{x}^*)\mathbf{s} \end{aligned}$$

Poznámka: Druhá derivace skalární funkce $f(\mathbf{x})$ podle vektorového argumentu $\mathbf{x}$ je Hessova matice, kterou značíme $\mathbf{H}(\mathbf{x}) = \nabla^2 f(\mathbf{x})$.

$$\mathbf{H}(\mathbf{x}) = \nabla^2 f(\mathbf{x}) = \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x}^2} = \begin{bmatrix} \frac{\partial^2 f(\mathbf{x})}{\partial x_1^2} & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_2} & \cdots & \frac{\partial f(\mathbf{x})}{\partial x_1 \partial x_n} \\ \cdots & & & \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_1} & \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_2} & \cdots & \frac{\partial f(\mathbf{x})}{\partial x_n \partial x_n} \end{bmatrix}$$

□

Z předchozí aproximace druhého řádu plynou nutné podmínky druhého řádu.

Věta: *Je-li* $\mathbf{x}^*$ *bodem relativního minima funkce* $f(\mathbf{x}) \in C^2$ *(*$f \in C^2$ *znamená, že funkce* f *má spojité i druhé parciální derivace) na množině* $\mathbf{X}$*, pak pro libovolný vektor* $\mathbf{s}$*, který je přípustným směrem v bodě* $\mathbf{x}^*$ *platí*

$$\begin{aligned} &1) \quad \nabla f(\mathbf{x}^*)\mathbf{s} \geq 0 \\ &2) \quad \text{Je-li} \quad \nabla f(\mathbf{x}^*)\mathbf{s} = 0, \quad \text{pak} \quad \mathbf{s}^T \nabla^2 f(\mathbf{x}^*)\mathbf{s} \geq 0. \end{aligned}$$

Je-li $\mathbf{x}^*$ vnitřní bod množiny $\mathbf{X}$, pak podmínka 1) platí pro libovolné $\mathbf{s}$, čili $\nabla f(\mathbf{x}^*) = \mathbf{0}$ a podmínka 2) je v tomto případě $\nabla^2 f(\mathbf{x}^*) \geq \mathbf{0}$, to znamená, že Hessova matice je pozitivně semidefinitní.

Pro nalezení globálních extrémů je nutné zavést předpoklady o konvexnosti kriteriální funkce a konvexnosti omezující množiny.

Množina $\mathbf{X}$ je konvexní, když pro libovolné dva body konvexní množiny platí, že celá úsečka mezi těmito body patří do množiny $\mathbf{X}$.

Definice: *Funkce f definovaná na konvexní množině $\mathbf{X}$ je konvexní, jestliže pro libovolné $\mathbf{x}_1, \mathbf{x}_2 \in \mathbf{X}$ a každé α, $0 \leq \alpha \leq 1$ platí*

$$f(\alpha\mathbf{x}_1 + (1-\alpha)\mathbf{x}_2) \leq \alpha f(\mathbf{x}_1) + (1-\alpha) f(\mathbf{x}_2)$$

Platí-li ostrá nerovnost (pro $\mathbf{x}_1 \neq \mathbf{x}_2$), pak funkce je striktně konvexní. Také součet i pozitivní lineární kombinace konvexních funkcí je funkce konvexní.

Je-li $h(\mathbf{x})$ konvexní funkce, pak množina $\mathbf{X}$ určená nerovností

$$\mathbf{X} = \{\mathbf{x} : h(\mathbf{x}) \leq b\}$$

je konvexní pro libovolné reálné b. Předchozí tvrzení platí i pro množinu $\mathbf{X}$ tvořenou soustavou nerovností $(h_1(\mathbf{x}) \leq b_1, \ldots, h_m(\mathbf{x}) \leq b_m)$, kde $h_i(\mathbf{x})$ jsou konvexní funkce. Pro konvexní funkce diferencovatelné platí následující dvě tvrzení:

Věta: *Je-li $f \in C^1$, pak funkce f je konvexní na konvexní množině $\mathbf{X}$ právě tehdy, když*

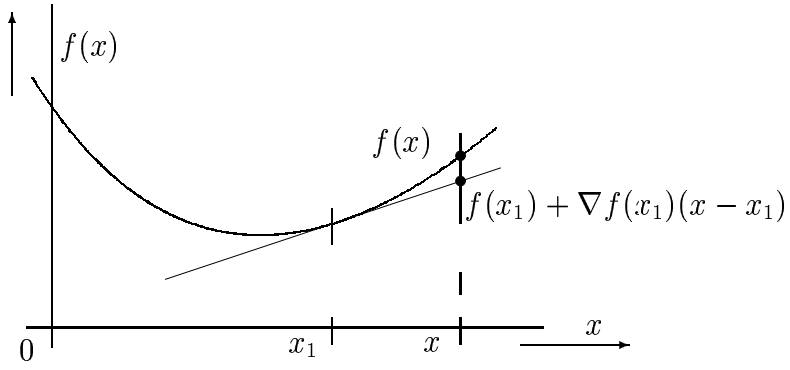
$$f(\mathbf{x}) \geq f(\mathbf{x}_1) + \nabla f(\mathbf{x})\ (\mathbf{x} - \mathbf{x}_1)$$

pro všechna $\mathbf{x}, \mathbf{x}_1 \in \mathbf{X}$.

Věta: *Je-li $f \in C^2$, pak funkce f je konvexní na konvexní množině $\mathbf{X}$ obsahující vnitřní bod právě tehdy, když Hessova matice $\mathbf{H}(\mathbf{x})$ je pozitivně semidefinitní pro $\mathbf{x} \in \mathbf{X}$* □

$$H(\mathbf{x}) = \nabla^2 f(\mathbf{x}) \geq \mathbf{0} \qquad \mathbf{x} \in X.$$

První tvrzení plyne z toho, že konvexní funkce leží nad tečnou ve svém libovolném bodě - viz obr. 2.1. Druhé tvrzení je mnoharozměrové zobecnění známého faktu, že konvexní funkce jedné proměnné má druhou derivaci kladnou. Předchozí nutné podmínky lokálních



Obrázek 2.1: Konvexní funkce leží nad tečnou v libovolném bodě

extrémů se pro konvexní funkce mění na globální podmínky nutné a postačující.

Věta: *Je-li f konvexní funkce definovaná na konvexní množině $\mathbf{X}$, potom množina $\mathbf{\Gamma}$, na které funkce f dosahuje minima, je také konvexní a libovolné relativní minimum je globální minimum.*

Věta: *Nechť $f \in C^1$ na konvexní množině $\mathbf{X}$. Je-li $\mathbf{x}^* \in \mathbf{X}$ a pro všechna $\mathbf{x} \in \mathbf{X}$ platí $\nabla f(\mathbf{x}^*)(\mathbf{x} - \mathbf{x}^*) \geq 0$, pak $\mathbf{x}^*$ je bod globálního minima funkce f na $\mathbf{X}$.*

Pro maximalizaci na konvexních funkcích platí pouze následující tvrzení:

Věta: *Nechť f je konvexní funkce definovaná na ohraničené, uzavřené konvexní množině* $\mathbf{X}$. *Má-li f maximum na* $\mathbf{X}$, *pak toto maximum je dosaženo v krajním bodě množiny* $\mathbf{X}$.

2.3 Vázané extrémy

Nyní budeme uvažovat minimalizační problém s funkcionálními omezeními ve tvaru rovnic a nerovnic

$$\min \{f(\mathbf{x}) \ : \ h_1(\mathbf{x}) = 0, \ldots, h_m(\mathbf{x}) = 0 \ ; \ g_1(\mathbf{x}) \leq 0, \ldots, g_p(\mathbf{x}) \leq 0\} \tag{2.9}$$

Zavedením vektorových funkcí $\mathbf{h}(\mathbf{x}) = \left[\ h_1, \ \ \ldots, \ \ h_m \ \right]^T$, a
$\mathbf{g}(\mathbf{x}) = \left[\ g_1, \ \ \ldots, \ \ g_p \ \right]^T$ můžeme předchozí úlohu zapsat v kompaktním tvaru

$$\min \{f(\mathbf{x}) \ : \ \mathbf{h}(\mathbf{x}) = 0 \ ; \ \mathbf{g}(\mathbf{x}) \leq 0\} \tag{2.10}$$

Nyní si zavedeme pojem aktivního omezení a regulárního bodu. Aktivní omezení je to omezení, které se aktivně uplatní, to znamená, že omezení ve tvaru rovnosti je aktivní v každém bodě a omezení $g_i(\mathbf{x}) \leq 0$ je aktivní v bodě $\mathbf{x}$, platí-li $g_i(\mathbf{x}) = 0$. Neaktivní omezení (v bodě $\mathbf{x}$ platí $g_i(\mathbf{x}) < 0$) se neuplatní a můžeme ho tedy ignorovat. Aktivní omezení $\mathbf{h}(\mathbf{x}) = 0$, kterých nechť je m, definují v n-rozměrném prostoru varietu. Jsou-li omezení regulární, pak varieta má dimenzi $n - m$.

V nějakém bodě $\mathbf{x}^*$ budeme konstruovat tečnou nadrovinu k varietě $\mathbf{h}(\mathbf{x}) = 0$. Definujeme si podprostor $\mathbf{M} = \{\mathbf{y} : \nabla\mathbf{h}(\mathbf{x}^*)\mathbf{y} = 0\}$. Tento podprostor je tečnou nadrovinou pouze tehdy, jsou-li gradienty $\nabla h_i(\mathbf{x})$ lineárně nezávislé. Zavedeme si tedy definici regulárního bodu:

Definice: *Bod $\mathbf{x}^*$ je regulární bod množiny $\mathbf{X} = \{\mathbf{x} : \mathbf{h}(\mathbf{x}) = 0 \ \}$, jsou-li gradienty $\nabla h_i(\mathbf{x})$ v bodě $\mathbf{x}^*$ lineárně nezávislé, neboli hodnost Jacobiho matice*

$$\nabla\mathbf{h}(\mathbf{x}^*) = \frac{\partial\mathbf{h}(\mathbf{x}^*)}{\partial\mathbf{x}} = \begin{bmatrix} \dfrac{\partial h_1(\mathbf{x})}{\partial x_1} & \cdots & \dfrac{\partial h_1(\mathbf{x})}{\partial x_n} \\ \cdots & & \\ \dfrac{\partial h_m(\mathbf{x})}{\partial x_1} & \cdots & \dfrac{\partial h_m(\mathbf{x})}{\partial x_n} \end{bmatrix}$$

je rovna počtu omezení m.

Poznámka: Je důležité si uvědomit, že regularita bodu není vlastností množiny X, ale její reprezentace pomocí funkcí $\mathbf{h}(\mathbf{x})$. Tak například množinu $\mathbf{X} = \{\mathbf{x} \in E^2, x_1 = 1\}$ můžeme popsat jako množinu všech řešení $h(\mathbf{x}) = x_1 - 1 = 0$ nebo jako množinu všech řešení $h(\mathbf{x}) = (x_1 - 1)^2 = 0$. V prvním případě jsou všechny body množiny regulární, kdežto v druhém případě není žádný bod množiny $\mathbf{X}$ regulární. Podobně pro dvě lineárně závislé omezující funkce h_i. □

2.3.1 Omezení typu rovnosti

Mějme tedy následující problém

$$\min\{f(\mathbf{x}) \ : \ \mathbf{h}(\mathbf{x}) = \mathbf{0}\} \tag{2.11}$$

Nyní uvedeme nutné podmínky prvního řádu pro omezení typu rovnosti. Platí následující lemma:

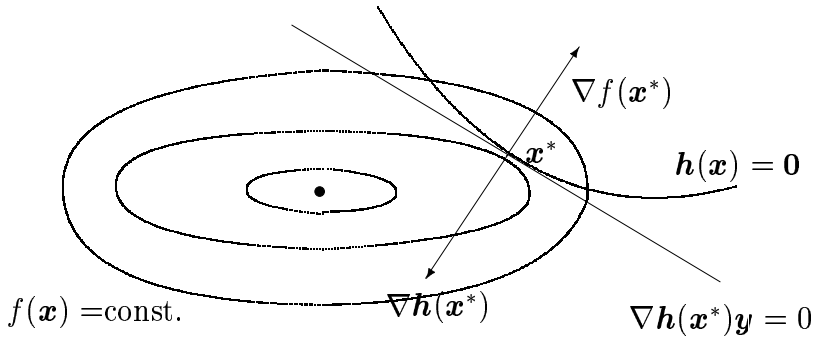
Lemma: *Nechť* $\mathbf{x}^*$ *je regulární bod omezení* $\mathbf{h}(\mathbf{x}) = 0$ *a je to lokální extrém funkce* $\mathbf{f}(\mathbf{x})$ *vzhledem k omezením. Pak pro všechna* $\mathbf{y} \in E^n$ *splňující*

$$\nabla\mathbf{h}(\mathbf{x}^*)\mathbf{y} = 0 \tag{2.12}$$

musí také platit

$$\nabla f(\mathbf{x}^*)\mathbf{y} = 0 \tag{2.13}$$

□



Obrázek 2.2: Gradienty funkce a omezení v extrému

To znamená, že $\nabla f(\mathbf{x}^*)$ je ortogonální k tečné nadrovině - viz obr. 2.2. Označme Jacobiho matici $\mathbf{A} = \nabla\mathbf{h}(\mathbf{x}^*)$. Tato matice má v regulárním bodě plnou řádkovou hodnost. Označme $\mathbf{b}^T = \nabla f(\mathbf{x}^*)$. Pak, aby podle (2.12) a (2.13) soustavy $\mathbf{Ay} = \mathbf{0}$, $\mathbf{b}^T\mathbf{y} = 0$ měly shodná řešení, musí být vektor $\mathbf{b}^T$ lineární kombinací řádků matice $\mathbf{A}$. Z toho plyne, že $\nabla f(\mathbf{x}^*)$ je lineární kombinací gradientů $\mathbf{h}(\mathbf{x})$ v bodě $\mathbf{x}^*$. Proto platí následující věta:

Věta: *Nechť* $\mathbf{x}^*$ *je bod lokálního extrému* $f(\mathbf{x})$ *vzhledem k omezením* $\mathbf{h}(\mathbf{x}) = \mathbf{0}$. *Dále předpokládáme, že* $\mathbf{x}^*$ *je regulární bod omezení. Pak existuje vektor* $\boldsymbol{\lambda} \in E^m$ *že*

$$\nabla f(\mathbf{x}^*) + \boldsymbol{\lambda}^T\nabla\mathbf{h}(\mathbf{x}^*) = \mathbf{0}.$$

□

Předchozí podmínky můžeme vyjádřit také pomocí **Lagrangeovy funkce** $L(\mathbf{x}, \boldsymbol{\lambda})$

$$L(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T\mathbf{h}(\mathbf{x})$$

Nutné podmínky z předchozí věty tvrdí, že gradient Lagrangeovy funkce vzhledem k $\mathbf{x}$ je nulový v bodě $\mathbf{x}^*$ a omezující podmínka $\mathbf{h}(\mathbf{x}) = \mathbf{0}$ je ekvivalentní podmínce nulovosti gradientu Lagrangeovy funkce vzhledem k $\boldsymbol{\lambda}$, čili

$$\begin{aligned} \nabla_x L(\mathbf{x}, \boldsymbol{\lambda}) &= \nabla_x f(\mathbf{x}) + \boldsymbol{\lambda}^T\nabla_x\mathbf{h}(\mathbf{x}) = \mathbf{0} \\ \nabla_\lambda L(\mathbf{x}, \boldsymbol{\lambda}) &= \mathbf{h}(\mathbf{x}) = \mathbf{0} \end{aligned} \tag{2.14}$$

Příklad: V tomto příkladě ukážeme, že při nesplnění podmínek regularity neplatí nutné podmínky formulované v předchozí větě. Budeme hledat minimum funkce $f(\mathbf{x}) = x_1^2 + x_2^2 + x_3^2$ za podmínky $h_1(\mathbf{x}) = (x_1 - 2)^2 - x_2^2 = 0$, $h_2(\mathbf{x}) = x_2 = 0$. Omezující podmínky určují pevné hodnoty $x_1 = 2$ a $x_2 = 0$. Proto globální minimum je zřejmě rovno $\mathbf{x}^* = \begin{bmatrix} 2 & 0 & 0 \end{bmatrix}^T$. Nutné podmínky nulovosti gradientu Lagrangeovy funkce vzhledem k x_i vedou na soustavu rovnic ve tvaru

$$\begin{bmatrix} 4 \\ 0 \\ 0 \end{bmatrix} + \lambda_1 \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + \lambda_2 \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Tato soustava však nemá řešení pro žádné λ_i, neboť $\mathbf{x}^*$ není regulární bod. Hodnost Jacobiho matice je rovna jedné, ale omezení v naší úloze jsou dvě.

Příklad: Určete rozměry kvádru (krabice) maximálního objemu. Při tom je dána plocha c materiálu, ze kterého se krabice vyrobí. Máme tedy úlohu ve tvaru

$$\max \{x\ y\ z\ :\ 2(xy + xz + yz) = c\}$$

Lagrangeova funkce je rovna $L(x, y, z, \lambda) = xyz + \lambda(2(xy + xz + yz) - c)$. Nutné podmínky optima vedou na soustavu rovnic ve tvaru

$$\begin{aligned} yz + 2\lambda(y + z) &= 0 \\ xz + 2\lambda(x + z) &= 0 \\ xy + 2\lambda(x + y) &= 0 \end{aligned}$$

Vzájemným odečtením předchozích rovnic snadno odvodíme, že $x = y = z$ a z omezující podmínky dostaneme $x = y = z = \sqrt{\frac{c}{6}}$.

Příklad: Nalezněte rozdělení pravděpodobnosti s maximální entropií. Mějme tedy náhodnou veličinu x, která nabývá n hodnot $(x_1, \ldots, x_n)$, každou s pravděpodobností $(p_1, \ldots, p_n)$. Při tom platí, že pravděpodobnosti jsou nezáporné $p_i \geq 0$ a jejich součet je roven jedné $\sum p_i = 1$. Je také dána střední hodnota $\sum x_i p_i = m$ náhodné veličiny x.

Entropie $\mathcal{E}$ náhodného jevu je definována $\mathcal{E} = -\sum p_i \log p_i$. Lagrangeova funkce pro tuto úlohu je rovna $L(p_i, \lambda_1, \lambda_2) = -\sum p_i \log p_i + \lambda_1(\sum p_i - 1) + \lambda_2(\sum x_i p_i - m)$. Při tom jsme ignorovali omezení na nezápornost pravděpodobností p_i, neboť toto omezení není aktivní. Z požadavku nulovosti Lagrangeovy funkce vzhledem k p_i plyne rovnice

$$-\log p_i - 1 + \lambda_1 + \lambda_2 x_i = 0$$

Odtud zřejmě plyne $p_i = e^{\lambda_1 - 1 + \lambda_2 x_i}$. Rozdělení s maximem entropie je tedy exponenciální rozdělení. □

Nutné podmínky druhého řádu jsou uvedeny v následující větě:

Věta: *Nechť $\mathbf{x}^*$ je bod lokálního minima $\mathbf{f}(\mathbf{x})$ vzhledem k omezením $\mathbf{h}(\mathbf{x}) = \mathbf{0}$ a $\mathbf{x}^*$ je regulární bod omezení. Pak existuje vektor $\boldsymbol{\lambda} \in E^m$, že jsou splněny nutné podmínky prvního řádu $\nabla \mathbf{f}(\mathbf{x}^*) + \boldsymbol{\lambda}^T \nabla \mathbf{h}(\mathbf{x}^*) = \mathbf{0}$.*

Je-li $\mathbf{M}$ tečná rovina k omezením v bodě $\mathbf{x}^*$, $\mathbf{M} = \{\mathbf{y} : \nabla \mathbf{h}(\mathbf{x}^*)\mathbf{y} = \mathbf{0}\}$, pak Hessova matice

$$\mathbf{H}(\mathbf{x}^*) = \nabla^2 L = \nabla^2 f(\mathbf{x}) + \boldsymbol{\lambda}^T\ \nabla^2 \mathbf{h}(\mathbf{x}^*)$$

je pozitivně semidefinitní na $\mathbf{M}$*, to je*

$$\mathbf{y}^T \; H(\mathbf{x}^*) \; \mathbf{y} \geq 0, \qquad \forall \mathbf{y} \in \mathbf{M} \tag{2.15}$$

□

Poznámka: Uvědomme si, že výraz $\nabla^2 \mathbf{h}(\mathbf{x}^*)$, který se vyskytuje v Hessově matici, je třírozměrný tenzor. Proto druhý člen v Hessově matici vyjádříme ve tvaru

$$\boldsymbol{\lambda}^T \nabla^2 \mathbf{h}(\mathbf{x}^*) = \sum_{i=1}^{m} \lambda_i \nabla_x^2 h_i(\mathbf{x})$$

Při tom výraz $\nabla_x^2 h_i(\mathbf{x})$ je matice. □

Postačující podmínky dostaneme zesílením nutných podmínek druhého řádu.

Věta: *Nechť* $\mathbf{x}^*$ *splňuje omezení* $\mathbf{h}(\mathbf{x}^*) = 0$ *a existuje* $\boldsymbol{\lambda}$*, že platí* $\nabla \mathbf{f}(\mathbf{x}^*) + \boldsymbol{\lambda}^T \nabla \mathbf{h}(\mathbf{x}^*) = 0$*. Dále předpokládáme, že Hessova matice* $\mathbf{H}(\mathbf{x}^*)$ *je pozitivně definitní na tečné nadrovině* $\mathbf{M}$*. Pak* $\mathbf{x}^*$ *je bod ostrého lokálního minima funkce* $\mathbf{f}(\mathbf{x})$ *vzhledem k omezením* $\mathbf{h}(\mathbf{x}) = \mathbf{0}$*.* □

Povšimněme si, že zde není třeba dodávat podmínky regularity.

2.3.2 Citlivostní věta - stínové ceny

Uvažujme nyní, jak se mění kritérium, pokud měníme omezující podmínky. Odpověď na tento problém nám dá následující citlivostní věta:

Věta: *Nechť* f*,* $\mathbf{h} \in C^2$*. Mějme problém*

$$\min \{f(\mathbf{x}) \; : \; \mathbf{h}(\mathbf{x}) = \mathbf{b}\}$$

Pro $\mathbf{b} = \mathbf{0}$ *je řešením regulární bod* $\mathbf{x}^*$ *a odpovídající* $\boldsymbol{\lambda}$ *je Lagrangeův koeficient. Při tom v bodě* $\mathbf{x}^*$ *jsou splněny postačující podmínky druhého řádu pro ostré lokální minimum. Potom pro* $\mathbf{b} \in E^m$ *je* $\mathbf{x}(\mathbf{b})$ *spojitě závislé na* $\mathbf{b}$*, takové, že* $\mathbf{x}(\mathbf{0}) = \mathbf{x}^*$ *a* $\mathbf{x}(\mathbf{b})$ *je lokální minimum uvedeného problému. Potom*

$$\nabla_b f(\mathbf{x}(\mathbf{b}))|_{b=0} = -\boldsymbol{\lambda}^T \tag{2.16}$$

□

Předchozí vztah plyne z následujícího rozboru. Gradient kriteria vzhledem k vektoru $\mathbf{b}$ je roven

$$\nabla_b f(\mathbf{x}(\mathbf{b}))|_{b=0} = \nabla_x f(\mathbf{x})|_{x=x^*} \; \nabla_b \mathbf{x}(\mathbf{b})|_{b=0}.$$

Protože v tomto problému je omezení $\mathbf{h}(\mathbf{x}) = \mathbf{b}$, platí

$$\nabla_b \mathbf{h}(\mathbf{x}(\mathbf{b}))|_{b=0} = \; \mathbf{I} \; = \nabla_x \mathbf{h}(\mathbf{x})|_{x=x^*} \; \nabla_b \mathbf{x}(\mathbf{b})|_{b=0},$$

kde $\mathbf{I}$ je jednotková matice. Z nulovosti gradientu Lagrangeovy funkce plyne $\nabla_x f(\mathbf{x})|_{x=x^*} = -\boldsymbol{\lambda}^T \nabla_x \mathbf{h}(\mathbf{x})|_{x=x^*}$. Proto

$$\nabla_b f(\mathbf{x}(\mathbf{b}))|_{b=0} = -\boldsymbol{\lambda}^T \underbrace{\nabla_x \mathbf{h}(\mathbf{x})|_{x=x^*} \; \nabla_b \mathbf{x}(b)|_{b=0}}_{\mathbf{I}} = -\boldsymbol{\lambda}^T$$

Z předchozího tedy plyne, že velikost Lagrangeova koeficientu nám říká, jak rychle se mění kritérium při změně omezení. Přibližně tedy platí $\frac{\Delta f}{\Delta b_i} \doteq -\lambda_i^T$, to znamená, že při jednotkové změně i-tého omezení se hodnota kritéria změní o $(-\lambda_i)$. Velikost Lagrangeova koeficientu ukazuje tedy na důležitost příslušného omezení. Protože často kritérium vyjadřuje cenu, ukazují Lagrangeovy koeficienty na cenu příslušného omezení. Proto se Lagrangeovy koeficienty nazývají také **"stínové ceny"**.

Pokud je nulový některý Lagrangeův koeficient, znamená to, že změna příslušného koeficientu nevyvolá žádnou změnu kritéria. Toto omezení se nazývá **degenerované**.

2.3.3 Omezení s nerovnostmi

Mějme tedy následující problém:

$$\min \{f(\mathbf{x}) \ : \mathbf{h}(\mathbf{x}) = \mathbf{0} \ ; \ \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\} \tag{2.17}$$

Dle předchozího je definován regulární bod jako bod splňující omezení, u něhož gradienty aktivních omezení jsou lineárně nezávislé.

Definice: *Nechť* $\mathbf{x}^*$ *je bod splňující omezení a nechť* J *je množina indexů* j *pro která* $g_j(\mathbf{x}) = 0$ *(omezení jsou aktivní). Pak* $\mathbf{x}^*$ *je regulární bod omezení, jestliže gradienty* $\nabla h_i(\mathbf{x})$, $\nabla g_j(\mathbf{x})$, *pro* $1 \leq i \leq m$; $j \in J$ *jsou lineárně nezávislé.*

Nutné podmínky prvního řádu jsou určeny slavnou větou, kterou poprvé formulovali Kuhn a Tucker:

Věta: *Nechť* $\mathbf{x}^*$ *je bod relativního minima problému (2.17) a předpokládáme, že* $\mathbf{x}^*$ *je regulární bod omezení. Pak existuje vektor* $\boldsymbol{\lambda} \in E^m$ *a vektor* $\boldsymbol{\mu} \in E^p$, *že*

$$\begin{aligned} \nabla f(\mathbf{x}^*) + \boldsymbol{\lambda}^T \nabla \mathbf{h}(\mathbf{x}^*) + \boldsymbol{\mu}^T \nabla \mathbf{g}(\mathbf{x}^*) &= \mathbf{0} \\ \boldsymbol{\mu}^T \mathbf{g}(\mathbf{x}^*) &= 0 \\ \boldsymbol{\mu} &\geq \mathbf{0} \end{aligned} \tag{2.18}$$

Lagrangeův koeficient příslušející nerovnostním omezením musí být nezáporný a součin $\mu_j g_j(\mathbf{x})$ musí být roven nule. To znamená, že pro neaktivní omezení musí být příslušný Lagrangeův koeficient nulový. Pokud je omezení aktivní, může být odpovídající Lagrangeův koeficient libovolý (ale nezáporný). Zavedením Lagrangeovy funkce

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{h}(\mathbf{x}) + \boldsymbol{\mu}^T \mathbf{g}(\mathbf{x}) \tag{2.19}$$

můžeme předchozí podmínky zapisovat ve tvaru

$$\nabla_x L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \mathbf{0} \tag{2.20}$$

$$\nabla_\lambda L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \mathbf{0} \tag{2.21}$$

$$\nabla_\mu L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) \leq \mathbf{0} \tag{2.22}$$

$$\nabla_\mu L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) \ \boldsymbol{\mu} = 0 \tag{2.23}$$

$$\boldsymbol{\mu} \geq \mathbf{0} \tag{2.24}$$

První rovnice a čtvrtá a pátá nerovnice v předchozí soustavě jsou totožné se soustavou (2.18) a druhá rovnice a třetí nerovnice v předchozí soustavě popisují omezení úlohy.

Předchozí Kuhnovy - Tuckerovy nutné podmínky snadno odvodíme z nutných podmínek pro úlohy s omezením ve tvaru rovnosti. Upravíme tedy naši úlohu na tento tvar podle následujícího postupu. Zavedeme si pomocnou proměnnou $\mathbf{y}$ a dostaneme ekvivalentní úlohu ve tvaru

$$\min \{f(\mathbf{x}) \ : \mathbf{h}(\mathbf{x}) = \mathbf{0} \ ; \ \mathbf{g}(\mathbf{x}) + \mathbf{y}^{[2]} = \mathbf{0}\}$$

kde $\mathbf{y}^{[2]} = \left[\ y_1^2, \ y_2^2, \ \ldots, \ y_p^2 \ \right]^T$ je nezáporný vektor s kvadratickými složkami. Lagrangeova funkce pro tuto úlohu je

$$\overline{L}(\mathbf{x}, \mathbf{y}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{h}(\mathbf{x}) + \boldsymbol{\mu}^T (\mathbf{g}(\mathbf{x}) + \mathbf{y}^{[2]})$$

Podmínku nezápornosti Lagrangeova koeficientu $\boldsymbol{\mu}$ odvodíme pomocí citlivosti kritéria na změnu omezení. Omezení $\mathbf{g}(\mathbf{x}) \leq \mathbf{0}$ určuje spolu s omezením $\mathbf{h}(\mathbf{x}) = \mathbf{0}$ množinu přípustných řešení $\mathbf{X}$. Uvažujme nyní nerovnostní omezení ve tvaru $\mathbf{g}(\mathbf{x}) \leq \mathbf{b}$, kde vektor $\mathbf{b}$ má nezáporné složky. Toto omezení spolu s omezením $\mathbf{h}(\mathbf{x}) = \mathbf{0}$ určuje novou množinu přípustných řešení $\overline{\mathbf{X}}$. Při tom platí, že $\overline{\mathbf{X}} \supset \mathbf{X}$ protože vlivem nezápornosti vektoru $\mathbf{b}$ jsme uvolnili omezení. Minimum na větší množině nemůže být větší než minimum na podmnožině. Proto platí

$$\min_{x \in \overline{X}} f(\mathbf{x}) \leq \min_{x \in X} f(\mathbf{x})$$

Přírůstek kritéria při změně omezení $\min_{x \in \overline{X}} f(\mathbf{x}) - \min_{x \in X} f(\mathbf{x}) \leq 0$ je tedy nekladný. Při tom podle citlivostní věty platí $\dfrac{\partial f(\mathbf{x})}{\partial \mathbf{b}} = -\boldsymbol{\mu}^T$. Z předchozího diferenciálního vztahu plyne pro přírůstek kritéria vztah

$$\Delta f(\mathbf{x}) = -\boldsymbol{\mu}^T \Delta \mathbf{b} = -\boldsymbol{\mu}^T \mathbf{b}$$

Odvodili jsme, že pro nezáporné $\mathbf{b} \geq \mathbf{0}$ je přírůstek kritéria nekladný ($\Delta f(\mathbf{x}) \leq 0$) a proto Lagrangeův koeficient $\boldsymbol{\mu}$ musí být nezáporný $\boldsymbol{\mu} \geq \mathbf{0}$.

Podmínku $\boldsymbol{\mu}^T \mathbf{g}(\mathbf{x}^*) = \mathbf{0}$ odvodíme z nutné podmínky minima, a totiž, že derivace Lagrangeovy funkce $\overline{L}$ vzhledem ke všem proměnným musí být nulové. Proto po složkách musí platit

$$\frac{\partial \overline{L}}{\partial y_j} = 2\mu_j y_j = 0$$

Platí tedy po složkách $\mu_j y_j = 0$ čili také $\mu_j y_j^2 = 0$. Z omezení plyne, že $y_j^2 = -g_j(\mathbf{x})$ a proto po složkách platí $\mu_j g_j(\mathbf{x}) = 0$. Odtud plyne podmínka $\boldsymbol{\mu}^T \mathbf{g}(\mathbf{x}^*) = 0$.

Nutné podmínky druhého řádu jsou formulovány v následující větě:

Věta: *Předpokládáme, že funkce f, $\mathbf{h}$, $\mathbf{g} \in C^2$ a $\mathbf{x}^*$ je regulární bod omezení. Je-li $\mathbf{x}^*$ bod relativního minima, pak existují vektory $\boldsymbol{\lambda}$ a $\boldsymbol{\mu} \geq \mathbf{0}$, že platí Kuhnovy - Tuckerovy podmínky a Hessova matice*

$$\nabla^2 L(\mathbf{x}^*) = \nabla^2 f(\mathbf{x}^*) + \boldsymbol{\lambda}^T \nabla^2 \mathbf{h}(\mathbf{x}^*) + \boldsymbol{\mu}^T \nabla^2 \mathbf{g}(\mathbf{x}^*) \tag{2.25}$$

je pozitivně semidefinitní na tečné nadrovině aktivních omezení. □

Pro postačující podmínky druhého řádu nestačí aby Hessova matice $\nabla^2 L(\mathbf{x}^*)$ byla v minimu pozitivně definitní na tečné nadrovině aktivních omezení, ale je třeba, aby $L(\mathbf{x}^*)$ byla pozitivně definitní na podprostoru

$$\overline{M} = \ \{ \ \mathbf{y} \ : \nabla \mathbf{h}(\mathbf{x}^*)\mathbf{y} = \mathbf{0} \ ; \ \nabla g_j(\mathbf{x}^*)\mathbf{y} = 0 \ ; \ j \in \overline{J}\}$$

kde

$$\overline{J} = \{\ j\ : g_j(\mathbf{x}^*) = 0\ ;\ \mu_j > 0\}$$

Vyžadujeme tedy navíc, aby Lagrangeovy koeficienty μ_j u aktivních omezení byly kladné (a ne pouze nezáporné). Vylučujeme tedy degenerovaná omezení.

Poznámka: Existují čtyři modifikace základní úlohy nelineárního programování

$$\begin{array}{ll} (a) & \min\{f(\mathbf{x})\ :\ \mathbf{g}(\mathbf{x}) \le \mathbf{0}\} \\ (b) & \max\{f(\mathbf{x})\ :\ \mathbf{g}(\mathbf{x}) \le \mathbf{0}\} \\ (a) & \min\{f(\mathbf{x})\ :\ \mathbf{g}(\mathbf{x}) \ge \mathbf{0}\} \\ (a) & \max\{f(\mathbf{x})\ :\ \mathbf{g}(\mathbf{x}) \ge \mathbf{0}\} \end{array}$$

Lagrangeovu funkci pro všechny úlohy volíme ve tvaru

$$L(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{g}(\mathbf{x})$$

Jak zní Kuhn - Tuckerovy podmínky pro tyto čtyři podobné úlohy? Podmínka

$$\nabla_x L(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{0}$$

platí pro všechny čtyři úlohy, určuje totiž stacionární body. Podmínka

$$\nabla_\lambda L(\mathbf{x}, \boldsymbol{\lambda}) \le \mathbf{0}$$

určuje omezení a proto platí pro prvé dvě úlohy. Pro třetí a čtvrtou úlohu platí obrácené znaménko. Podmínka

$$\nabla_\lambda L(\mathbf{x}, \boldsymbol{\lambda})\ \boldsymbol{\lambda} = \mathbf{0}$$

platí pro všechny čtyři úlohy, neboť je odvozena z podmínek stacionárnosti. Podmínka nezápornosti Lagrangeových koeficientů $\boldsymbol{\lambda} \ge \mathbf{0}$ platí pro minimalizační úlohy (problém (a) a (c)). Pro úlohy maximalizace platí obrácená nerovnost.

Poznámka: Jsou-li v úloze nelineárního programování omezení na nezápornost některých proměnných, lze Lagrangeovy koeficienty u těchto omezení eliminovat. Ukážeme to na následujícím příkladu: Mějme úlohu minimalizace

$$\min\{f(\mathbf{x})\ :\ x_j \ge 0; j \in J\}$$

Omezení převedeme do námi užívaného tvaru $g_j(\mathbf{x}) = (-x_j) \le 0\ ;\ j \in J$. Kuhnovy-Tuckerovy podmínky pro naši úlohu jsou

$$\frac{\partial f}{\partial x_i} + \sum_{j\in J} \lambda_j \frac{\partial(-x_j)}{\partial x_i} = \frac{\partial f}{\partial x_i} + \sum_{j\in J} \lambda_j(-\delta_{ij}) = 0$$
$$x_j \ge 0\ ;\ \lambda_j x_j = 0\ ;\ \lambda_j \ge 0\ ;\ j \in J$$

Odtud po jednoduchých úpravách plynou podmínky, ve kterých nejsou Lagrangeovy koeficienty. Upravené Kuhnovy - Tuckerovy podmínky pro naši úlohu jsou

$$\frac{\partial f}{\partial x_j} = 0 \qquad \text{pro}\ \ j \notin J$$
$$\frac{\partial f}{\partial x_j} \ge 0\ ; \quad \frac{\partial f}{\partial x_j} x_j = 0\ ; \quad x_j \ge 0 \qquad \text{pro}\ \ j \in J$$

□

2.4 Sedlový bod a dualita

2.4.1 Sedlové vlastnosti Lagrangeovy funkce

Při řešení úlohy nelineárního programování se ukazuje, že Lagrangeova funkce má v optimálních bodech $\mathbf{x}^*$, $\boldsymbol{\lambda}^*$ tzv. **sedlový bod**. Proto si nejprve vysvětlíme, co to je sedlový bod a potom si uvedeme příslušná tvrzení.

Mějme tedy funkci $s(\mathbf{x}, \mathbf{y})$ definovanou na $\mathbf{X} \times \mathbf{Y}$. Říkáme, že funkce $s(\mathbf{x}, \mathbf{y})$ má v bodě $\mathbf{x}^*$,$\mathbf{y}^*$ sedlový bod, platí-li pro všechny $(\mathbf{x}, \mathbf{y}) \in \mathbf{X} \times \mathbf{Y}$

$$s(\mathbf{x}^*, \mathbf{y}) \leq s(\mathbf{x}^*, \mathbf{y}^*) \leq s(\mathbf{x}, \mathbf{y}^*)$$

respektive

$$s(\mathbf{x}^*, \mathbf{y}) \geq s(\mathbf{x}^*, \mathbf{y}^*) \geq s(\mathbf{x}, \mathbf{y}^*)$$

V prvním případě říkáme, že se jedná o sedlový bod typu minimaxu a v druhém případě o sedlový bod typu maximinima. Uvědomme si, že obecně platí

$$\max_{x \in X} \min_{y \in Y} g(\mathbf{x}, \mathbf{y}) \leq \min_{y \in Y} \max_{x \in X} g(\mathbf{x}, \mathbf{y}) \tag{2.26}$$

Předchozí tvrzení snadno dokážeme. Zřejmě platí

$$\min_{y \in Y} g(\mathbf{x}, \mathbf{y}) \leq g(\mathbf{x}, \mathbf{y}), \quad \text{pro } \mathbf{x} \in \mathbf{X},\ \mathbf{y} \in \mathbf{Y}$$

Nerovnost se neporuší maximalizací obou stran vzhledem k $\mathbf{x}$, proto

$$\max_{x \in X} \min_{y \in Y} g(\mathbf{x}, \mathbf{y}) \leq \max_{x \in X} g(\mathbf{x}, \mathbf{y}), \quad \text{pro } \mathbf{y} \in \mathbf{Y}$$

Protože předchozí nerovnice platí pro všechna $\mathbf{y} \in \mathbf{Y}$, platí i pro $\mathbf{y}$, pro kterou je pravá strana minimální. Odtud plyne vztah (2.26).

Mějme následující problém

$$\min \{f(\mathbf{x}) \ : \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\ ;\ \mathbf{x} \geq \mathbf{0}\} \tag{2.27}$$

Potom platí tvrzení:

Věta: *Jestliže* $\mathbf{x}^* \geq \mathbf{0}$, $\boldsymbol{\lambda}^* \geq \mathbf{0}$ *je sedlovým bodem Lagrangeovy funkce*

$$L(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{g}(\mathbf{x}), \tag{2.28}$$

pak $\mathbf{x}^*$ *je řešením úlohy (2.27).*

Předchozí tvrzení je postačující podmínkou optima. Není třeba žádných předpokladů o regularitě. Postačující podmínku snadno dokážeme. Sedlový bod Lagrangeovy funkce splňuje nerovnice

$$L(\mathbf{x}, \boldsymbol{\lambda}^*) \geq L(\mathbf{x}^*, \boldsymbol{\lambda}^*) \geq L(\mathbf{x}^*, \boldsymbol{\lambda}) \tag{2.29}$$

pro $\boldsymbol{\lambda} \geq \mathbf{0}$, $\boldsymbol{\lambda}^* \geq \mathbf{0}$, $\mathbf{x} \in \mathbf{X}$.

Pravá nerovnost zaručuje, že $\mathbf{x}^*$ je přípustné řešení. Pokud by totiž $\mathbf{x}^*$ nebylo přípustné, pak by některé omezení bylo $g_j \geq 0$ a pak v Lagrangeově funkci $L(\mathbf{x}, \boldsymbol{\lambda})$ dle (2.28) bychom vždy mohli volit $\lambda_j \geq 0$ takové, aby pravá nerovnost v (2.29) byla porušena.

Je-li $\mathbf{x}^*$ přípustné a $\lambda_j^* > 0$ a $g_j(\mathbf{x}) \leq 0$, pak pro splnění $\lambda_j g_j(\mathbf{x}) \leq \lambda_j^* g_j(\mathbf{x})$ pro všechny $\lambda_j \geq 0$ musí být $g_j(\mathbf{x}) = 0$.

Je-li $\mathbf{x}^*$ přípustné ale $\lambda_j^* = 0$, pak nerovnost $\lambda_j g_j(\mathbf{x}) \leq \lambda_j^* g_j(\mathbf{x}) = 0$ je splněna pro libovolné $\lambda_j \geq 0$.

Pravá nerovnice v (2.29) zaručuje, že $\mathbf{x}^*$ je přípustné a Lagrangeova funkce je rovna $f(\mathbf{x})$, neboť $\lambda_j^* g_j(\mathbf{x}) = 0$ pro všechna j. Platí tedy

$$L(\mathbf{x}, \boldsymbol{\lambda}^*) = \{\ f(\mathbf{x})\ :\ \mathbf{g}(\mathbf{x}) \leq \mathbf{0},\ \mathbf{x} \geq \mathbf{0}\}$$

Levá nerovnost v (2.29) zaručuje, že $\mathbf{x}$ je řešením úlohy (2.27).

Abychom mohli dokázat nutnost věty o sedlovém bodě, je nutno doplnit další předpoklady o regularitě a konvexnosti.

Úloha konvexní optimalizace je úloha, v níž jsou kritérium $f(\mathbf{x})$ i funkce $\mathbf{g}(\mathbf{x})$ (v omezeních $\mathbf{g}(\mathbf{x}) \leq \mathbf{0}$) konvexními funkcemi. Omezení ve tvaru rovnosti je lineární. Potom množina přípustných řešení $\mathbf{X}$ je konvexní množina. V omezeních $g_j(\mathbf{x}) \leq 0$ stačí uvažovat pouze nelineární funkce $g_j(\mathbf{x})$.

Existence sedlového bodu za předpokladu regularity a konvexnosti je nutnou a postačující podmínkou řešení optimalizační úlohy.

2.4.2 Dualita úloh nelineárního programování

Nyní si povšimneme problému duality v úlohách nelineárního programování. Mějme tedy úlohu

$$\min\ \{f(\mathbf{x})\ : \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\} \tag{2.30}$$

Lagrangeova funkce je pro tuto úlohu zřejmě $L(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{g}(\mathbf{x})$. Vytvořme následující dvě funkce

$$\varphi(\mathbf{x}) = \max_{\lambda \geq 0} L(\mathbf{x}, \boldsymbol{\lambda}) \tag{2.31}$$

$$\psi(\boldsymbol{\lambda}) = \min_{x} L(\mathbf{x}, \boldsymbol{\lambda}), \tag{2.32}$$

kde $\mathbf{x}$ není omezeno.

Na funkcích $\varphi(\mathbf{x})$ a $\psi(\boldsymbol{\lambda})$ si můžeme definovat dvě následující úlohy:

$$(A) \qquad \min_{x} \varphi(\mathbf{x}) = \min_{x} \max_{\lambda \geq 0} L(\mathbf{x}, \boldsymbol{\lambda}) \tag{2.33}$$

$$(B) \qquad \max_{\lambda \geq 0} \psi(\boldsymbol{\lambda}) = \max_{\lambda \geq 0} \min_{x} L(\mathbf{x}, \boldsymbol{\lambda}). \tag{2.34}$$

Snadno ukážeme, že úloha (A) je totožná s původní úlohou (2.30). Platí totiž

$$\min_{x \in X}\ f(\mathbf{x}) = \min_{x} \varphi(\mathbf{x})$$

neboť volbou $\boldsymbol{\lambda} \geq 0$

$$\begin{aligned} \varphi(\mathbf{x}) &= f(\mathbf{x}) \quad \text{pro} \quad \mathbf{x} \in X \\ \varphi(\mathbf{x}) &= \infty \quad \text{pro} \quad \mathbf{x} \notin X \end{aligned}$$

Proto platí

$$\min_{x \in X} f(\mathbf{x}) = \min_{x} \max_{\lambda \geq 0} L(\mathbf{x}, \boldsymbol{\lambda}) \tag{2.35}$$

Úloha (A) je totožná s původní úlohou (2.30) a proto se nazývá **primární úloha**. Úloha (B) je **duální úloha**. Mezi primární a duální úlohou nelineárního programování platí vztah - viz (2.26)

$$\text{(Primární úloha)} \qquad \min_{x} \max_{\lambda \geq 0} L(\mathbf{x}, \boldsymbol{\lambda}) \;\geq\; \max_{\lambda \geq 0} \min_{x} L(\mathbf{x}, \boldsymbol{\lambda}) \qquad \text{(Duální úloha)} \tag{2.36}$$

Pro úlohu konvexního programování při splnění podmínek regularity platí věta o sedlovém bodě, pak

$$\min_{x} \max_{\lambda \geq 0} L(\mathbf{x}, \boldsymbol{\lambda}) = \max_{\lambda \geq 0} \min_{x} L(\mathbf{x}, \boldsymbol{\lambda}) = L(\mathbf{x}^*, \boldsymbol{\lambda}^*) = f(\mathbf{x}^*) \tag{2.37}$$

a primární i duální úloha mají stejné řešení, které je totožné s řešením původní úlohy.

Duální úlohu můžeme upravit. Protože $\mathbf{x}$ není omezeno, platí

$$\min_{x} L \quad \Rightarrow \quad \frac{\partial L}{\partial \mathbf{x}} = \frac{\partial f}{\partial \mathbf{x}} + \boldsymbol{\lambda}^T \frac{\partial \mathbf{g}}{\partial \mathbf{x}} = \mathbf{0}$$

Duální úlohu můžeme tedy vyjádřit ve tvaru

$$\max_{\lambda} \left\{ f(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{g}(\mathbf{x}) \;:\; \frac{\partial f}{\partial \mathbf{x}} + \boldsymbol{\lambda}^T \frac{\partial \mathbf{g}}{\partial \mathbf{x}} = \mathbf{0} \;;\; \boldsymbol{\lambda} \geq \mathbf{0} \right\} \tag{2.38}$$

Dualita úloh matematického programování je velmi důležitá, neboť řešením duální úlohy se k optimu blížíme zdola. Naopak řešením primární úlohy se k optimu blížíme shora. Existuje celá řada algoritmů nelineárního programování, které jsou založeny na sedlových vlastnostech Lagrangeovy funkce.

Příklad: Nalezněte duální úlohu k úloze lineárního programování.

Nejběžnější tvar úlohy lineárního programování je

$$\max \left\{ \mathbf{c}^T \mathbf{x} \;:\; \mathbf{A}\mathbf{x} \leq \mathbf{b},\; \mathbf{x} \geq \mathbf{0} \right\} \tag{2.39}$$

Nejprve upravíme tuto úlohu na úlohu minimalizace

$$\min \left\{ -\mathbf{c}^T \mathbf{x} \;:\; \mathbf{A}\mathbf{x} - \mathbf{b} \leq \mathbf{0},\; -\mathbf{x} \leq \mathbf{0} \right\}$$

Lagrangeova funkce pro tuto úlohu je

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = -\mathbf{c}^T \mathbf{x} + \boldsymbol{\lambda}^T (\mathbf{A}\mathbf{x} - \mathbf{b}) + \boldsymbol{\mu}^T (-\mathbf{x})$$

Duální úloha je tedy

$$\max_{\lambda \geq 0, \mu \geq 0} \min_{x} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \max_{\lambda, \mu} \{ L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) : \nabla_x L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \mathbf{0}, \boldsymbol{\lambda} \geq \mathbf{0}, \boldsymbol{\mu} \geq \mathbf{0} \} \tag{2.40}$$

Lagrangeovu funkci L můžeme upravit do tvaru

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \mathbf{x}^T(-\mathbf{c} + \mathbf{A}^T\boldsymbol{\lambda} - \boldsymbol{\mu}) - \boldsymbol{\lambda}^T\mathbf{b}$$

Z $\nabla_x L = \mathbf{0}$ plyne $(-\mathbf{c} + \mathbf{A}^T\boldsymbol{\lambda} - \boldsymbol{\mu}) = \mathbf{0}$. Odtud

$$\mathbf{A}^T\boldsymbol{\lambda} - \mathbf{c} = \boldsymbol{\mu} \geq \mathbf{0}$$

neboli $\mathbf{A}^T\boldsymbol{\lambda} - \mathbf{c} \geq \mathbf{0}$. Lagrangeova funkce se podle předchozího zjednoduší na

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = -\boldsymbol{\lambda}^T\mathbf{b}$$

Duální úloha je tedy

$$\max \{ \ -\boldsymbol{\lambda}^T\mathbf{b} \ : \mathbf{A}^T\boldsymbol{\lambda} \geq \mathbf{c}, \ \boldsymbol{\lambda} \geq \mathbf{0} \ \}$$

neboli

$$\min \{ \ \boldsymbol{\lambda}^T\mathbf{b} \ : \mathbf{A}^T\boldsymbol{\lambda} \geq \mathbf{c}, \ \boldsymbol{\lambda} \geq \mathbf{0} \ \}. \tag{2.41}$$

Řešení primární i duální úlohy jsou totožná. Při velkém počtu omezení $m \gg n$ může být duální úloha mnohem jednodušší. □

2.5 Vícekriteriální optimalizace

V reálných rozhodovacích situacích často sledujeme více kritérií, podle nichž hodnotíme nejlepší varianty řešení. Zde je problém správně definovat úlohu optimalizace v případě vícekriteriálnosti.

V tomto odstavci uvedeme některé myšlenky týkající se tohoto problému. Již při formulování kritérií se mohou vyskytnout problémy, některá hodnotící kritéria jsou často formulována pouze kvalitativně. Často vyšší zisk je spojen s vyšším rizikem - zde se uplatní teorie užitku, která přiřazuje variantám řešení číselné ohodnocení (užitek) tak, aby hodnota užitku byla v souladu s preferencemi rozhodujícího subjektu v tom smyslu, že vyšší užitek znamená vyšší preferenci.

Pro vícekriteriální optimalizaci se používají alternativní pojmy jako víceaspektní, víceatributní či vektorová optimalizace. To vede na vektorové či kompromisní programování.

Často se snažíme vícekriteriální optimalizační problém převést na **problém skalární s jedním optimalizačním kritériem**. To je možné následujícími způsoby:

- volbou vah u jednotlivých kritérií
- metodou cílového programování
- lexikografickým uspořádáním kritérií
- minimaxovou optimalizací

První možností je volba vah jednotlivých kritérií. Tím problém vícekriteriální optimalizace převedeme na problém optimalizace jediného kritéria, tvořeného lineární kombinací jednotlivých kritérií, v němž váhy vyznačují preference jednotlivých kritérií. Označíme-li jednotlivá kritéria $f_1(\mathbf{x})$ až $f_p(\mathbf{x})$ pak jediné kritérium je v tomto případě rovno

$$f(\mathbf{x}) = \alpha_1 f_1(\mathbf{x}) + \cdots + \alpha_p f_p(\mathbf{x})$$

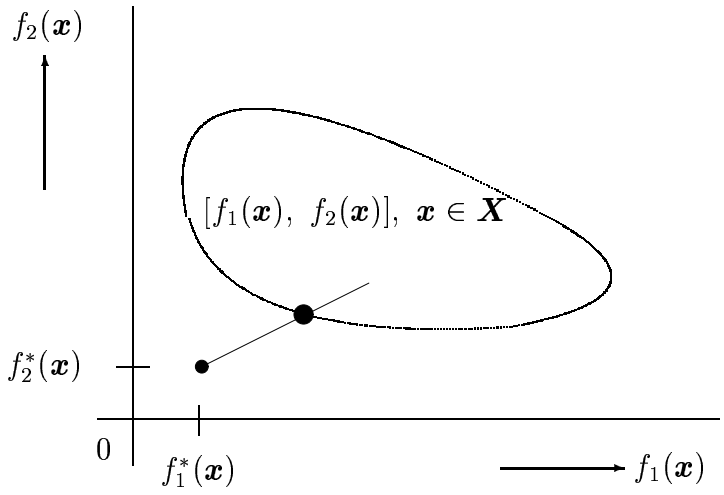
kde koeficienty α_i vyjadřují preference jednotlivých kritérií.

Druhou možností je stanovení cíle, kterého se snažíme dosáhnout ve všech kritériích. Protože stanoveného cíle nelze dosáhnout současně ve všech kritériích, pak se snažíme alespoň minimalizovat vzdálenost konkrétního řešení od cíle. Tato metoda má jednoduchou geometrickou interpretaci. Označme jako $f_i^*(\mathbf{x})$ cíl, který chceme dosáhnout v i-tém kritériu. Metoda cílového programování vede na úlohu

$$\min\{\ \lambda\ : f_i(\mathbf{x}) - w_i\lambda\ \leq\ f_i^*(\mathbf{x}),\ ,i = 1,\ldots,p,\ \mathbf{x} \in \mathbf{X})\ \} \tag{2.42}$$

kde λ je skalár, na který neklademe žádné omezení a $w_i > 0$ jsou váhové koeficienty. Při tom λw_i vyjadřuje rozdíl mezi stanoveným cílem a skutečnou hodnotou příslušného kritéria. To znamená, že pokud některé kritérium (např. j-té) chceme co nejvíce přiblížit k cíli, je třeba volit u něho výrazně menší váhový koeficient $w_j << w_i$ pro všechna $i \neq j$.

Přípustná řešení $\mathbf{x} \in \mathbf{X}$ vytvářejí v prostoru kritérií množinu možných hodnot kritérií. My podle předchozího kritéria hledáme takové přípustné řešení, které leží v průsečíku polopřímky vycházející z cílového bodu $f_i^*(\mathbf{x})$ s množinou přípustných hodnot kritérií (v obr 2.3 je tento bod vyznačen černou tečkou). Směrnice polopřímky vycházející z cílového bodu $f_i^*(\mathbf{x})$ je určena váhami w_i - viz obr. 2.3 pro dvě kritéria.



Obrázek 2.3: Optimální řešení podle cílového programování

Třetí možností je uspořádat kritéria podle důležitosti a optimalizovat první kritérium a z množiny optimálních řešení (je-li celá množina optimálních řešení pro první kritérium) vybrat řešení takové, které minimalizuje druhé kritérium v pořadí atd.

Varianta této metody je tzv. **metoda s ε-omezením**. V této metodě minimalizujeme nějaké vybrané preferenční kritérium a ostatní kritéria podrobíme omezením, pak

$$\min_{\mathbf{x}\in\mathbf{X}} \{f_s(\mathbf{x})\ : f_i(\mathbf{x}) \leq \varepsilon_i, \quad i = 1,2,\ldots,p, \quad i \neq s\} \tag{2.43}$$

Zde je problém volby omezení ε_i. Další nevýhodou je, že omezení jsou pevná, což zřídka vyjadřuje skutečné požadavky.

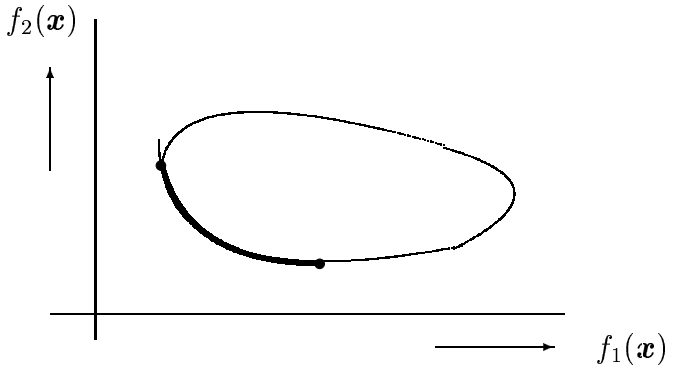
Minimaxová optimalizace vede na úlohu optimalizace, ve které hledáme takové přípustné řešení, v němž je maximální hodnota nějakého kritéria minimální

$$\min_{\mathbf{x}\in\mathbf{X}} \max_{i} f_i(\mathbf{x}),$$

To je vlastně totožný problém jako při cílovém programování, kde volíme váhy w_i všechny stejné a cíl je v počátku. Tato formulace vícekriteriální optimalizace připomíná problémy z teorie her.

□

Pokud nechceme převést problém vícekriteriální optimalizace na skalární problém, je třeba postupovat jiným způsobem. Budeme hledat **nedominované varianty**, to jsou takové varianty přípustného řešení, že neexistuje jiná přípustná varianta, při které některé kritérium dosahuje nižší hodnoty a ostatní kritéria nerostou - jedno kritérium lze zlepšit pouze na úkor zhoršení jiného kritéria. Nedominované varianty se také nazývají **paretovsky optimální**, nebo **neinferiorní** či **eficientní varianty**. Na obr. 2.4 je množina paretovsky optimálních řešení vyznačena tučně. Pokud je jediná nedominovaná varianta, pak ji můžeme označit za optimální variantu. Pokud je nedominovaných variant více,



Obrázek 2.4: Paretovsky optimální varianty řešení

pak je třeba dalším postupem zúžit výběr. Zde neexistuje jediný exaktní postup řešení. Můžeme volit hypotetické ideální hodnoty jednotlivých kritérií a nejhorší tzv. bazální hodnoty kritérií. Reálné přípustné varianty řešení se pohybují mezi těmito krajnostmi.

Příklad: Problémy vícekriteriální optimalizace si ilustrujeme na jednoduché úloze určení optimální skládky odpadů. Je třeba vybrat optimální lokalitu skládky - jednu ze čtyř možností, které označíme x_1 až x_4. Při tom hodnotíme pět kritérií - zábor půdy v hektarech, investiční náklady v mil.Kč, negativní důsledky pro okolí ve stupnici od 1 (velmi negativní) do 4 (nepatrné) , negativní důsledky na kvalitu vody (také od 1 do 4) a konečně kapacitu skládky v mil. tun odpadu. Velikosti jednotlivých kritérií pro různé lokality jsou vyneseny v tabulce 2.1.

Při optimalizaci se budeme snažit minimalizovat zábor půdy i investiční náklady, ale naopak maximalizovat v hodnotící stupnici negativních důsledků i maximalizovat kapacitu skládky.

Aby všechna kritéria byla minimalizována, upravíme poslední tři sloupce předchozí tabulky. Úpravu provedeme tak, že v příslušném sloupci nalezneme maximální prvek a

Tabulka 2.1: Skládka odpadků - hodnocení variant

	zábor půdy	investiční náklady	neg. důsl. pro okolí	neg. důsl. pro vodu	kapacita
x_1	6	1.2	4	2	6
x_2	11.2	14.4	2	2	4.5
x_3	1.9	4.8	2	4	7.5
x_4	6.4	13.4	2	2	4.5

Tabulka 2.2: Skládka odpadků - úprava hodnocení variant

	zábor půdy	investiční náklady	neg. důsl. $(f_3)_{max} - f_3$	neg. důsl. $(f_4)_{max} - f_4$	kapacita $(f_5)_{max} - f_5$
x_1	6	1.2	0	2	1.5
x_2	11.2	14.4	2	2	3
x_3	1.9	4.8	2	0	0
x_4	6.4	13.4	2	2	3

nové hodnocení bude rovno rozdílu maximálního prvku ve sloupci a příslušného prvku. Dostaneme tak novou tabulku 2.2, ve které již chceme všechna hodnocení minimalizovat Je zřejmé, že strategie x_1 dominuje strategii x_2 i x_4 a naopak strategie x_3 dominuje strategii x_2 i x_4. Při tom strategie x_1 a x_3 nejsou vzájemně dominovány. Nedominované strategie (Paretovsky optimální strategie) jsou tedy dvě strategie x_1 a x_3. Pro výběr jediné optimální strategie nemáme žádné objektivní hledisko. □

Již z tohoto příkladu je zřejmé, že výběr optimální varianty není jednoznačný a problém má značný subjektivní motiv. Jistě budou existovat různé preference i případné lokální zájmy, které lze i zdůvodnit vhodným výběrem argumentů. Je zřejmé, že každý reálný rozhodovací problém je vícekriteriální.

2.6 Příklady

1. Nalezněte minimum funkce

$$f(\mathbf{x}) = \frac{1}{2}\left(\frac{x_1^2}{a^2} + v\frac{x_2^2}{b^2}\right)$$

při omezení

$$h(\mathbf{x}) = x_1 + \alpha x_2 + \beta = 0$$

Určete geometrickou interpretaci této úlohy a řešte ji pro různé a, b, α, β. Nalezněte podmínky řešitelnosti.

2. Určete výšku a průměr válcové nádrže maximálního objemu při daném povrchu.

3. Pomocí Kuhnových - Tuckerových podmínek nalezněte stacionární body funkce

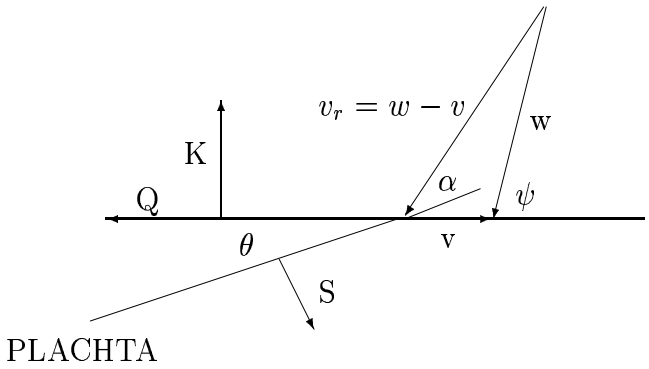
$$f(\mathbf{x}) = x_1^3 + 2x_2^2 + 6 - 10x_1 + 2x_2^3$$

vzhledem k omezením

$$x_1 x_2 \leq 10, \qquad x_1 \geq 0, \qquad x_2 \geq 0$$

a určete jejich charakter.

4. Optimalizace úhlu natočení plachty a kursu plachetnice.
Zjednodušené schéma sil působících na plachetnici při pohybu proti větru je na obr. 2.5., kde v je rychlost plachetnice, w je rychlost větru, ψ úhel mezi směrem větru a rychlostí loďky, θ je úhel plachty vzhledem k ose loďky, S je aerodynamická síla působící na loďku vlivem větru, Q je síla odporu loďky, K je kýlová síla, v_r je relativní rychlost větru vůči plachetnici. Aerodynamická síla S se rozkládá na



Obrázek 2.5: Síly působící na plachetnici

dvě kolmé síly. Jedna složka působící ve směru pohybu loďky (to je hnací síla) se vyrovnává se silou odporu loďky Q (tření při pohybu ve vodě). Druhá složka síly S se vyruší s kýlovou silou K a způsobuje náklon plachetnice, což zanedbáme. Aerodynamická síla S závisí na relativní rychlosti větru v_r a úhlu α, pod kterým vítr působí na plachtu. Předpokládejme, že platí

$$S = c_1 v_r^2 \sin\alpha$$

Síla odporu loďky Q závisí na rychlosti loďky podle vztahu $Q = c_2 v^2$. Ukažte, že
a) při zadaném úhlu ψ je rychlost v maximální pro $\alpha = \theta$;
b) Při plavbě po větru ($\psi = 180^o$) je maximální rychlost při $\alpha = 90^o$ a je rovna

$$v_{max} = w\frac{c}{1+c}, \qquad c^2 = \frac{c_1}{c_2}$$

c) maximální rychlost proti větru (veličina $v\cos\psi$) je rovna $w\frac{c}{4}$ a dosáhneme ji při úhlech $\psi = 45^o$, $\theta = ((c+2)^2+4)^{-\frac{1}{2}}$. Při tom předpokládáme, že úhly α a θ jsou malé a proto $\sin\alpha \doteq \alpha$, $\sin\theta \doteq \theta$, $\cos\alpha \doteq 1$, $\cos\theta \doteq 1$.

5. Vypočtěte optimální konstantu r_0 proporcionálního regulátoru, který v regulačním obvodu reguluje systém s přenosem

$$G(s) = \frac{5}{s(s+1)^5}.$$

Vstupní signál je skok řízení $w(t) = 1(t)$. Kritérium kvality řízení je

$$J(r_0) = \int_0^\infty \left(e^2(t) + 3u^2(t)\right)\ dt$$

kde $e(t)$ je regulační odchylka a $u(t)$ je akční veličina.

6. Mějme stejný regulační obvod jako v předchozím příkladě. Řízení je nulové, ale na vstupu systému působí porucha, kterou můžeme považovat za náhodný signál blízký bílému šumu. Navrhněte optimální konstantu proporcionálního regulátoru, při které je rozptyl regulační odchylky minimální.

7. Je dán diskrétní systém popsaný stavovými rovnicemi

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{M}\mathbf{x}(k) + \mathbf{N}\mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{C}\mathbf{x}(k), \end{aligned}$$

který řídíme pomocí lineární zpětné vazby ve tvaru
a) $\mathbf{u}(k) = -\mathbf{K}_1\mathbf{x}(k)$
b) $\mathbf{u}(k) = -\mathbf{K}_2\mathbf{y}(k)$.
Vypočtěte optimální zpětnovazební matici $\mathbf{K}_i$ v obou případech, která minimalizuje kritérium

$$J(\mathbf{K}) = \sum_{k=0}^{\infty} \mathbf{y}^T(k)\mathbf{Q}\mathbf{y}(k) + \mathbf{u}^T(k)\mathbf{R}\mathbf{u}(k).$$

Návod k řešení:
Dosaďte rovnici regulátoru do stavové rovnice systému. Řešení této rovnice dosaďte do kritéria. Použijte vztah $\mathbf{x}^T\mathbf{Q}\mathbf{x} = \mathrm{tr}(\mathbf{x}\mathbf{x}^T\mathbf{Q})$. Optimální zpětnou vazbu určíme z podmínky $\dfrac{\partial J}{\partial \mathbf{K}} = 0$. Derivaci kritéria podle zpětnovazební matice $\mathbf{K}$ provedeme tak, že stanovíme přírůstek kritéria $J(\mathbf{K} + \varepsilon\Delta\mathbf{K})$ a použijeme dvě následující tvrzení:

1) Mějme maticovou funkci $\mathbf{F}(\mathbf{X}) = (\mathbf{A} + \mathbf{B}\mathbf{X})^k$ a ε je nekonečně malá veličina prvního řádu. Pak platí

$$\begin{aligned} \mathbf{F}\left(\mathbf{X} + \varepsilon\Delta\mathbf{X}\right) &= \left(\mathbf{A} + \mathbf{B}\mathbf{X}\right)^k + \\ &\quad \varepsilon \sum_{i=0}^{k} \left(\mathbf{A} + \mathbf{B}\mathbf{X}\right)^{k-i-1} \mathbf{B}\Delta\mathbf{X} \left(\mathbf{A} + \mathbf{B}\mathbf{X}\right)^i + \Delta(\varepsilon^2) \end{aligned}$$

2) Mějme maticovou funkci $f(\mathbf{X}) = \mathrm{tr}\left(\mathbf{F}(\mathbf{X})\right)$ a ε je nekonečně malá veličina prvního řádu. Platí-li

$$f\left(\mathbf{X} + \varepsilon\Delta\mathbf{X}\right) = f\left(\mathbf{X}\right) + \varepsilon\ \mathrm{tr}\left(\mathbf{M}(\mathbf{X})\Delta\mathbf{X}\right) + \Delta(\varepsilon^2),$$

pak

$$\frac{\partial f}{\partial \mathbf{X}} = \mathbf{M}^T(\mathbf{X})$$

8. Proveďte řešení předchozího příkladu pro skalární verzi - uvažujte systém pouze prvního řádu.

9. Systém se stavovými rovnicemi

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \qquad \mathbf{x}(0) = \mathbf{a}$$

řídíme lineární zpětnou vazbou od stavu $\mathbf{u}(t) = -\mathbf{K}\mathbf{x}(t)$. Určete optimální zpětnovazební matici $\mathbf{K}$, která minimalizuje kritérium

$$J(\mathbf{K}) = \int_0^\infty \left(\mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^T(t)\mathbf{R}\mathbf{u}(t)\right)\ dt$$

kde $\mathbf{Q}$, $\mathbf{R}$ jsou dané pozitivně semidefinitní matice.

10. Mějme stochastický systém

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{v}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{w}(t) \end{aligned}$$

kde $\mathbf{v}(t)$, $\mathbf{w}(t)$ jsou nezávislé bílé stochastické procesy s nulovou střední hodnotou a kovariančními maticemi $\mathbf{Q}$, $\mathbf{R}$. Navrhněte optimálního pozorovatele stavu systému ve tvaru

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{A}\hat{\mathbf{x}}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{L}\left(\mathbf{y}(t) - \mathbf{C}\hat{\mathbf{x}}(t)\right)$$

Určete optimální hodnotu matice stavové injekce $\mathbf{L}$, aby ustálený rozptyl odchylky $\mathbf{e}(t) = \mathbf{x}(t) - \hat{\mathbf{x}}(t)$ byl minimální.

11. Nalezněte duální úlohu k úloze

$$\max\left\{\mathbf{c}^T\mathbf{x} \ :\ \mathbf{A}_1\mathbf{x} \le \mathbf{b}_1, \mathbf{A}_2\mathbf{x} = \mathbf{b}_2, \mathbf{A}_3\mathbf{x} \ge \mathbf{b}_3, \mathbf{x} \ge 0, \mathbf{b}_i \ge 0\right\}$$

12. Nalezněte duální úlohu k úloze kvadratického programování

$$\max\left\{\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{c}^T\mathbf{x} \ :\ \mathbf{A}\mathbf{x} \le \mathbf{b}, \mathbf{x} \ge 0\right\}$$

13. Ověřte dualitu následujících úloh

$$\max\left\{\mathbf{c}^T\mathbf{x} \ :\ \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \ge 0\right\} \qquad \text{duální:} \quad \min\left\{\mathbf{b}^T\mathbf{y} \ :\ \mathbf{A}^T\mathbf{y} \ge \mathbf{c}\right\}.$$

$$\max\left\{\mathbf{c}^T\mathbf{x} \ :\ \mathbf{A}\mathbf{x} = \mathbf{b}\right\} \qquad \text{duální:} \quad \min\left\{\mathbf{b}^T\mathbf{y} \ :\ \mathbf{A}^T\mathbf{y} = \mathbf{c}\right\}.$$

14. Mějme drát délky l, který rozdělíme na dvě části. Z jedné části vytvarujeme čtverec a z druhé kružnici. V jakém poměru musíme přeseknout drát, aby součet ploch ohraničených vytvořeným čtvercem a kružnicí byl minimální (maximální)?

15. Uvažujte konstrukci trajektu sloužícího k přepravě daného počtu tun nákladu po určité dráze. Výrobní náklady trajektu bez motoru lineárně rostou s jeho nosností, zatímco cena motoru je úměrná součinu jeho nosnosti a kvadrátu jeho rychlosti.

Ukažte, že celkové konstrukční náklady jsou nejmenší, když vydáme za samotný trajekt dvojnásobek peněz než za jeho motor. Zanedbejte čas nakládky a vykládky a předpokládejte, že trajekt pracuje nepřetržitě.

Kapitola 3

Minimalizace kvadratických forem

S problémem minimalizace kvadratických forem se setkáme při řešení mnoha problémů aproximace, estimace parametrů i optimálního řízení systémů. Je to problém, pro který byla vypracována řada spolehlivých a rychlých numerických algoritmů. Nejprve uvedeme dva motivační příklady, které ilustrují použitelnost tohoto přístupu.

Příklad 1. Aproximace charakteristiky termočlánku

Pro termočlánek Ch-A jsou v normě uvedeny hodnoty napětí termočlánku v závislosti na jeho teplotě. Hodnoty termonapětí jsou tabelovány po 10 ^0C od 100 do 1370 ^0C. Následuje výpis hodnot termonapětí pro uvedený rozsah teplot

U=4.095 4.508 4.919 5.327 5.733 6.137 6.539 6.939 7.338 7.737 08.137 08.537 08.938 09.341 09.745 10.151 10.560 10.969 11.381 11.793 12.207 12.623 13.039 13.456 13.874 14.292 14.712 15.132 15.552 15.974 16.395 16.818 17.241 17.664 18.088 18.513 18.938 19.363 19.788 20.214 20.640 21.066 21.493 21.919 22.346 22.772 23.198 23.624 24.050 24.476 24.902 25.327 25.751 26.176 26.599 27.022 27.445 27.867 28.288 28.709 29.128 29.547 29.965 30.383 30.799 31.214 31.629 32.042 32.455 32.866 33.277 33.686 34.095 34.502 34.909 35.314 35.718 36.121 36.524 36.925 37.325 37.724 38.122 38.519 38.915 39.310 39.703 40.096 40.488 40.879 41.269 41.657 42.045 42.432 42.817 43.202 43.585 43.968 44.349 44.729 45.108 45.486 45.863 46.238 46.612 46.985 47.356 47.726 48.095 48.462 48.828 49.192 49.555 49.916 50.267 50.633 50.990 51.344 51.697 52.049 52.398 52.747 53.093 53.439 53.782 54.125 54.466 54.807.

Problémem je interpolovat dané hodnoty v bodech mimo tabelované hodnoty. Můžeme to provést tak, že charakteristiku termočlánku aproximujeme polynomem. Aproximaci chceme provést tak, aby chyba aproximace byla v daných bodech minimální.

Řešení: Zvolíme-li aproximační polynom třetího stupně, pak je vztah mezi termonapětím U a teplotou t aproximován vztahem

$$U = a + bt + ct^2 + dt^3 \tag{3.1}$$

V bodech sítě tedy přibližně platí

$$U_i = \begin{bmatrix} 1 & t_i & t_i^2 & t_i^3 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}, \qquad i = 1, \ldots 128$$

kde číslo 128 je počet daných bodů sítě. Zavedením vektoru napětí $\mathbf{b}$, jehož složky jsou napětí U_i, vektoru hledaných parametrů $\mathbf{x} = \begin{bmatrix} a & b & c & d \end{bmatrix}^T$ a matice dat $\mathbf{A}$ jejíž řádky jsou tvořeny vektory teplot $\begin{bmatrix} 1 & t_i & t_i^2 & t_i^3 \end{bmatrix}$, můžeme předchozí soustavu zapsat maticově ve tvaru

$$\mathbf{b} = \mathbf{A}\mathbf{x} + \mathbf{e} \tag{3.2}$$

kde $\mathbf{e}$ je vektor chyb, který chceme minimalizovat, aby aproximace byla co nejpřesnější. Jako kriterium aproximace volíme kvadratickou normu odchylky

$$J = \|\mathbf{e}\|_2^2 = \mathbf{e}^T\mathbf{e} \tag{3.3}$$

Po dosazení do kritéria za vektor odchylky $\mathbf{e} = \mathbf{b} - \mathbf{A}\mathbf{x}$ dostaneme

$$J(\mathbf{x}) = (\mathbf{A}\mathbf{x} - \mathbf{b})^T (\mathbf{A}\mathbf{x} - \mathbf{b}) = \mathbf{x}^T\mathbf{A}^T\mathbf{A}\mathbf{x} - \mathbf{x}^T\mathbf{A}^T\mathbf{b} - \mathbf{b}^T\mathbf{A}\mathbf{x} + \mathbf{b}^T\mathbf{b}$$

Hledáme tedy takový vektor $\mathbf{x}^*$, který minimalizuje předchozí kritérium aproximace

$$J(\mathbf{x}^*) = \min_{\mathbf{x}} J(\mathbf{x}), \qquad \mathbf{x}^* = \arg\min_{\mathbf{x}} J(\mathbf{x})$$

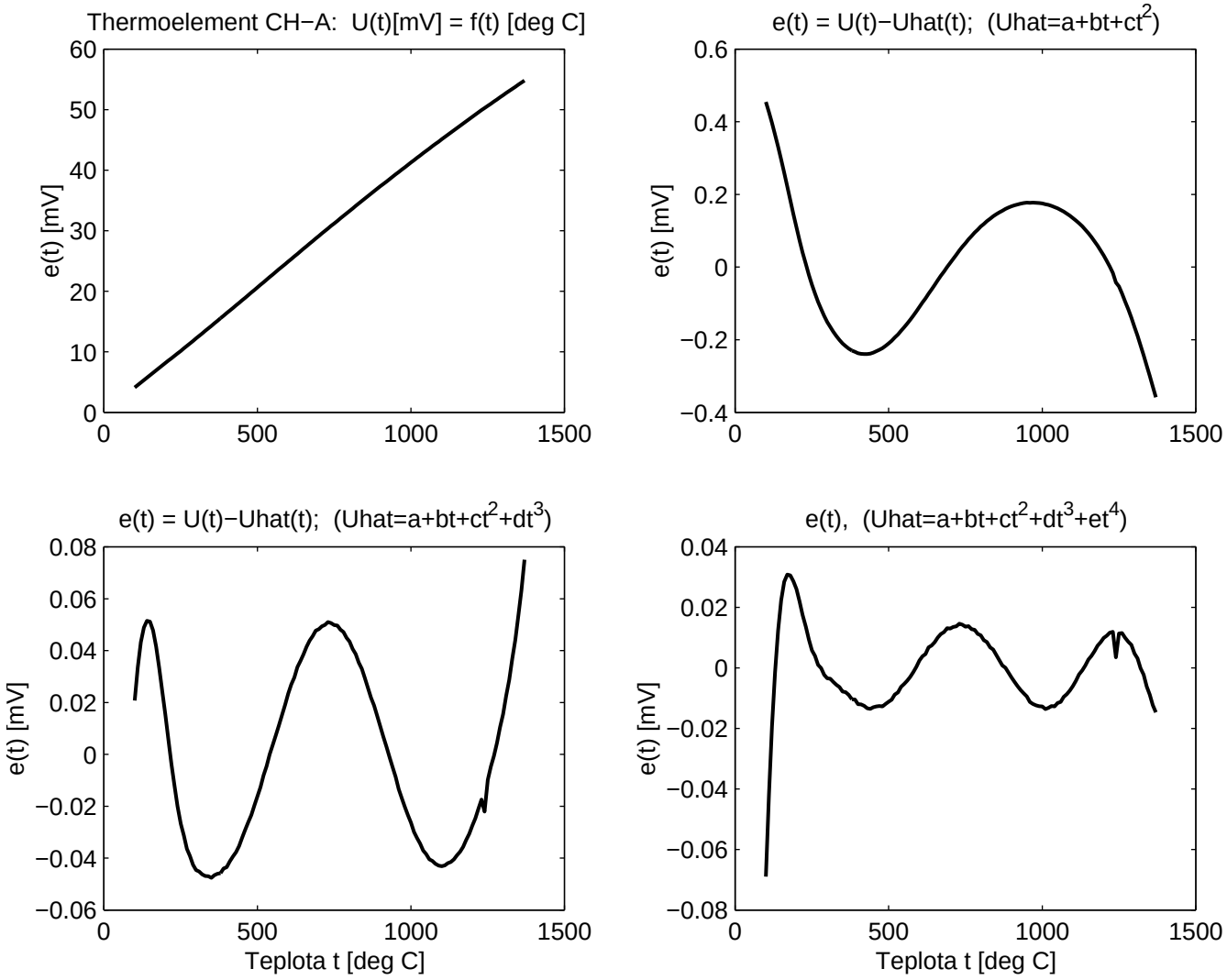
Původně se jednalo o problém řešení přeurčené soustavy algebraických rovnic $\mathbf{A}\mathbf{x} = \mathbf{b}$. V jazyce MATLAB se tento problém jednoduše řeší příkazem $\mathbf{x}^* = \mathbf{A}\backslash\mathbf{b}$. Řešení přeurčené soustavy algebraických rovnic jsme zde převedli na problém minimalizace kvadratické formy.

Minimalizací předchozí kvadratické formy určíme koeficienty polynomu třetího stupně $\mathbf{x}^* = \begin{bmatrix} 0.13577 & 3.87510^{-2} & 6.74310^{-6} & -4.33710^{-9} \end{bmatrix}$. Při tom maximální chyba aproximace $\max|e_i| = 0.07$.

Zvolíme-li aproximační polynom pouze druhého stupně, pak vektor parametrů je roven $\mathbf{x}^* = \begin{bmatrix} -0.803061 & 4.471710^{-2} & -0.2810^{-6} \end{bmatrix}$ a maximální chyba aproximace vzroste na hodnotu $\max|e_i| = 0.45$. Na obr. 3 je charakteristika termočlánku Ch-A a průběh chyb při aproximaci polynomy druhého, třetího a čtvrtého stupně. Uvědomte si, že zvyšování stupně aproximačního polynomu nemusí přinést zmenšení chyb aproximace. Naopak při vysokém stupni aproximačního polynomu jsou průběhy aproximační křivky mezi danými body nevhodné. Ověřte! Problémem je, jaký je tedy nejvhodnější stupeň aproximačního polynomu.

Příklad 2. Identifikace parametrů diskrétního systému z naměřených dat.
Předpokládejme vnější popis diskrétního systému ve tvaru

$$y(t) = -a_1 y(t-1) - \ldots - a_n y(t-n) + b_0 u(t) + \ldots + b_m u(t-m) + e(t)$$



Obrázek 3.1: Charakteristika termočlánku a chyby při aproximaci polynomy druhého až čtvrtého stupně

kde $y(t)$ je výstup systému v čase t, $u(t)$ je vstup systému a $e(t)$ náhodná veličina reprezentující nepřesnosti měření i aproximace.

Předpokládejme, že máme k dispozici množinu měření výstupní i vstupní veličiny v po sobě jdoucích diskrétních časových okamžicích $t = \ldots, 0, 1, \ldots$. Naším úkolem je pomocí množiny měření vstupů a výstupů systému určit co nejvěrněji parametry systému $a_i,\ b_i$. Parametry systému určíme z podmínky minimalizace chyby diferenční rovnice systému.

Zavedeme si vektor hodnot výstupu $\mathbf{b} = \left[\ y(1),\ \ y(2)\ \ \ldots\ \ y(n)\ \right]^T$, vektor neznámých parametrů $\mathbf{x} = \left[\ a_1\ \ a_2\ \ \ldots\ \ a_n\ \ b_0\ \ \ldots\ \ b_m\ \right]^T$ a matici dat $\mathbf{A}$ podle následujícího

předpisu

$$\mathbf{A} = \begin{bmatrix} -y(0) & -y(-1) & \dots & -y(1-n), & u(1) & \dots & u(1-n) \\ -y(1) & -y(0) & \dots & -y(2-n), & u(2) & \dots & u(2-n) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ -y(\nu-1) & -y(\nu-2) & \dots & -y(\nu-n), & u(\nu) & \dots & u(\nu-n) \end{bmatrix}$$

Pro diskrétní čas $t = i$ diferenční rovnici systému můžeme zapsat ve tvaru $y_i = \mathbf{A}_i\mathbf{x}+e_i$, kde y_i je i-tý prvek vektoru výstupů $\mathbf{b}$, $\mathbf{A}_i$ je i-tý řádek matice dat $\mathbf{A}$ a $e_i = e(i)$. Pro $t = 1, \dots, \nu$ s použitím zavedených vektorů dostaneme soustavu algebraických rovnic ve tvaru

$$\mathbf{b} = \mathbf{A}\mathbf{x} + \mathbf{e}$$

Minimalizace normy vektoru chyb rovnic vede opět na problém minimalizace kvadratické formy. □

3.1 Minimalizace - analytické vztahy

Mějme tedy kvadratickou formu

$$J(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{x}^T\mathbf{B}\mathbf{y} + \mathbf{y}^T\mathbf{D}\mathbf{x} + \mathbf{y}^T\mathbf{C}\mathbf{y} \tag{3.4}$$

kde $\mathbf{A}$ je pozitivně semidefinitní matice. Hledáme minimum této kvadratické formy vzhledem k vektorové proměnné $\mathbf{x}$. Hledáme tedy

$$\mathbf{x}^* = \arg\min_{\mathbf{x}} J(\mathbf{x}, \mathbf{y})$$

Ukážeme dva postupy vedoucí k analytickému výrazu pro určení optimálního $\mathbf{x}^*$ a minima kvadratické formy. V prvním postupu použijeme vztahy pro derivaci kvadratických forem. Snadno lze prokázat, že platí (pokud derivace skalární funkce podle vektoru je řádkový vektor)

$$\frac{\partial \mathbf{x}^T\mathbf{A}\mathbf{x}}{\partial \mathbf{x}} = \mathbf{x}^T\left(\mathbf{A} + \mathbf{A}^T\right) \tag{3.5}$$

$$\frac{\partial \mathbf{x}^T\mathbf{B}\mathbf{y}}{\partial \mathbf{x}} = \mathbf{y}^T\mathbf{B}^T, \qquad \frac{\partial \mathbf{y}^T\mathbf{D}\mathbf{x}}{\partial \mathbf{x}} = \mathbf{y}^T\mathbf{D} \tag{3.6}$$

Minimum kvadratické formy (3.4) nalezneme z podmínky nulového gradientu kritéria vzhledem k $\mathbf{x}$, čili

$$\frac{\partial J(\mathbf{x}, \mathbf{y})}{\partial \mathbf{x}} = \mathbf{0}, \quad \text{pro} \quad \mathbf{x} = \mathbf{x}^*.$$

Použitím vztahů pro derivaci kvadratických forem dostaneme z předchozího vztahu

$$\left(\mathbf{A} + \mathbf{A}^T\right)\mathbf{x} + \mathbf{B}\mathbf{y} + \mathbf{D}^T\mathbf{y} = \mathbf{0}, \qquad \text{pro} \quad \mathbf{x} = \mathbf{x}^*$$

Odtud plyne optimální $\mathbf{x}^*$

$$\mathbf{x}^* = -\left(\mathbf{A} + \mathbf{A}^T\right)^{-1}\left(\mathbf{B} + \mathbf{D}^T\right)\mathbf{y} \tag{3.7}$$

Předchozí vztah platí pro pozitivně definitní matici $\mathbf{A}$. Pokud je matice $\mathbf{A}$ pouze pozitivně semidefinitní, pak místo inverze matice je třeba použít pseudoinverzi (viz dále).

Hessova matice druhých derivací minimalizované kvadratické formy je rovna matici $(\mathbf{A} + \mathbf{A}^T)$ a proto při pozitivně semidefinitní matici $\mathbf{A}$ je existence minima zajištěna.

Pozn: Úplně podobným postupem můžeme hledat minimum kvadratické formy vzhledem k vektoru $\mathbf{y}$. □

Jednodušší metoda minimalizace kvadratických forem spočívá v jejich úpravě na úplný čtverec. Kvadratickou formu (3.4) upravíme do tvaru

$$J(\mathbf{x}, \mathbf{y}) = (\mathbf{x} - \mathbf{x}^*)^T \mathbf{A} (\mathbf{x} - \mathbf{x}^*) + \mathbf{y}^T \mathbf{C} \mathbf{y} - (\mathbf{x}^*)^T \mathbf{A} \mathbf{x}^* \tag{3.8}$$

kde $\mathbf{x}^*$ je zatím nějaký vektor. Aby předchozí kvadratická forma byla rovna kvadratické formě (3.4), musí zřejmě platit

$$-(\mathbf{x}^*)^T \mathbf{A} \mathbf{x} = \mathbf{y}^T \mathbf{D} \mathbf{x}, \qquad -\mathbf{x}^T \mathbf{A} \mathbf{x}^* = \mathbf{x}^T \mathbf{B} \mathbf{y}$$

Protože předchozí vztahy platí pro všechna $\mathbf{x}$, platí zřejmě

$$-(\mathbf{x}^*)^T \mathbf{A} = \mathbf{y}^T \mathbf{D}, \qquad -\mathbf{A} \mathbf{x}^* = \mathbf{B} \mathbf{y}$$

Po transpozici první rovnice a následném sečtení obou rovnic dostaneme

$$-\left(\mathbf{A} + \mathbf{A}^T\right) \mathbf{x}^* = \left(\mathbf{B} + \mathbf{D}^T\right) \mathbf{y}$$

Neznámý vektor $\mathbf{x}^*$ je tedy roven

$$\mathbf{x}^* = -\left(\mathbf{A} + \mathbf{A}^T\right)^{-1} \left(\mathbf{B} + \mathbf{D}^T\right) \mathbf{y}.$$

Kvadratická forma (3.8) je minimální vzhledem k proměnné $\mathbf{x}$ tehdy, když $\mathbf{x} = \mathbf{x}^*$, neboť $\mathbf{x}^*$ není na $\mathbf{x}$ závislé. Proto $\mathbf{x}^*$ je optimální hodnota $\mathbf{x}$, minimalizující kvadratickou formu.

Výsledek je stejný jako při použití derivací - viz (3.7). Při minimalizaci kvadratických forem jejich úpravou na úplný čtverec získáme jejich minimum jednoduše bez použití derivací. Minimální hodnotu kvadratické formy dostaneme dosazením $\mathbf{x}^*$ do (3.8).

Z předchozích vztahů je zřejmé, že můžeme předpokládat, že matice $\mathbf{A}$ je symetrická matice. Pokud tomu tak není, můžeme ji symetrizovat zavedením nové symetrické matice $\mathbf{A}_s = \frac{1}{2}(\mathbf{A} + \mathbf{A}^T)$ a výsledek optimalizace se zřejmě nezmění. Podobně lze symetrizovat i matici $\mathbf{C}$. Také můžeme předpokládat, že matice $\mathbf{B}$ a $\mathbf{D}$ jsou si až na transpozici rovny. Pokud tomu tak není, pak zavedeme novou matici $\mathbf{B}_1$ podle vztahu $\mathbf{B} + \mathbf{D}^T = 2\mathbf{B}_1$ a touto novou maticí $\mathbf{B}_1$ nahradíme matice $\mathbf{B}$ a $\mathbf{D}$ v kvadratické formě podle vztahu $\mathbf{B} = \mathbf{B}_1$, $\mathbf{D} = \mathbf{B}_1^T$. Hodnota kvadratické formy se zřejmě nezmění. Proto budeme dále předpokládat, že matice $\mathbf{A}$ i matice $\mathbf{C}$ jsou symetrické matice a matice $\mathbf{B}$ a $\mathbf{D}$ jsou si až na transpozici rovny.

Kvadratické formy můžeme zapisovat také ve tvaru

$$J(\mathbf{x}, \mathbf{y}) = \begin{bmatrix} \mathbf{x}^T & \mathbf{y}^T \end{bmatrix} \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}. \tag{3.9}$$

Přitom předpokládáme, že matice $\mathbf{A}$ i matice $\mathbf{C}$ jsou symetrické matice a matice $\mathbf{A}$ je pozitivně semidefinitní. Dále předpokládáme, že složená matice $\mathbf{M}$

$$\mathbf{M} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C} \end{bmatrix}$$

je symetrická pozitivně semidefinitní matice (kvadratická forma $J(\mathbf{y}, \mathbf{x})$ je nezáporná). Minimalizací kvadratické formy (3.9) vzhledem k $\mathbf{x}$ získáme pro optimální $\mathbf{x}^*$ vztah

$$\mathbf{A}\mathbf{x}^* = -\mathbf{B}\mathbf{y}$$

Obecné řešení předchozího vztahu je

$$\mathbf{x}^* = -\mathbf{A}^+\mathbf{B}\mathbf{y} + \mathbf{x}^0 \tag{3.10}$$

kde $\mathbf{A}^+$ je pseudoinverze matice $\mathbf{A}$, která vyhovuje vztahu $\mathbf{A}\mathbf{A}^+\mathbf{A} - \mathbf{A} = \mathbf{0}$ a vektor $\mathbf{x}^0$ je libovolný vektor z jádra zobrazení $\mathbf{A}$, to je takový vektor, který splňuje rovnici $\mathbf{A}\mathbf{x}^0 = \mathbf{0}$.

Protože pro pseudoinverzi platí $\mathbf{A}(\mathbf{A}^+\mathbf{A} - \mathbf{I}) = \mathbf{0}$ je vektor $\mathbf{x}^0$ roven

$$\mathbf{x}^0 = \left(\mathbf{A}^+\mathbf{A} - \mathbf{I}\right)\mathbf{z}$$

kde $\mathbf{z}$ je libovolný vektor stejné dimenze jako je dimenze vektoru $\mathbf{x}$.

Minimální hodnota kvadratické formy je rovna

$$J^*(\mathbf{y}) = J(\mathbf{x}^*, \mathbf{y}) = \mathbf{y}^T\left(\mathbf{C} - \mathbf{B}^T\mathbf{A}^+\mathbf{B}\right)\mathbf{y} \tag{3.11}$$

Optimální $\mathbf{x}^*$ není jediné, ale minimum kvadratické formy je samozřejmě jediné.

Předchozí vztahy nejsou vhodné pro numerický výpočet. Dále uvedeme dvě metody vhodné pro numerický výpočet optimálního vektoru $\mathbf{x}^*$ a to zobecněnou Choleskyho faktorizaci a LDU (Biermanovu) faktorizaci.

3.2 Zobecněná Choleskyho faktorizace

Standardní Choleskyho faktorizace je rozklad (faktorizace) libovolné **pozitivně definitní matice** $\mathbf{M}$ do tvaru

$$\mathbf{M} = \mathbf{F}^T\mathbf{F}, \tag{3.12}$$

kde $\mathbf{F}$ je reálná horní trojúhelníková matice. To znamená, že $F_{i,j} = 0$ pro $i > j$.

Faktorizace pozitivně definitní matice $\mathbf{M}$ je jediná až na znaménko každého řádku matice $\mathbf{F}$. Faktorizace je jediná omezíme-li se podmínkou, že diagonální prvky matice $\mathbf{F}$ jsou kladné. To je známý výsledek standardní Choleskyho faktorizace.

Jak je to v případě, je-li matice $\mathbf{M}$ pouze **pozitivně semidefinitní**. V tomto případě není Choleskyho faktorizace jednoznačná. Jedná se o **zobecněnou Choleskyho faktorizaci** a tento případ nyní rozebereme.

Pro $i \leq j$ platí pro jednotlivé prvky matice $\mathbf{M}$ vztah

$$M_{i,j} = \sum_{k=1}^{i} F_{k,i}F_{k,j} = \sum_{k=1}^{i-1} F_{k,i}F_{k,j} + F_{i,i}F_{i,j}$$

Z předchozího vztahu pro prvky matice $\mathbf{M}$ plyne algoritmus výpočtu prvků Choleskyho rozkladu $\mathbf{F}$. Pro $i = 1,\ 2, \ldots$ postupně počítáme nejprve diagonální prvky matice $\mathbf{F}$ podle vztahu

$$F_{i,i} = \sqrt{M_{i,i} - \sum_{k=1}^{i-1} F_{k,i}^2}$$

a potom jednotlivé prvky matice $\mathbf{F}$ vpravo od diagonály podle vztahu

$$F_{i,j} = \frac{1}{F_{i,i}} \left[M_{i,j} - \sum_{k=1}^{i-1} F_{k,i} F_{k,j} \right], \qquad j > i.$$

Je zřejmé, že při výpočtu faktorů se vyskytují dva problémy. Při výpočtu diagonálních prvků matice $\mathbf{F}$ se vyskytuje odmocnina, což zdržuje výpočet. Proto byla vyvinuta jiná faktorizace (popsaná později), která nevyžaduje výpočet odmocniny.

Závažnější problém nastane, když diagonální prvek $F_{i,i}$ je roven nule, neboť se jím při výpočtu nediagonálních prvků matice $\mathbf{F}$ dělí. Choleskyho faktorizace není v tomto případě jednoznačná a vždy jednoznačný rozklad dostaneme tím, že je-li některý diagonální prvek matice $\mathbf{F}$ nulový, pak položíme rovný nule celý odpovídající řádek matice $\mathbf{F}$. To znamená, že vztah pro mimodiagonální prvky $F_{i,j}$ doplníme podmínkou

$$\text{Jestliže} \quad F_{i,i} = 0, \quad \text{pak} \quad F_{i,j} = 0, \quad \forall j > i.$$

Po rozkladu matice $\mathbf{M}$ provedeme již snadno minimalizaci kvadratické formy i výpočet optimálního $\mathbf{x}$. Provedeme následující serii úprav kvadratické formy

$$J(\mathbf{x}, \mathbf{y}) = \left[\mathbf{x}^T \mathbf{y}^T\right] \mathbf{M} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \left[\mathbf{x}^T \mathbf{y}^T\right] \mathbf{F}^T \mathbf{F} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \left\| \mathbf{F} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \right\|^2$$

Horní trojúhelníkovou matici $\mathbf{F}$ rozložíme na submatice

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_x & \mathbf{F}_{x,y} \\ \mathbf{0} & \mathbf{F}_y \end{bmatrix}$$

kde dimenze rozkladu odpovídají dimenzi vektorů $\mathbf{x}$ a $\mathbf{y}$. Při tom zřejmě submatice $\mathbf{F}_x$ i $\mathbf{F}_y$ jsou horní trojúhelníkové matice. Kvadratická forma je potom rovna

$$J(\mathbf{x}, \mathbf{y}) = \left\| \begin{bmatrix} \mathbf{F}_x & \mathbf{F}_{x,y} \\ \mathbf{0} & \mathbf{F}_y \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \right\|^2 = \| \mathbf{F}_x \mathbf{x} + \mathbf{F}_{x,y} \mathbf{y} \|^2 + \| \mathbf{F}_y \mathbf{y} \|^2$$

Minimum kvadratické formy vzhledem k proměnné $\mathbf{x}$ nastane zřejmě, když první člen na pravé straně předchozího výrazu bude roven nule. Proto optimální $\mathbf{x}^*$ získáme z rovnice

$$\mathbf{F}_x \mathbf{x}^* + \mathbf{F}_{x,y} \mathbf{y} = \mathbf{0} \tag{3.13}$$

Protože matice $\mathbf{F}_x$ je horní trojúhelníková matice, provedeme výpočet optimálního $\mathbf{x}^*$ po jednotlivých prvcích tohoto vektoru odspodu. Je-li pro některé i diagonální prvek matice $\mathbf{F}_x$ nulový, pak je podle předchozího nulový celý i-tý řádek této matice. Proto odpovídající složka x_i^* optimálního vektoru může být zvolena libovolně. V závislosti na libovolně zvoleném x_i^* vyjdou potom další prvky x_k^* pro $k < i$. Všimněme si, že pro

výpočet optimálního $\mathbf{x}^*$ není třeba provádět celou faktorizaci matice $\mathbf{F}$, ale stačí provést faktorizaci pouze horní části matice $\mathbf{M}$, která je potřebná pro získání submatic $\mathbf{F}_x$ a $\mathbf{F}_{x,y}$.

Minimální hodnota kvadratické formy je

$$J^*(\mathbf{y}) = J(\mathbf{x}^*, \mathbf{y}) = ||\mathbf{F}_y\mathbf{y}||^2 = \mathbf{y}^T\mathbf{F}_y^T\mathbf{F}_y\mathbf{y}$$

Provedeme úpravu kvadratické formy. Matici $\mathbf{M}$ vyjádříme pomocí submatic Choleskyho faktorů, pak

$$\begin{aligned}\mathbf{M} &= \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C} \end{bmatrix} = \mathbf{F}^T\mathbf{F} = \\ &= \begin{bmatrix} \mathbf{F}_x & \mathbf{F}_{x,y} \\ \mathbf{0} & \mathbf{F}_y \end{bmatrix}^T \begin{bmatrix} \mathbf{F}_x & \mathbf{F}_{x,y} \\ \mathbf{0} & \mathbf{F}_y \end{bmatrix} = \begin{bmatrix} \mathbf{F}_x^T\mathbf{F}_x & \mathbf{F}_x^T\mathbf{F}_{x,y} \\ \mathbf{F}_{x,y}^T\mathbf{F}_x & \mathbf{F}_y^T\mathbf{F}_y + \mathbf{F}_{x,y}^T\mathbf{F}_{x,y} \end{bmatrix}\end{aligned}$$

Odtud plyne, že matice $\mathbf{C}$ je rovna

$$\mathbf{C} = \mathbf{F}_y^T\mathbf{F}_y + \mathbf{F}_{x,y}^T\mathbf{F}_{x,y}$$

Proto optimální hodnotu kvadratické formy můžeme vyjádřit ve tvaru

$$J^*(\mathbf{y}) = \mathbf{y}^T\mathbf{F}_y^T\mathbf{F}_y\mathbf{y} = \mathbf{y}^T\left(\mathbf{C} - \mathbf{F}_{x,y}^T\mathbf{F}_{x,y}\right)\mathbf{y}$$

Proto i při výpočtu minima kvadratické formy opět není třeba provádět úplnou faktorizaci matice $\mathbf{F}$, ale je možno počítat pouze její horní část potřebnou pro získání submatic $\mathbf{F}_u$ a $\mathbf{F}_{u,x}$.

3.3 LDU faktorizace

Matici $\mathbf{M}$ budeme nyní faktorizovat jiným způsobem podle následujícího vztahu

$$\mathbf{M} = \mathbf{U}^T\mathbf{D}\mathbf{U} \tag{3.14}$$

kde $\mathbf{U}$ je monická horní trojúhelníková matice (monická znamená, že má jednotkovou diagonálu). Matice $\mathbf{D} = \text{diag}\,(\mathbf{d})$ je diagonální matice (v její diagonále je vektor $\mathbf{d}$). Někdy se tento rozklad píše ve tvaru $\mathbf{M} = \mathbf{L}\mathbf{D}\mathbf{L}^T$, kde $\mathbf{L}$ je monická dolní trojúhelníková matice. Zřejmě jsou oba zápisy identické.

Předchozí rozklad se často schematicky vyjadřuje ve tvaru

$$\mathbf{M} = |\mathbf{d}; \mathbf{U}|$$

nebo $\mathbf{M} = \left|\mathbf{d}; \mathbf{L}^T\right|$ při rozkladu ve tvaru $\mathbf{M} = \mathbf{L}\mathbf{D}\mathbf{L}^T$. Pro $i \leq j$ platí pro prvky matice $\mathbf{M}$

$$M_{i,j} = \sum_{k=1}^{i} U_{k,i}d_kU_{k,j} = \sum_{k=1}^{i-1} U_{k,i}d_kU_{k,j} + U_{i,i}d_iU_{i,j}$$

Z předchozího vztahu můžeme počítat jednotlivé prvky rozkladu. Pro $i = 1, 2, \ldots$ postupně počítáme nejprve prvky diagonální matice. Pro $j = i$ platí

$$d_i = M_{i,i} - \sum_{k=1}^{i-1} d_kU_{k,i}^2$$

Je-li $d_i = 0$, pak je nulový celý odpovídající řádek matice $\mathbf{U}$, čili

$$\text{if} \quad d_i = 0 \quad \text{then} \quad U_{i,j} = 0 \quad \forall j > i$$

Pro nenulové prvky d_i počítáme prvky v odpovídajícím řádku matice $\mathbf{U}$ podle vztahu

$$U_{i,j} = \frac{1}{d_i}\left(M_{i,j} - \sum_{k=1}^{i-1} d_k U_{k,i} U_{k,j}\right)$$

Po nalezení rozkladu snadno provedeme minimalizaci kvadratické formy. Použitím faktorů je kvadratická forma rovna

$$J(\mathbf{x},\mathbf{y}) = \begin{bmatrix} \mathbf{x}^T & \mathbf{y}^T \end{bmatrix} \mathbf{U}^T\mathbf{D}\mathbf{U} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}$$

Provedeme rozklad monické horní trojúhelníkové matice $\mathbf{U}$ i vektoru $\mathbf{d}$ tvořícího digonálu diagonální matice $\mathbf{D}$

$$\mathbf{U} = \begin{bmatrix} \mathbf{U}_x & \mathbf{U}_{x,y} \\ \mathbf{0} & \mathbf{U}_y \end{bmatrix}, \qquad \mathbf{d} = \begin{bmatrix} \mathbf{d}_x \\ \mathbf{d}_y \end{bmatrix}$$

Dimenze submatic jsou voleny shodně s dimenzemi vektorů $\mathbf{x}$ a $\mathbf{y}$. Potom kvadratická forma je rovna

$$\begin{aligned} J(\mathbf{x},\mathbf{y}) &= \begin{bmatrix} \mathbf{x}^T & \mathbf{y}^T \end{bmatrix} \begin{bmatrix} \mathbf{U}_x & \mathbf{U}_{x,y} \\ \mathbf{0} & \mathbf{U}_y \end{bmatrix}^T \text{diag}\left(\begin{matrix} \mathbf{d}_x \\ \mathbf{d}_y \end{matrix}\right) \begin{bmatrix} \mathbf{U}_x & \mathbf{U}_{x,y} \\ \mathbf{0} & \mathbf{U}_y \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \\ &= [\mathbf{U}_x\mathbf{x} + \mathbf{U}_{x,y}\mathbf{y}]^T \,\text{diag}\,(\mathbf{d}_x)\,[\mathbf{U}_x\mathbf{x} + \mathbf{U}_{x,y}\mathbf{y}] + \mathbf{y}^T\mathbf{U}_y^T\text{diag}\,(\mathbf{d}_y)\mathbf{U}_y\mathbf{y} \end{aligned}$$

Minimum této kvadratické formy vzhledem k vektoru $\mathbf{x}$ je určeno rovnici

$$\mathbf{U}_x\mathbf{x}^* + \mathbf{U}_{x,y}\mathbf{y} = \mathbf{0} \tag{3.15}$$

Optimální vektor $\mathbf{x}^*$ počítáme po složkách odzdola. Protože matice $\mathbf{U}_x$ je monická horní trojúhelníková matice, provádíme výpočet optimálního $\mathbf{x}^*$ bez dělení.

Jestliže i-tý diagonální prvek vektoru $\mathbf{d}_x$ je nulový ($(d_x)_i = 0$), pak příslušná složka x_i^* může být volena libovolně. Volba x_i^* potom ale ovlivňuje další složky x_j^* pro $j < i$.

Minimum kvadratické formy je jediné a je rovno

$$J(\mathbf{x}^*,\mathbf{y}) = \mathbf{y}^T\mathbf{U}_y^T\text{diag}\,(\mathbf{d}_y)\mathbf{U}_y\mathbf{y}$$

Po úpravě můžeme toto minimum vyjádřit ve tvaru

$$J(\mathbf{x}^*,\mathbf{y}) = \mathbf{y}^T\left(\mathbf{C} - \mathbf{U}_{x,y}^T\text{diag}\,(\mathbf{d}_x)\mathbf{U}_{x,y}\right)\mathbf{y}$$

ve kterém využíváme pouze horní faktory rozkladu, které byly třeba i pro určení optimálního $\mathbf{x}^*$.

3.4 Aktualizace Choleskyho faktoru

Z předchozího výkladu víme, že minimalizace kvadratických forem je výsledkem řešení soustavy lineárních algebraických rovnic, které minimalizuje kvadratickou normu vektoru chyby soustavy.

Máme tedy lineární rovnici ve tvaru $\mathbf{Ax} - \mathbf{b} = \mathbf{0}$. Vektor chyby této rovnice zapíšeme ve tvaru $\mathbf{e} = [\mathbf{A} - \mathbf{b}] \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix}$

Minimalizace kvadrátu normy vektoru $\mathbf{e}$ vede na minimalizaci kvadratické formy

$$J(\mathbf{x}) = \mathbf{e}^T\mathbf{e} = \begin{bmatrix} \mathbf{x}^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{A} & -\mathbf{b} \end{bmatrix}^T \begin{bmatrix} \mathbf{A} & -\mathbf{b} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix}$$

Provedeme faktorizaci matice dat $\mathbf{H} = \begin{bmatrix} \mathbf{A} & -\mathbf{b} \end{bmatrix}$. Potom platí $\mathbf{H}^T\mathbf{H} = \mathbf{F}^T\mathbf{F}$, kde Choleskyho faktor $\mathbf{F}$ je čtvercová horní trojúhelníková matice. Již popsaným postupem vypočteme optimální vektor $\mathbf{x}^*$ a optimální (minimální) hodnotu kvadratické formy $J^*(\mathbf{x})$.

Představme si situaci, že po provedené faktorizaci matice dat a výpočtu optimálního vektoru $\mathbf{x}^*$ dostaneme nová měření vedoucí na další lineární rovnici $\mathbf{ax} - \beta = 0$, kde řádkový vektor $\mathbf{a}$ a konstanta β obsahují nová data. Problémem je nyní jak efektivně vypočíst nový optimální vektor $\mathbf{x}^*$, aniž bychom museli provádět Choleskyho faktorizaci celou úplně znova.

Přidáním nově získané lineární rovnice k předchozí soustavě dostaneme nový tvar kvadratické formy

$$J(\mathbf{x}) = \begin{bmatrix} \mathbf{x}^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{A} & -\mathbf{b} \\ \mathbf{a} & -\beta \end{bmatrix}^T \begin{bmatrix} \mathbf{A} & -\mathbf{b} \\ \mathbf{a} & -\beta \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix}$$

Pomocí matice dat $\mathbf{H}$ a řádkového vektoru nových dat $\mathbf{h} = \begin{bmatrix} \mathbf{a} & -\beta \end{bmatrix}$ vyjádříme aktualizovanou kvadratickou formu ve tvaru

$$J(\mathbf{x}) = \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix}^T \begin{bmatrix} \mathbf{H} \\ \mathbf{h} \end{bmatrix}^T \begin{bmatrix} \mathbf{H} \\ \mathbf{h} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{x}^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{H}^T\mathbf{H} + \mathbf{h}^T\mathbf{h} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix}$$

Protože platí $\mathbf{H}^T\mathbf{H} = \mathbf{F}^T\mathbf{F}$ pak nová matice kvadratické formy je rovna

$$\mathbf{M} = \mathbf{H}^T\mathbf{H} + \mathbf{h}^T\mathbf{h} = \mathbf{F}^T\mathbf{F} + \mathbf{h}^T\mathbf{h}$$

Aktualizace Choleskyho faktoru spočívá ve výpočtu nového Choleskyho faktoru $\bar{\mathbf{F}}$ podle vztahu

$$\mathbf{M} = \mathbf{F}^T\mathbf{F} + \mathbf{h}^T\mathbf{h} = \bar{\mathbf{F}}^T\bar{\mathbf{F}}$$

kde $\bar{\mathbf{F}}$ je opět horní trojúhelníková matice.

Navíc velmi často chceme stará data nějakým způsobem "znevážit", nebo "zapomenout" a proto matici "starých " dat $\mathbf{H}$ respektive její Choleskyho faktor $\mathbf{F}$ násobíme číslem $0 < \varphi < 1$, které nazýváme **faktor zapomínání**. Aktualizace Choleskyho faktoru potom vede na výpočet čtvercové horní trojúhelníkové matice $\bar{\mathbf{F}}$ pro kterou platí

$$\bar{\mathbf{F}}^T\bar{\mathbf{F}} = \begin{bmatrix} \varphi\mathbf{F} \\ \mathbf{h} \end{bmatrix}^T \begin{bmatrix} \varphi\mathbf{F} \\ \mathbf{h} \end{bmatrix} = \varphi^2\mathbf{F}^T\mathbf{F} + \mathbf{h}^T\mathbf{h}.$$

Triangulizace matice (to je úprava matice na horní trojúhelníkový tvar) se provádí pomocí specielní **ortogonální matice T**, zvané **matice planárních rotací**, nebo také **Givensova rotace**. Ortogonální matice je taková matice, pro kterou platí $\mathbf{T}^T\mathbf{T} = \mathbf{I}$. Matici $\mathbf{T}$ volíme takovou, aby platilo

$$\mathbf{T}\begin{bmatrix} \varphi\mathbf{F} \\ \mathbf{h} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{F}} \\ \mathbf{0} \end{bmatrix}$$

Potom platí

$$\mathbf{M} = \begin{bmatrix} \varphi\mathbf{F} \\ \mathbf{h} \end{bmatrix}^T \begin{bmatrix} \varphi\mathbf{F} \\ \mathbf{h} \end{bmatrix} = \begin{bmatrix} \varphi\mathbf{F} \\ \mathbf{h} \end{bmatrix}^T \mathbf{T}^T\mathbf{T} \begin{bmatrix} \varphi\mathbf{F} \\ \mathbf{h} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{F}} \\ \mathbf{0} \end{bmatrix}^T \begin{bmatrix} \bar{\mathbf{F}} \\ \mathbf{0} \end{bmatrix} = \bar{\mathbf{F}}^T\bar{\mathbf{F}}$$

Přitom předpokládáme, že matice $\mathbf{F}$ je horní trojúhelníková matice. Triangulizace spočívá v podstatě ve vynulování posledního řádku matice $\begin{bmatrix} \mathbf{F} \\ \mathbf{h} \end{bmatrix}$ (faktor zapomínání φ nebudeme pro jednoduchost uvažovat).

Triangulizaci provádíme sekvenčně tak, že nejprve vynulujeme první prvek posledního řádku vhodně zvolenou ortogonální maticí $\mathbf{T}$, potom vynulujeme druhý prvek posledního řádku opět vhodně zvolenou ortogonální maticí $\mathbf{T}$ a tak postupně vynulujeme všechny další prvky posledního řádku. Při tom matice $\mathbf{F}$ mění koeficienty, ale zachovává trojúhelníkový tvar a postupně se mění na horní trojúhelníkovou matici $\bar{\mathbf{F}}$.

Postup ukážeme při nulování i-tého prvku posledního řádku, to je při (i)-té iteraci popisovaného sekvenčního postupu. Ortogonální matice $\mathbf{T}$ je v tomto případě rovna

$$\mathbf{T} = \begin{bmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & & c_i & \dots & s_i \\ & \dots & & & & \\ & & & -s_i & & c_i \end{bmatrix}$$

Ortogonální matice $\mathbf{T}$ je rovna jednotkové matici s výjimkou prvků $T_{i,i} = c_i$, $T_{i,\nu} = s_i$, $T_{\nu,i} = -s_i$ a $T_{\nu,\nu} = c_i$, kde ν je dimenze matice $\mathbf{T}$. Aby matice $\mathbf{T}$ byla ortogonální maticí, musí být koeficienty c_i a s_i omezeny vztahem $c_i^2 + s_i^2 = 1$. Násobíme-li takovou ortogonální maticí matici $\begin{bmatrix} \mathbf{F} \\ \mathbf{h} \end{bmatrix}$, pak se ve výsledné matici změní pouze i-tý a poslední ν-tý řádek. Pro tyto řádky platí

$$\mathbf{T}\begin{bmatrix} \mathbf{F} \\ \mathbf{h} \end{bmatrix}_{i,\nu;\ .} = \begin{bmatrix} c_i & \dots & s_i \\ \dots & & \\ -s_i & \dots & c_i \end{bmatrix} \begin{bmatrix} \dots & 0 & F_{i,i} & F_{i,i+1} & \dots & F_{i,\nu} \\ \dots & & & & & \\ \dots & 0 & h_i^{(i-1)} & h_{i+1}^{(i-1)} & \dots & h_\nu^{(i-1)} \end{bmatrix} =$$

$$= \begin{bmatrix} \dots & 0 & \bar{F}_{i,i} & \bar{F}_{i,i+1} & \dots & \bar{F}_{i,\nu} \\ \dots & & & & & \\ \dots & 0 & 0 & h_{i+1}^{(i)} & \dots & h_\nu^{(i)} \end{bmatrix}$$

kde horním indexem v závorce jsme označili krok (iteraci) algoritmu. Protože chceme nulovat i-tý prvek posledního řádku, musí platit $h_i^{(i)} = -s_i F_{i,i} + c_i h_i^{(i-1)} = 0$. Odtud

plynou vztahy pro prvky c_i a s_i ortogonální matice $\mathbf{T}$

$$c_i = \frac{F_{i,i}}{\sqrt{(F_{i,i})^2 + (h_i^{(i-1)})^2}}, \qquad s_i = \frac{h_i^{(i-1)}}{\sqrt{(F_{i,i})^2 + (h_i^{(i-1)})^2}}.$$

Přitom jediná potíž by nastala, pokud by jmenovatel v obou předchozích výrazech byl roven nule. To ale může nastat pouze tehdy, když bude prvek $h_i^{(i-1)} = 0$ (oba prvky ve jmenovateli jsou v kvadrátech). Potom je ale redukce i-tého prvku posledního řádku již hotová a není ji tudíž nutno provádět. To znamená, že i-tý krok redukce provádíme pouze tehdy, když $h_i^{(i-1)} \neq 0$.

Předchozí postup můžeme použít i pro transformaci matice dat $\mathbf{H}$ do Choleskyho faktoru $\mathbf{F}$, při čemž platí $\mathbf{M} = \mathbf{H}^T\mathbf{H} = \mathbf{F}^T\mathbf{F}$. Transformace se provádí buď po jednotlivých prvcích ve sloupcích nebo řádcích již popsaným postupem pomocí vhodně zvolených ortogonálních matic.

Jiný, efektivnější, algoritmus transformace matice dat $\mathbf{H}$ do Choleskyho faktoru $\mathbf{F}$ je **Householderova reflexe**. Pomocí specielní ortogonální matice $\mathbf{P}$ vynulujeme všechny prvky matice dat $\mathbf{H}$ v jednom sloupci (kromě prvního) najednou.

Myšlenku metody ukážeme nejprve na vektoru. Vektor $\mathbf{x} = [x_1, \ldots, x_n]^T$ transformujeme na vektor $\overline{\mathbf{x}} = \mathbf{P}\mathbf{x} = [\overline{x}_1, 0, \ldots, 0]^T$ pomocí symetrické ortogonální matice $\mathbf{P}$

$$\mathbf{P} = \mathbf{I} - \frac{2}{\mathbf{v}^T\mathbf{v}}\mathbf{v}\mathbf{v}^T \tag{3.16}$$

kde vektor $\mathbf{v}$ je roven

$$\mathbf{v} = \mathbf{x} - \|\mathbf{x}\|_2\mathbf{e}_1 \tag{3.17}$$

kde $\mathbf{e}_1 = [1, 0, \ldots 0]^T = \mathbf{I}(:, 1)$. Znamená to, že vektor $\mathbf{v}$ je rovný vektoru $\mathbf{x}$ ve všech složkách kromě první, která je rovna $v_1 = x_1 - \|\mathbf{x}\|_2$. Při tom norma vektoru $\mathbf{x}$ je samozřejmě $\|\mathbf{x}\|_2 = \sqrt{\mathbf{x}^T\mathbf{x}} = \left(\sum x_i^2\right)^{0.5}$. Předchozí tvrzení snadno dokážeme přímým výpočtem vektoru $\overline{\mathbf{x}}$. Platí

$$\overline{\mathbf{x}} = \mathbf{P}\mathbf{x} = \left(\mathbf{I} - \frac{2\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}}\right)\mathbf{x} = \mathbf{x} - \frac{2\mathbf{v}(\mathbf{v}^T\mathbf{x})}{\mathbf{v}^T\mathbf{v}} = \mathbf{x} - \frac{2\mathbf{v}^T\mathbf{x}}{\mathbf{v}^T\mathbf{v}}\mathbf{v}$$

pro $\mathbf{v}$ dle (3.17) dostaneme

$$\overline{\mathbf{x}} = \mathbf{P}\mathbf{x} = \left(1 - \frac{2\mathbf{v}^T\mathbf{x}}{\mathbf{v}^T\mathbf{v}}\right)\mathbf{x} - \frac{2\mathbf{v}^T\mathbf{x}}{\mathbf{v}^T\mathbf{v}}\|\mathbf{x}\|_2\mathbf{e}_1 = \|\mathbf{x}\|_2\mathbf{e}_1$$

Aby předchozí vztah platil, musí být $\dfrac{2\mathbf{v}^T\mathbf{x}}{\mathbf{v}^T\mathbf{v}} = 1$. To snadno prokážeme přímým dosazením

$$2\mathbf{v}^T\mathbf{x} - \mathbf{v}^T\mathbf{v} = \left(2\mathbf{x}^T\mathbf{x} - 2\|\mathbf{x}\|_2\mathbf{e}_1\mathbf{x}\right) - \left(\mathbf{x}^T\mathbf{x} - 2\|\mathbf{x}\|_2\mathbf{e}_1^T\mathbf{x} + \|\mathbf{x}\|_2^2\mathbf{e}_1^T\mathbf{e}_1\right) = 0$$

protože $\mathbf{x}^T\mathbf{x} = \|\mathbf{x}\|_2^2$ a $\mathbf{e}_1^T\mathbf{e}_1 = 1$.

Touto transformací můžeme převádět matici dat na horní trojůhelníkovou matici po celých sloupcích. Popsaný postup aplikujeme nejprve na celou matici dat $\mathbf{H}$ a pomocí ortogonální matice $\mathbf{P}$ vynulujeme celý první sloupec kromě prvního prvku. Pak postup aplikujeme na submatici matice $\mathbf{H}$, ve které vynecháme první řádek a první sloupec a popsaným postupem vynulujeme druhý sloupec submatice, opět kromě prvního prvku. Tento postup znovu opakujeme na zmenšenou submatici.

3.5 Aktualizace LDU faktorů

Podobný problém je aktualizace LDU faktorů, který můžeme zobecnit na problém transformace matice $\mathbf{F}$ a diagonální matice $\mathbf{G} = \text{diag}\,(\mathbf{g})$ na monickou horní trojúhelníkovou matici $\mathbf{U}$ a diagonální matici $\mathbf{D} = \text{diag}\,(\mathbf{d})$, že platí

$$\mathbf{M} = \mathbf{F}^T\mathbf{G}\mathbf{F} = \mathbf{U}^T\mathbf{D}\mathbf{U}$$

Poznámka: Pokud provádíme aktualizaci "starého" rozkladu novými daty obsaženými v matici dat $\mathbf{h}$, pak

$$\mathbf{F} = \begin{bmatrix} \mathbf{U}_s \\ \mathbf{h} \end{bmatrix}, \qquad \mathbf{G} = \text{diag} \begin{pmatrix} \varphi^2\mathbf{d}_s \\ 1 \end{pmatrix},$$

kde $\mathbf{U}_s$ je horní trojúhelníková matice "starého" rozkladu, $\mathbf{d}_s$ je vektor v diagonále diagonální matice $\mathbf{D}$ "starého" rozkladu a φ je koeficient zapomínání starých dat. □

Nejprve si zavedeme pojem **dyáda**, což je matice vzniklá násobením dvou vektorů, např. $\mathbf{f}\alpha\mathbf{f}^T$, kde $\mathbf{f}$ je sloupcový vektor a α je skalár. Dyáda je tedy matice, která má hodnost nejvýše rovnou jedné.

Podle předchozí rovnice si můžeme představit matici $\mathbf{M}$ jako součet dyád

$$\mathbf{M} = \sum_{i=1}^{\nu} \mathbf{f}_i g_i \mathbf{f}_i^T = \sum_{i=1}^{\nu} \mathbf{u}_i d_i \mathbf{u}_i^T$$

kde $\mathbf{f}_i^T$ a $\mathbf{u}_i^T$ jsou i-té řádky matic $\mathbf{F}$ a $\mathbf{U}$ a g_i, d_i jsou i-té diagonální prvky diagonálních matic $\mathbf{G}$ a $\mathbf{D}$.

Z předchozího součtu všech dyád uvažujme dále pouze součet dvou dyád, které tvoří matici, jejíž hodnost je rovna nejvýše dvěma

$$\mathbf{M}^{(i,j)} = \mathbf{f}_i g_i \mathbf{f}_i^T + \mathbf{f}_j g_j \mathbf{f}_j^T \tag{3.18}$$

kde vektory $\mathbf{f}_i$ a $\mathbf{f}_j$ jsou rovny

$$\begin{aligned} \mathbf{f}_i^T &= \begin{bmatrix} 0, & \ldots, & 0, & 1 & , & f_{i,i+1}, & \ldots, & f_{i,\nu} \end{bmatrix} \\ \mathbf{f}_j^T &= \begin{bmatrix} 0, & \ldots, & 0, & f_{j,i}, & & f_{j,i+1}, & \ldots, & f_{j,\nu} \end{bmatrix} \end{aligned}$$

a g_i i g_j jsou nezáporné konstanty. Uvědomme si, že matice $\mathbf{M}^{(i,j)}$ je vskutku matice rozměru (ν, ν) tvořená součtem dvou dyád, platí tedy pro celou matici $\mathbf{M} = \sum_i \sum_j \mathbf{M}^{(i,j)}$. Pokud provádíme aktualizaci novými daty, pak $j = \nu + 1$ a $\mathbf{f}_{\nu+1}^T = \mathbf{h}$, $g_j = 1$.

Naším záměrem je provést modifikaci předchozích dyád tak, aby prvek $f_{i,i}$ vektoru první dyády zůstal roven jedné a prvek $f_{j,i}$ vektoru druhé diády se vynuloval a nulové prvky v obou vektorech se nezměnily. Modifikace má tedy tvar

$$\mathbf{M}^{(i,j)} = \mathbf{u}_i d_i \mathbf{u}_i^T + \mathbf{u}_j d_j \mathbf{u}_j^T \tag{3.19}$$

kde nové vektory $\mathbf{u}_i$ a $\mathbf{u}_j$ jsou rovny

$$\begin{aligned} \mathbf{u}_i^T &= \begin{bmatrix} 0, & \ldots, & 0, & 1 & , & u_{i,i+1}, & \ldots, & u_{i,\nu} \end{bmatrix} \\ \mathbf{u}_j^T &= \begin{bmatrix} 0, & \ldots, & 0, & 0 & , & u_{j,i+1}, & \ldots, & u_{j,\nu} \end{bmatrix} \end{aligned}$$

a d_i i d_j jsou nezáporné konstanty.

Modifikaci součtu dvou dyád provedeme pomocí **algoritmu dyadické redukce**. Prvky modifikovaných dyád jsou rovny

$$\begin{aligned} d_i &= g_i + (f_{j,i})^2 g_j \\ d_j &= \left(\frac{g_j}{d_i}\right) g_i \\ \mu &= \left(\frac{g_j}{d_i}\right) f_{j,i} \\ u_{j,k} &= f_{j,k} - f_{j,i}\ f_{i,k}, \qquad k = i+1, \ldots, \nu \\ u_{i,k} &= f_{i,k} + \mu\ u_{j,k}, \qquad k = i+1, \ldots, \nu \end{aligned} \tag{3.20}$$

Odvození předchozích vztahů:

Z rovnosti vztahů (3.18) a (3.19) pro staré a nové dyády plyne vztah pro prvek k, l matice $\mathbf{M}^{(i,j)}$

$$\mathbf{M}_{k,l}^{(i,j)} = f_{i,k}\ g_i\ f_{i,l} + f_{j,k}\ g_j\ f_{j,l} = u_{i,k}\ d_i\ u_{i,l} + u_{j,k}\ d_j\ u_{j,l} \tag{3.21}$$

Z předchozího vztahu plyne pro $k = l = i$

$$f_{i,i}\ g_i\ f_{i,i} + f_{j,i}\ g_j\ f_{j,i} = u_{i,i}\ d_i\ u_{i,i} + u_{j,i}\ d_j\ u_{j,i}$$

Vezmeme-li v úvahu, že $f_{i,i} = 1$, $u_{i,i} = 1$ a $u_{j,i} = 0$, pak z předchozí rovnice dostaneme hledaný vztah pro prvek d_i

$$d_i = g_i + (f_{j,i})^2\, g_j \tag{3.22}$$

Pro $l > i$ a $k = i$ plyne z (3.21)

$$f_{i,i}\ g_i\ f_{i,l} + f_{j,i}\ g_j\ f_{j,l} = u_{i,i}\ d_i\ u_{i,l} + u_{j,i}\ d_j\ u_{j,l}$$

Protože $u_{j,i} = 0$, pak z předchozího vztahu dostaneme

$$u_{i,l} = \frac{1}{d_i}\left(g_i\ f_{i,l} + f_{j,i}\ g_j\ f_{j,l}\right)$$

Nyní dosadíme do (3.21) za $u_{i,l}$ a $u_{i,k}$ podle předchozího vztahu a dostaneme

$$f_{i,k}\left[g_i - \frac{g_i^2}{d_i}\right] f_{i,l} + f_{j,k}\left[g_j - \frac{(f_{j,i})^2 g_j^2}{d_i}\right] f_{j,l} - f_{i,k}\left[\frac{g_i g_j f_{j,i}}{d_i}\right] f_{j,l} - f_{j,k}\left[\frac{f_{j,i} g_i g_j}{d_i}\right] f_{i,l} = u_{j,k} d_j u_{j,l} \tag{3.23}$$

Upravíme výrazy v hranatých závorkách v předchozím vztahu

$$g_i - \frac{g_i^2}{d_i} = g_i \frac{d_i - g_i}{d_i} = \frac{g_i (f_{j,i})^2 g_j}{d_i} \qquad g_j - \frac{(f_{j,i})^2 g_j^2}{d_i} = g_j \frac{d_i - (f_{j,i})^2 g_j}{d_i} = \frac{g_j g_i}{d_i}$$

kde jsme využili (3.22). Po dosazení upravených výrazů můžeme (3.23) vyjádřit ve tvaru

$$\left(f_{j,k} - f_{i,k} f_{j,i}\right) \frac{g_i g_j}{d_i} \left(f_{j,l} - f_{i,l} f_{j,i}\right) = u_{j,k} d_j u_{j,l}$$

Odtud plynou další dva vztahy dyadického algoritmu (3.20) a sice

$$u_{j,k} = f_{j,k} - f_{i,k} f_{j,i}, \qquad d_j = \left(\frac{g_j}{d_i}\right) g_i$$

Pro $l = i$ a $k > i$ plyne z (3.21) vztah

$$f_{i,k} g_i f_{i,i} + f_{j,k} g_j f_{j,i} = u_{i,k} d_i u_{i,i} + u_{j,k} d_j u_{j,i}$$

Odtud plyne

$$u_{i,k} = \frac{1}{d_i}\left(g_i f_{i,k} + f_{j,i} g_j f_{j,k}\right) = \frac{g_i}{d_i} f_{i,k} + \frac{f_{j,i} g_j}{d_i}\left(u_{j,k} + f_{i,k} f_{j,i}\right)$$

Odtud plynou poslední vztahy algoritmu dyadických redukcí a sice

$$u_{i,k} = f_{i,k} + \mu u_{j,k}, \quad \text{kde} \quad \mu = \frac{g_j f_{j,i}}{d_i}$$

neboť dle (3.22) je roven jedné koeficient u $f_{i,k}$. Dyadickou redukci můžeme provádět pouze tehdy, když $d_i \neq 0$. Aby $d_i = 0$ musí podle (3.22) být $g_i = 0$ a současně $(f_{j,i})^2 g_j = 0$. Pokud $f_{j,i} = 0$, není třeba redukci provádět. Redukce je totiž již hotová, protože záměrem redukce je nulování tohoto koeficientu. Pokud $g_j = 0$, pak mohu druhou dyádu úplně vypustit, protože nemá žádný vliv, je totiž celá rovna nule.

Vtip celé dyadické redukce je nulování jednoho prvku druhé dyády, při čemž je nutné, aby první dyáda měla na odpovídajícím místě jednotkový prvek. Jednotkové prvky docílíme pomocí pomocných jednotkových dyád s nulovou váhou. Matice $\mathbf{M}$ je při tomto rozšíření rovna

$$\mathbf{M} = \begin{bmatrix} \mathbf{I} \\ \mathbf{F} \end{bmatrix}^T \text{diag} \begin{pmatrix} \mathbf{0} \\ \mathbf{g} \end{pmatrix} \begin{bmatrix} \mathbf{I} \\ \mathbf{F} \end{bmatrix}$$

Sekvenčním nulováním prvků vybraných dyád a vypouštěním nulových dyád docílíme toho, že matice $\mathbf{U}$ je opět monická horní trojúhelníková matice.

Poznámka: Pokud diagonální prvek $f_{i,i} \neq 1$, ale je nenulový, pak modifikaci dyád můžeme přesto provést a výsledné vztahy jsou

$$\begin{aligned}
d_i &= (f_{i,i})^2 g_i + (f_{j,i})^2 g_j \\
d_j &= \left(\frac{g_j}{d_i}\right) g_i (f_{i,i})^2 \\
\mu &= \left(\frac{g_j}{d_i}\right) f_{j,i} \\
u_{j,k} &= f_{j,k} - \frac{f_{j,i}\, f_{i,k}}{f_{i,i}}, \qquad k = i+1, \ldots, \nu \\
u_{i,k} &= \frac{f_{i,k}}{f_{i,i}} + \mu\, u_{j,k}, \qquad k = i+1, \ldots, \nu
\end{aligned}$$

□

Kapitola 4

Lineární programování

V této kapitole budeme zkoumat systémy (matematické modely), které jsou popsány soustavou lineárních rovnic a nerovnic. Kritéria optimalizace těchto systémů jsou také lineární. Proměnné v těchto systémech nabývají reálných hodnot, které leží v určitých mezích. Takové problémy se tradičně nazývají **lineární programování** (LP).

4.1 Typické problémy vedoucí na LP

Řada reálných problémů vede na úlohu lineárního programování. Uvedeme si nyní některé z nich.

Optimální výrobní program
Máme n výrobních procesů (vyrábíme n různých výrobků) a každý z nich požaduje m různých zdrojů (surovin a pod.). Na výrobu jednoho výrobku j-tého druhu potřebujeme a_{ij} jednotek i-tého zdroje. Zdroje jsou omezené a máme k dispozici b_i jednotek i-tého zdroje. Zisk při výrobě jednoho výrobku j-tého typu je c_j. Počty výrobků j-tého typu označíme x_j.

Naší úlohou je vyrábět tak, abychom měli maximální zisk, přičemž musíme respektovat omezení kladená na zdroje surovin. Problém je tedy nalézt

$$\max \{\mathbf{c}^T\mathbf{x} \ : \ \mathbf{A}\mathbf{x} \leq \mathbf{b}, \ \mathbf{x} \geq \mathbf{0} \},$$

kde $\mathbf{x}$ je vektor počtu výrobků, $\mathbf{c}$ je vektor zisků, $\mathbf{b}$ je vektor omezení zdrojů a $\mathbf{A}$ je matice spotřeby s prvky a_{ij}.

Tento lineární model má ale některé nepříjemné a z praktického hlediska neudržitelné důsledky. Jak uvidíme později, ze simplexového algoritmu plyne, že při optimálním výrobním programu má podnik vyrábět alespoň tolik výrobků, kolik má úzkoprofilových zdrojů (to jsou ty zdroje, které úplně využije). To ale znamená, že nastane-li omezení dalšího zdroje, měli bychom na to reagovat rozšířením sortimentu výroby o jeden výrobek, nebo nevyužitím dalšího zdroje. Intuitivně ale čekáme, že na snížení zdrojů bychom měli reagovat úplně opačně - zúžit sortiment a naopak plně využívat zdroje. Promyslete si podrobně tuto úvahu.

Směšovací problém
Máme n základních surovin. Úkolem je namíchat základní suroviny tak, aby výsledný výrobek měl předepsané složení a surovinové náklady byly minimální. Množství jednotek suroviny j-tého typu označíme x_j, její cena za jednotku je c_j. Požadované složení výsledného produktu je popsáno vektorem $\mathbf{b}$, jehož složky b_i jsou rovny požadovanému obsahu látky i ve výsledném produktu. Jednotkové množství základní suroviny j-tého typu obsahuje a_{ij} jednotek látky typu i. Hledáme tedy

$$\min \{\mathbf{c}^T\mathbf{x} \ : \ \mathbf{A}\mathbf{x} = \mathbf{b}, \ \mathbf{x} \geq \mathbf{0}, \ \}$$

kde matice $\mathbf{A}$ má prvky a_{ij}.

Podobný je problém volby nejekonomičtější diety, spočívající ve volbě n různých druhů potravin tak, aby jejich celková cena byla minimální a přitom jsme splnili požadavky na vyrovnanou dietu.

Dopravní problém
Máme m výrobců a n spotřebitelů. i-tý výrobce vyrábí a_i jednotek zboží a j-tý spotřebitel potřebuje b_j jednotek zboží. Veličina c_{ij} je rovna nákladům na přepravu jednotky zboží od i-tého výrobce k j-tému spotřebiteli. Proměnná x_{ij} je rovna množství jednotek zboží od i-tého výrobce k j-tému spotřebiteli. Chceme rozvést zboží od výrobců ke spotřebitelům s minimálními náklady při respektování omezení. Hledáme tedy

$$\min \left\{ \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \ : \ \sum_{j=1}^{n} x_{ij} \leq a_i, \ \sum_{i=1}^{m} x_{ij} = b_j, \ x_{ij} \geq 0, \ \sum_{i=1}^{m} a_i \geq \sum_{j=1}^{n} b_j \right\}$$

Distribuční problém
Zde se jedná o matematický model problému optimálního rozpisu výroby na stroje či jiná zařízení.

Pro každý z m strojů máme určit, kolik výrobků typu 1 až n se na něm bude vyrábět. Přitom jsou známy počty hodin a_i, $1 \leq i \leq m$, které jsou k dispozici na jednotlivých strojích a také požadované množství výrobků b_j, $1 \leq j \leq n$, typu j. Konstanty c_{ij} jsou náklady na hodinovou práci i-tého stroje při výrobě j-tého výrobku, k_{ij} je hodinový výkon i-tého stroje při výrobě j-tého výrobku a proměnné x_{ij} jsou počty hodin i-tého stroje, po které bude stroj vyrábět j-tý výrobek. Tento problém je specielní úloha lineárního programování

$$\min \left\{ \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \ : \ \sum_{j=1}^{n} x_{ij} \leq a_i, \ \sum_{i=1}^{m} k_{ij}x_{ij} = b_j, \ x_{ij} \geq 0, \ \right\}$$

Mnoho problémů optimálního řízení dynamických systémů je možno převést na úlohu lineárního programování. Typický příklad je optimální řízení lineárního diskrétního dynamického systému s omezením velikosti vstupních veličin a lineárním kritériem (např. časově optimální diskrétní řízení).

Také maticová hra, která je modelem antagonistického konfliktu dvou hráčů s konečným počtem strategií vede na úlohu lineárního programování.

4.2 Ekvivalentní formy lineárních úloh

V tomto odstavci si uvedeme některé ekvivalentní formy úlohy lineárního programování.

Základní úloha je úloha

$$\max \{\mathbf{c}^T\mathbf{x} \ : \ \mathbf{A}_1\mathbf{x} \leq \mathbf{b}_1, \ \mathbf{A}_2\mathbf{x} = \mathbf{b}_2, \ \mathbf{A}_3\mathbf{x} \geq \mathbf{b}_3, \ \mathbf{x} \geq \mathbf{0}, \ \} \tag{4.1}$$

Charakteristika základní úlohy - je to úloha na maximum či minimum, omezení jsou ve tvaru rovnosti i nerovností obou typů, proměnné jsou nezáporné.

Standardní úloha je úloha

$$\max \{\mathbf{c}^T\mathbf{x} \ : \ \mathbf{A}_1\mathbf{x} \leq \mathbf{b}_1, \ \mathbf{A}_2\mathbf{x} \leq \mathbf{b}_2, \ -\mathbf{A}_2\mathbf{x} \leq -\mathbf{b}_2, \ -\mathbf{A}_3\mathbf{x} \leq -\mathbf{b}_3, \ \mathbf{x} \geq \mathbf{0}, \ \} \tag{4.2}$$

Charakteristika standardní úlohy - je to úloha na maximum, omezení ve tvaru nerovností jednoho typu, proměnné jsou nezáporné.

Kanonický tvar úlohy lineárního programování je úloha typu

$$\max\{\mathbf{c}^T\mathbf{x} : \mathbf{A}_1\mathbf{x} + \mathbf{u} = \mathbf{b}_1, \mathbf{A}_2\mathbf{x} = \mathbf{b}_2, \mathbf{A}_3\mathbf{x} - \mathbf{v} = \mathbf{b}_3, \mathbf{x} \geq \mathbf{0}, \mathbf{u} \geq \mathbf{0}, \mathbf{v} \geq \mathbf{0}, \} \tag{4.3}$$

Charakteristika úlohy v kanonickém tvaru - je to úloha na maximum, omezení ve tvaru rovností, proměnné jsou nezáporné.

Normální úloha lineárního programování je úloha na maximum, omezení ve tvaru $\mathbf{A}\mathbf{x} \leq \mathbf{b}$, ve kterém jsou nezáporné pravé strany omezení ($\mathbf{b} \geq \mathbf{0}$).

4.3 Grafické řešení optimalizace lineárních modelů

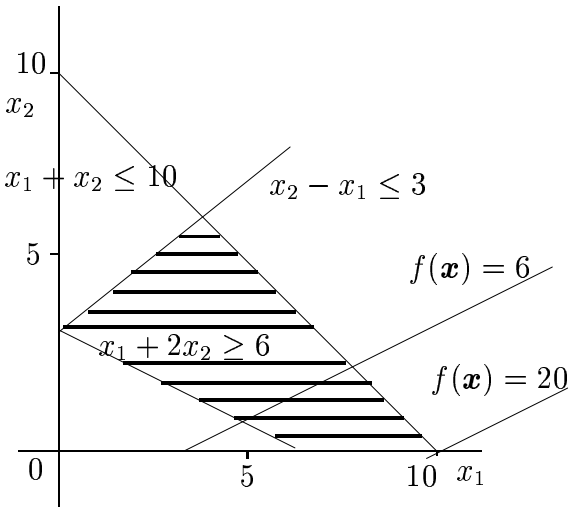
Pro lepší představu ukážeme nyní grafické řešení lineárních úloh. Abychom problém mohli řešit v rovině, budeme uvažovat problém, ve kterém se vyskytují pouze dvě proměnné.

Jak bude ukázáno později je množina $\mathcal{X}$, určená lineárními omezeními ve tvaru $\mathcal{X} = \{\mathbf{x}|\mathbf{A}\mathbf{x} \leq \mathbf{b}\}$ vypuklá množina a proto lineární funkcionál definovaný na vypuklé množině má optimum na hranici množiny. Je-li množina přípustných řešení vypuklý mnohostěn, pak optimum nastává v některém krajním bodě přípustné množiny.

Příklad: Mějme následující lineární problém

$$\max \left\{ 2x_1 - 3x_2 \left| \begin{array}{rcl} x_1 + 2x_2 & \geq & 6 \\ x_2 - x_1 & \leq & 3 \\ x_1 + x_2 & \leq & 10 \end{array} \right. , \ x_1, \ x_2 \geq 0 \right\}.$$

Grafické řešení je patrné z obr. 4.1. Každé omezení definuje přímku v rovině o souřadnicích $x_1, \ x_2$, která rovinu rozděluje na dvě části, z nichž pouze jedna je přípustná podle znaménka nerovnosti. Tři omezení spolu s podmínkami nezápornosti proměnných omezují množinu přípustných řešení, kterou je zde vypuklý mnohoúhelník. Kritérium $J = f(\mathbf{x}) = 2x_1 - 3x_2$ určuje pro různé velikosti J svazek rovnoběžných přímek, které pro dosažitelnou hodnotu kritéria se protínají s přípustnou oblastí. Pro $J = 20$ se přímka dotýká množiny



Obrázek 4.1: Grafické řešení příkladu

přípustných řešení pouze v jednom vrcholu a pro $J > 20$ se přímky s množinou přípustných řešení neprotínají. Řešením naší úlohy je $\max(J) = 20$, které nastává pro $x_1 = 10$ a $x_2 = 0$. Zřejmě také platí že $\min(J) = -12.5$, které nastává pro $x_1 = 3.5$ a $x_2 = 6.5$.

Vyřešte graficky duální úlohu (k předchozí primární úloze), která má tvar

$$\min\left\{-6\lambda_1 + 3\lambda_2 + 10\lambda_3 \;\middle|\; \begin{array}{rcl} -\lambda_1 - \lambda_2 + \lambda_3 & \geq & 2 \\ -2\lambda_1 + \lambda_2 + \lambda_3 & \geq & 3 \end{array}, \;\; \lambda_1,\ \lambda_2,\ \lambda_3 \geq 0\right\}.$$

Protože v primární úloze se první omezení neuplatní, můžeme první duální proměnnou vynechat (bude rovna nule) a duální úlohu řešit také pouze v rovině dvou duálních proměnných. Řešení duální úlohy je stejné jako úlohy primární $\min\{6\lambda_1+3\lambda_2+10\lambda_3\} = 20$, které nastává pro $\lambda_1 = 0$, $\lambda_2 = 0$ a $\lambda_3 = 2$. Z řešení plyne, že ani druhé omezení primární úlohy se neuplatnilo a stínová cena třetího omezení je rovna hodnotě třetí duální proměnné. Zvětšíme-li pravou stranu omezení o jedna (z 10 na 11), pak optimální hodnota kritéria vzroste z 20 na 22.

Protože řešíme úlohu na maximum, pak derivace kritéria vzhledem k omezením je rovna $+\lambda^T$ a ne $-\lambda^T$, jak tomu bylo v úloze na minimum.

4.4 Předběžná analýza problému

Analyzujme nejprve následující jednoduchou úlohu

$$\max\left\{2x_1 + 3x_2 + 7x_3 + 9x_4 \;\middle|\; \begin{array}{rcl} x_1 + x_2 + x_3 + x_4 & \leq & 9 \\ x_1 + 2x_2 + 4x_3 + 8x_4 & \leq & 24 \end{array}, \quad x_i \geq 0\right\}.$$

Pomocí přídavných proměnných x_5 a x_6 převedeme omezení na rovnice. Omezení mají tedy tvar

$$x_1 + x_2 + x_3 + x_4 + x_5 \;\; = \;\; 9, \qquad x_1 + 2x_2 + 4x_3 + 8x_4 + x_6 = 24$$

Vidíme, že kritérium nejvíce závisí na koeficientu x_4 (u proměnné x_4 je největší kladný koeficient rovný 9). Z rovnic omezení plyne, že maximální hodnota koeficientu x_4 je $(x_4)_{max} = 3$. Kritérium potom bude rovno $f(\mathbf{x}) = 27$ (při nulových hodnotách ostatních proměnných). To ale není optimální hodnota kritéria, poněvadž například při maximální hodnotě koeficientu x_3, který může být roven $(x_3)_{max} = 6$ bude kritérium rovno $f(\mathbf{x}) = 42$.

Bude zřejmě nejvýhodnější vybírat různé kombinace proměnných. Předpokládejme, že optimum lze nalézt při kladných hodnotách proměnných x_2 a x_3 a nulových hodnotách ostatních proměnných. Z rovnic omezení vyjádříme zvolené proměnné x_2 a x_3 jako funkce ostatních proměnných. Z rovnic omezení plyne

$$\begin{aligned} x_2 + x_3 &= 9 - x_1 - x_4 - x_5 \\ 2x_2 + 4x_3 &= 24 - x_1 - 8x_4 - x_6 \end{aligned}$$

Po Gaussově eliminaci dostaneme

$$\begin{aligned} x_2 &= 6 - \frac{3}{2}x_1 + 2x_4 - 2x_5 + \frac{1}{2}x_6 \\ x_3 &= 3 + \frac{1}{2}x_1 - 3x_4 + x_5 - \frac{1}{2}x_6 \end{aligned}$$

Pro $x_1 = x_4 = x_5 = x_6 = 0$ je tedy $x_2 = 6$ a $x_3 = 3$. Kritérium je potom rovno $f(\mathbf{x}) = 39$. Jak zjistíme zda je to optimum? Dosadíme x_2 a x_3 do kritéria, pak

$$f(\mathbf{x}) = 39 + x_1 - 6x_4 + 1x_5 - 2x_6$$

Odtud je zřejmé, že zvětšení proměnné x_1 a x_5 z nuly do kladných hodnot povede ke zvýšení kritéria. Je to zřejmé z toho, že v předchozím výrazu pro kritérium jsou u uvedených proměnných kladné koeficienty.

Je zřejmé, že zvětšení x_1 o jednotku (při $x_4 = x_5 = x_6 = 0$) povede ke zmenšení proměnné x_2 o 3/2 a zvětšení proměnné x_3 o 1/2. Protože všechny proměnné musí být nezáporné, pak maximální hodnota proměnné x_1 je $(x_1)_{max} = 4$. Potom ale je $x_2 = 0$ a $x_3 = 5$ a kritérium je rovno $f(\mathbf{x}) = 39 + 4 = 43$.

Nyní opět z rovnic omezení vyjádříme uvedené proměnné x_1 a x_3. Po eliminaci platí

$$\begin{aligned} x_1 &= 4 - \frac{2}{3}x_2 + \frac{4}{3}x_4 - \frac{4}{3}x_5 + \frac{1}{3}x_6 \\ x_3 &= 5 - \frac{1}{3}x_2 - \frac{7}{3}x_4 + \frac{1}{3}x_5 - \frac{1}{3}x_6. \end{aligned}$$

Pro $x_2 = x_4 = x_5 = x_6 = 0$ a $x_1 = 4$, $x_3 = 5$ je kritérium rovno $f(\mathbf{x}) = 2 * 4 + 7 * 5 = 43$. Po dosazení výrazů pro x_2 a x_3 do kritéria dostaneme

$$f(\mathbf{x}) = 43 - \frac{2}{3}x_2 - \frac{14}{3}x_4 - \frac{1}{3}x_5 - \frac{5}{3}x_6$$

Protože všechna znamémka koeficientů u proměnných x_2, x_4 x_5 a x_6 ve výrazu pro kritérium jsou záporná, je zřejmé, že při libovolné kladné hodnotě těchto koeficientů nastane pokles kritéria. Proto $x_1 = 4$ a $x_3 = 5$ zajišťuje maximum kritéria (při nulové hodnotě ostatních proměnných), tedy $\mathbf{x}^* = \begin{bmatrix} 4, & 0, & 5, & 0 \end{bmatrix}^T$ je řešením naší úlohy.

Simplexová metoda, která bude popsána v následujícím odstavci, pouze jednodušším způsobem kopíruje předchozí postup.

4.5 Simplexová metoda

V tomto odstavci popíšeme základní verzi simplexové metody řešení úlohy lineárního programování. Pomocí simplexového algoritmu budeme řešit následující problém

$$\max\left\{2x_1+3x_2+7x_3+9x_4 \;:\; \begin{array}{rcl} x_1+x_2+x_3+x_4 & \leq & 9 \\ x_1+2x_2+4x_3+8x_4 & \leq & 24 \end{array},\quad x_i\geq 0\right\}.$$

Tento problém je stejný jako úloha řešená v odstavci pojednávajícím o předběžné analýze problému. Zavedením přídavných proměnných x_5 a x_6 převedeme omezení ve tvaru nerovnosti na omezení ve tvaru rovnosti

$$\begin{aligned} x_1+x_2+x_3+x_4+x_5 &= 9, \\ x_1+2x_2+4x_3+8x_4+x_6 &= 24. \end{aligned}$$

Omezení i upravené kritérium budeme zapisovat do tabulky, nazývané **simplexová tabulka**, v jejíž sloupcích jsou pouze koeficienty u proměnných v příslušných omezeních a v kritériu.

x_1	x_2	x_3	x_4	x_5	x_6	**b**
1	1	1	1	1	0	9
1	2	4	8	0	1	24
-2	-3	-7	**-9**	0	0	0

Simplexová tabulka v první části (v prvních dvou řádcích) popisuje naše dvě omezení úlohy. Z prvního omezení plyne $x_5 = 9-(x_1+x_2+x_3+x_4)$ a z druhého omezení zase plyne $x_6 = 24-(x_1+2x_2+4x_3+8x_4)$. Odtud dostaneme základní řešení $x_5 = 9$, $x_6 = 24$ (a ostatní proměnné nulové). Toto řešení přímo čteme v posledním sloupci předchozí tabulky. Je to proto, že pátý a šestý sloupec matice $\mathbf{A}$ tvoří jednotkovou submatici.

Poslední řádek vyjadřuje kritérium zapsané ve tvaru $f(\mathbf{x}) = 0-(-2x_1-3x_2-7x_3-9x_4+0x_5+0x_6)$. Do posledního řádku tedy píšeme koeficienty u příslušných proměnných a 0 v posledním sloupci (posledního řádku) vyjadřuje hodnotu kritéria při základním řešení $x_5 = 9$, $x_6 = 24$ (a ostatní proměnné rovné nule). Kritérium je pochopitelně rovno nule, protože ani x_5 ani x_6, které jsou nenulové, se v kritériu nevyskytují.

Proměnné x_5 a x_6 jsou **bázové proměnné**. Sloupcové vektory matice omezení jim odpovídající tvoří jednotkovou submatici. Kritérium je upraveno tak, že koeficienty u bázových proměnných jsou nulové. Potom v posledním sloupci (sloupec označený **b**) čteme jednak hodnoty bázových proměnných a v posledním řádku velikost kritéria. Tuto základní charakteristiku simplexové tabulky budeme dodržovat i při dalších iteracích.

Nyní se podíváme na koeficienty v posledním řádku předchozí tabulky. Některé koeficienty v posledním řádku jsou záporné a to značí, že pokud příslušné proměnné jim odpovídající budou nenulové, hodnota kritéria vzroste. Proto najdeme minimální koeficient v posledním řádku (záporný koeficient s největší absolutní hodnotou). Sloupec jemu odpovídající se nazývá **klíčový sloupec**.

V našem případě je to koeficient (-9) (je pro názornost vyznačen tučně), proto je čtvrtý sloupec klíčovým sloupcem. To znamená, že příslušná proměnná odpovídající

klíčovému sloupci bude nenulová. V našem případě tedy bude nenulová čtvrtá proměnná a kritérium bude dosahovat vyšší hodnoty. Můžeme tedy vyslovit následující tvrzení:

Simplexové kritérium I.: *Jestliže v posledním řádku simplexové tabulky jsou některé koeficienty u nebázových proměnných záporné, pak jako novou bázovou proměnnou vybereme tu proměnnou, která má záporný a v absolutní hodnotě maximální koeficient v posledním řádku. Pokud všechny koeficienty v posledním řádku jsou nulové nebo kladné, je získané řešení optimální.*

□

Poté co jsme rozhodli (pomocí klíčového sloupce), která proměnná bude tvořit novou bázovou proměnnou, musíme zjistit, která proměnná z báze vypadne. Z první rovnice omezení plyne, že pokud proměnná x_5 bude nulová, může být proměnná x_4 nejvýše rovna 9. Z druhé rovnice omezení plyne, že pokud proměnná x_6 bude nulová, pak proměnná x_4 může být nejvýše rovna 3. Toto omezení na volbu velikosti čtvrté proměnné je přísnější a proto jej musíme respektovat. Odtud plyne následující tvrzení:

Simplexové kritérium II.: *Nechť j je index klíčového sloupce. Pro všechna $a_{ij} > 0$ vypočteme podíly koeficientů b_i ve sloupci* **b** *a prvků a_{ij} v j-tém sloupci. Vypočteme tedy podíly*

$$\frac{b_i}{a_{ij}} \quad \text{pro} \quad \forall i, \quad \text{pro která} \quad a_{ij} > 0. \tag{4.4}$$

Nyní vyberu takové $i = k$, aby $\frac{b_k}{a_{kj}} = \min_i \frac{b_i}{a_{ij}}$. Prvek s indexem (k, j) je **klíčový prvek**. *Znamená to, že proměnná k vstoupí do nové báze a příslušná bázová proměnná odpovídající k-tému řádku z báze vypadne. Pokud pro všechna i platí $a_{ij} < 0$, pak úloha nemá konečné řešení.*

□

V našem případě je klíčový prvek v druhém řádku a čtvrtém sloupci $(k, j) = (2, 4)$. Tento prvek je pro názornost v předchozí tabulce umístěn v rámečku.

Tím jsme ukončili rozhodování, jak pokračovat v řešení. Novou iteraci píšeme do nové tabulky, kterou umístíme pod naší tabulku, která popisovala předchozí iteraci. Tabulku popisující starou iteraci nyní musíme upravit tak, aby klíčový prvek byl roven jedné (v předchozí tabulce je roven 8). Musíme proto celý druhý řádek dělit osmi. Do nové tabulky opíšeme tedy v našem případě koeficienty ve druhém řádku dělené osmi.

Nové bázové proměnné jsou tedy původní bázová proměnná x_5 a nová bázová proměnná x_4, která nahradila původní bázovou proměnnou x_6. Nyní musíme zajistit, aby ve čtvrtém sloupci (sloupci odpovídajícímu nové bázové proměnné) byly nulové koeficienty v prvním řádku i posledním řádku (to je v řádcích, které nemají řádkový index klíčového prvku). To docílíme tak, že klíčový řádek násobený vhodným koeficientem přičteme k příslušným řádkům tak, aby se nulovaly potřebné koeficienty. Dostaneme tak novou tabulku, která obsahuje v prvních třech řádcích původní tabulku a v posledních třech řádcích je další iterace simplexové metody.

x_1	x_2	x_3	x_4	x_5	x_6	**b**
1	1	1	1	1	0	9
1	2	4	$\boxed{8}$	0	1	24
-2	-3	-7	**-9**	0	0	0
7/8	3/4	1/2	0	1	-1/8	6
1/8	2/8	$\boxed{4/8}$	1	0	1/8	3
-7/8	-3/4	**-2.5**	0	0	9/8	27

Z této nové tabulky plynou následující fakta:
První řádek popisuje omezení ve tvaru $x_5 = 6 - (x_1 7/8 + x_2 3/4 + x_3 1/2 + 0x_4 - x_6 1/8)$, z druhého řádku nové tabulky zase plyne $x_4 = 3 - (x_1 1/8 + x_2 2/8 + x_3 4/8 + 0x_5 + x_6 1/8)$. Odtud dostaneme základní řešení $x_5 = 6$, $x_4 = 3$ (a ostatní proměnné nulové). Toto řešení přímo čteme v posledním sloupci předchozí tabulky. Je to proto, že pátý a čtvrtý sloupec modifikované matice $\mathbf{A}$ opět tvoří jednotkovou submatici. Poslední řádek vyjadřuje kritérium ve tvaru $f(\mathbf{x}) - x_1 7/8 - x_2 3/4 - 2.5x_3 + 0x_4 + 0x_5 + x_6 9/8 = 27$. V posledním řádku tedy jsou koeficienty u příslušných proměnných a 27 v posledním sloupci (posledního řádku) vyjadřuje hodnotu kritéria při základním řešení $x_5 = 6$, $x_4 = 3$ (a ostatní proměnné rovné nule). Kritérium vzrostlo proti předchozí iteraci.

Nyní se opět podíváme na poslední řádek simplexové tabulky. Protože některé koeficienty v posledním řádku jsou záporné, není získané řešení optimální. Největší v absolutní hodnotě záporný prvek je -2.5, který leží ve třetím sloupci. Klíčový sloupec je tedy třetí sloupec. Budeme tedy do báze dávat třetí proměnnou. Zbývá rozhodnout, která proměnná bude z báze odstraněna. Podle simplexového kritéria II. vytvoříme podíly $\frac{b_i}{a_{ij}}$, kde $j = 3$ je index klíčového sloupce. Minimální hodnota podílu nastane při $i = 2$ a proto z báze bude odstraněna čtvrtá proměnná. Klíčový prvek má index $i, j = 2, 3$ a je v předchozí tabulce opět v rámečku.

Tím je ukončeno rozhodování jak pokračovat v další iteraci simplexového algoritmu. Vytvoříme tedy novou tabulku, ve které bude jednotka na místě klíčového prvku. Musíme proto celý druhý řádek dělit 4/8.

Také musíme zajistit, aby ve třetím sloupci (sloupci odpovídajícím nové bázové proměnné) byly nulové koeficienty v prvním řádku i posledním řádku (to je v řádcích, které nemají řádkový index klíčového prvku). Proto klíčový řádek násobený vhodným koeficientem přičteme k příslušným řádkům tak, aby se nulovaly potřebné koeficienty.

Dostaneme tak novou tabulku, která kromě této iterace obsahuje další iteraci, kterou již nebudeme popisovat a která nám konečně dá optimální řešení. Optimální řešení je $\mathbf{x}^* = \begin{bmatrix} 4 & 0 & 5 & 0 & 0 & 0 \end{bmatrix}^T$ a optimální hodnota kritéria je $f(\mathbf{x}^*) = 43$. Řešení je optimální, protože všechny koeficienty v posledním řádku simplexové tabulky jsou nezáporné.

Protože v posledním řádku jsou nulové koeficienty pouze u bázových proměnných (a ostatní koeficienty jsou kladné), úloha má pouze jediné řešení.

V této základní úloze můžeme ze simplexové tabulky přímo získat i duální řešení. Je to řešení následující duální úlohy $\min\{\mathbf{b}^T\boldsymbol{\lambda} \ : \ \mathbf{A}^T\boldsymbol{\lambda} \geq \mathbf{c} \ , \ \boldsymbol{\lambda} \geq \mathbf{0}\}$, což v našem případě

Tabulka 4.1: **SIMPLEXOVÁ TABULKA**

x_1	x_2	x_3	x_4	x_5	x_6	**b**
1	1	1	1	1	0	9
1	2	4	[8]	0	1	24
-2	-3	-7	**-9**	0	0	0
7/8	3/4	1/2	0	1	-1/8	6
1/8	2/8	[4/8]	1	0	1/8	3
-7/8	-3/4	**-2.5**	0	0	9/8	27
[3/4]	1/2	0	-1	1	-1/4	3
1/4	1/2	1	2	0	1/4	6
-1/4	5/4	0	5	0	7/4	42
1	2/3	0	-4/3	4/3	-1/3	4
0	1/3	1	7/3	-1/3	1/3	5
0	17/12	0	14/3	1/3	5/3	43

je úloha

$$\min\left\{9\lambda_1+24\lambda_2 \;:\; \begin{array}{lll}\lambda_1+\lambda_2 & \geq & 2\\ \lambda_1+2\lambda_2 & \geq & 3\\ \lambda_1+4\lambda_2 & \geq & 7\\ \lambda_1+8\lambda_2 & \geq & 9\end{array},\quad \lambda_i\geq 0\right\}.$$

Duální řešení čteme v posledním řádku simplexové tabulky ve sloupcích odpovídajících přídavným proměnným (zde proměnná x_5 a x_6). V našem případě je tedy duální řešení $\lambda_1^*=1/3$, $\lambda_2^*=5/3$. Proč tomu tak je, odvodíme pomocí maticového zápisu simplexové metody, viz odst. 4.7.

Protože Lagrangeovy koeficienty mají význam citlivosti kritéria na změnu omezení, znamená to, že změníme-li v prvním omezení konstantu o jedna (místo 9 bude v prvním omezení 10), vzroste optimální hodnota kritéria o $\lambda_1^*=1/3$. Změníme-li ve druhém omezení naší úlohy konstantu o jednotku (místo 24 bude v druhém omezení 25), vzroste optimální hodnota kritéria o $\lambda_2^*=5/3$.

Poznámka: Uvědomme si, že řešíme úlohu na maximum a proto na rozdíl od citlivostní věty bereme zde koeficienty λ_i s kladným znaménkem.

4.6 Vlastnosti množiny přípustných a optimálních řešení

Z povahy úlohy lineárního programování jsou proměnné nezáporné a musí ještě vyhovovat dalším omezujícím podmínkám ve tvaru

$$\mathbf{Ax}=\mathbf{b},\qquad \mathbf{x}\geq\mathbf{0}, \tag{4.5}$$

kde $\mathbf{A}$ je daná matice typu (m, n) a $\mathbf{b}$ je daný vektor rozměru m a vektor $\mathbf{x}$ má rozměr n, kde $m \leq n$. Pokud je omezení dané ve tvaru nerovnosti $\mathbf{Ax} \leq \mathbf{b}$, můžeme použitím pomocných přídatných proměnných změnit omezení na omezení ve tvaru rovnosti. Proto v tomto odstavci budeme uvažovat pouze omezení ve tvaru rovnosti a nezáporné proměnné.

Vektor $\mathbf{x}$, který splňuje omezení se nazývá **přípustné řešení**. Množinu přípustných řešení budeme označovat $\mathcal{X}$. V úloze lineárního programování maximalizujeme lineární kritérium ve tvaru $f(\mathbf{x}) = \mathbf{c}^T\mathbf{x}$.

Přípustné řešení, které maximalizuje kritérium se nazývá **optimální řešení** úlohy lineárního programování. Množinu všech optimálních řešení označíme $\mathcal{X}_{opt}$. Nyní popíšeme vlastnosti množiny přípustných řešení $\mathcal{X}$.

Nejprve označíme sloupce matice $\mathbf{A}$ jako vektory $\mathbf{a}^{(1)}$, $\mathbf{a}^{(2)}$, ... , $\mathbf{a}^{(n)}$, pak tedy

$$\mathbf{A} = \left[\begin{array}{cccc} \mathbf{a}^{(1)}, & \mathbf{a}^{(2)}, & \ldots, & \mathbf{a}^{(n)} \end{array} \right]$$

Protože máme m omezení, jsou vektory $\mathbf{a}^{(i)}$ rozměru m. Omezení (4.5) můžeme zapsat ve tvaru

$$x_1\mathbf{a}^{(1)} + x_2\mathbf{a}^{(2)} + \cdots + x_n\mathbf{a}^{(n)} = \mathbf{b} \tag{4.6}$$

Řešení předchozího vztahu, mající nejvýše m kladných složek x_i vektoru $\mathbf{x}$, se nazývá **základní řešení** soustavy (4.5).

Základní řešení, které má právě m kladných složek x_i se nazývá **nedegenerované**. Vektory $\mathbf{a}^{(i)}$, které mají v lineární kombinaci (4.6) kladné koeficienty se nazývají **základní**. Je-li kladných složek méně než m, nazývá se toto řešení **degenerované**. Degenerované řešení můžeme formálně doplnit na počet m a potom taková soustava základních vektorů doplněná na počet m se nazývá **báze**. Vektory tvořící bázi se nazývají **bázové**. Základních vektorů nemůže být evidentně více než $\left(\begin{array}{c} n \\ m \end{array} \right)$.

Po zavedených definicích můžeme vyslovit následující tvrzení:

Věta: *Bod $\mathbf{x} \in \mathcal{X}$ je základním řešením pouze tehdy, je-li krajním bodem množiny $\mathcal{X}$.*

Předchozím tvrzením je vlastně určen algoritmus výpočtu krajních bodů množiny $\mathcal{X}$. Důkaz předchozího tvrzení je založen na tom faktu, že krajní bod množiny nelze vyjádřit jako konvexní kombinaci jiných dvou různých bodů množiny $\mathcal{X}$.

Je-li možno vyjádřit libovolný bod konvexní množiny jako konvexní kombinaci jejích krajních bodů, pak se taková množina nazývá **konvexní polyedr**. Konvexní polyedr je tedy taková množina $\mathcal{X}$ pro kterou platí

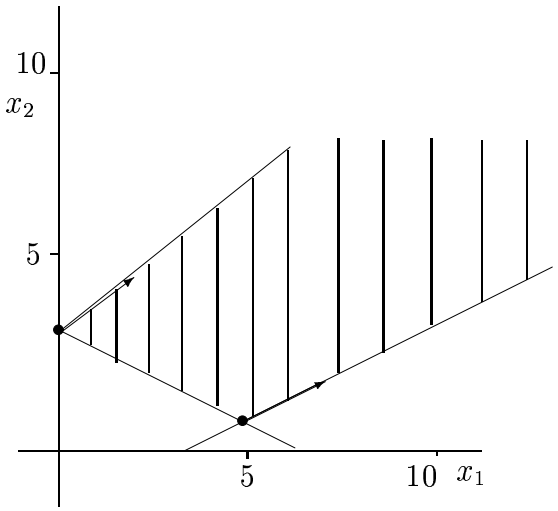
$$\mathbf{x} = \sum_{i=1}^{r} k_i\mathbf{x}^{(i)}, \qquad \sum_{i=1}^{r} k_i = 1, \qquad k_i \geq 0, \quad i = 1,\ 2, \ldots, r,$$

kde $\mathbf{x}^{(1)}$ až $\mathbf{x}^{(r)}$ jsou všechny krajní body množiny $\mathcal{X}$.

To platí pro omezenou konvexní polyedrickou množinu. Neomezenou konvexní polyedrickou množinu charakterizujeme ještě **krajními paprsky** - viz obr. (4.2). Nenulový vektor $\mathbf{y} \in \mathcal{X}$ se nazývá **reprezentant krajního paprsku** množiny $\mathcal{X}$ jestliže platí, že

a) existuje krajní bod $\mathbf{x}^{(i)} \in \mathcal{X}$ takový, že $\mathbf{x}^{(i)} + \alpha\mathbf{y} \in \mathcal{X}$ pro všechna $\alpha \geq 0$.

b) vektor $\mathbf{y}$ se nedá vyjádřit jako lineární kombinace, s kladnými koeficienty, jiných dvou krajních paprsků, lineárně nezávislých na $\mathbf{y}$.



Obrázek 4.2: Krajní body a krajní paprsky konvexní množiny

Každý bod konvexní polyedrické množiny $\mathcal{X}$ je možno vyjádřit ve tvaru

$$\mathbf{x} = \sum_{i=1}^{r} k_i \mathbf{x}^{(i)} + \sum_{j=1}^{s} \alpha_j \mathbf{y}^{(j)}, \ \sum_{i=1}^{r} k_i = 1, k_i \geq 0, i = 1, \ldots, r, \alpha_j \geq 0, j = 1, \ldots, s.$$

Vektor $\mathbf{y}$ je pouze reprezentant krajního paprsku, protože libovolný kladný násobek tohoto vektoru je také reprezentant krajního paprsku. Důležitý je tedy pouze jeho směr a ne jeho velikost. Konvexní polyedr je konvexní polyedrická množina, v níž neexistuje žádný krajní paprsek.

Pro množinu $\mathcal{X} = \{\mathbf{x} \ : \ \mathbf{A}\mathbf{x} = \mathbf{b}\}$ jsme v předchozím tvrzení určili její krajní body. Zbývá tedy ještě určit reprezentanty krajních paprsků. To je obsaženo v následujícím tvrzení:

Věta: *Nechť* $\mathbf{x}$ *je základním řešením, které má prvních* $k \leq m$ *složek kladných a ostatní nulové. Dále nechť je*

$$\begin{aligned} \mathbf{a}^{(1)} &= \begin{bmatrix} 1 & 0 & \ldots & 0 & \ldots \end{bmatrix}^T \\ \mathbf{a}^{(2)} &= \begin{bmatrix} 0 & 1 & \ldots & 0 & \ldots \end{bmatrix}^T \\ &\ldots \\ \mathbf{a}^{(k)} &= \begin{bmatrix} 0 & 0 & \ldots & 1 & \ldots \end{bmatrix}^T \end{aligned}$$

a dále platí $\mathbf{a}^{(k+1)} \leq \mathbf{0}$. *Potom reprezentant krajního paprsku množiny* $\mathcal{X}$ *příslušný k základnímu řešení (krajnímu bodu)* $\mathbf{x}$ *je roven*

$$\mathbf{y} = \begin{bmatrix} -a_1^{(k+1)}, & -a_2^{(k+1)}, \ldots, & -a_k^{(k+1)}, & 1, & 0, & \ldots, & 0 \end{bmatrix}^T. \tag{4.7}$$

Snadno dokážeme, že pro každé $\alpha \geq 0$ platí $\mathbf{x} + \alpha\mathbf{y} \in \mathcal{X}$. Po dosazení za $\mathbf{y}$ platí

$$\mathbf{A}\left(\mathbf{x} + \alpha\mathbf{y}\right) = \mathbf{a}^{(1)}\left(x_1 - \alpha a_1^{(k+1)}\right) + \ldots + \mathbf{a}^{(k)}\left(x_k - \alpha a_k^{(k+1)}\right) + \alpha\mathbf{a}^{(k+1)} = \mathbf{b}.$$

Platí totiž

$$\mathbf{a}^{(1)}x_1 + \ldots + \mathbf{a}^{(k)}x_k = \mathbf{b}$$

neboť $\mathbf{x}$ je základním řešením. Dále platí $\mathbf{a}^{(1)}\alpha a_1^{(k+1)} = \left[\ \alpha a_1^{(k+1)},\ 0,\ \ldots,\ 0,\ \ldots\ \right]^T$ a podobně pro $\mathbf{a}^{(i)}\alpha a_i^{(k+1)}$, $i = 2, \ldots, k$. Proto je nulový následující součet členů

$$\left[\ -\alpha a_1^{(k+1)},\ -\alpha a_2^{(k+1)},\ \ldots,\ -\alpha a_k^{(k+1)},\ \ldots\ \right] + \alpha\mathbf{a}^{(k+1)} = \mathbf{0}$$

Přitom $\mathbf{x} + \alpha\mathbf{y} \geq \mathbf{0}$.

Protože předchozí tvrzení je pouze postačující podmínkou pro reprezentanta krajního paprsku, je třeba doplnit přechozí větu tak, abychom získali všechny krajní paprsky. Ke všem základním řešením s jednotkovými bazickými vektory hledáme všechny sloupce $\mathbf{a}^{(j)} \leq \mathbf{0}$ a z nich tvoříme (lineárně nezávislé) reprezentanty krajních paprsků podle předchozích zásad.

Platí několik velmi důležitých tvrzení:

Množina všech přípustných řešení je konvexní polyedrická množina.

Také množina všech optimálních řešení $\mathcal{X}_{opt}$ je konvexní polyedrická množina.

Přitom platí, že každý krajní bod množiny $\mathcal{X}_{opt}$ je i krajním bodem množiny $\mathcal{X}$ a každý krajní paprsek množiny $\mathcal{X}_{opt}$ je i krajním paprskem množiny $\mathcal{X}$.

Tím jsme dostali jednoduchý návod k řešení úlohy lineárního programování: Nalezneme všechny krajní body a krajní paprsky množiny přípustných řešení $\mathcal{X}$ a v krajních bodech spočteme hodnotu kritéria $\mathbf{c}^T\mathbf{x}$. Protože krajních bodů je konečný počet, je úloha v principu vyřešena. Efektivnější metodou řešení lineárních úloh je simplexový algoritmus, který je popsán v odst. 4.5.

Příklad: Nalezněte krajní body a krajní paprsky množiny $\mathcal{X} = \{\mathbf{x} : \mathbf{A}\mathbf{x} = \mathbf{b},\ \mathbf{x} \geq \mathbf{0}\}$, kde

$$\mathbf{A} = \begin{bmatrix} 2 & 1.5 & -1.5 \\ 3 & 2.5 & -2.2 \end{bmatrix}, \qquad \mathbf{b} = \begin{bmatrix} 5 \\ 8 \end{bmatrix}.$$

Protože se jedná o dvě omezení, budeme vybírat z matice $\mathbf{A}$ vždy dva sloupce a ověřovat, zda tvoří základní řešení. Máme tedy pouze tři možnosti výběru. Označíme $\mathbf{B}_1$ matici tvořenou prvním a druhým sloupcem matice $\mathbf{A}$. Pak

$$\mathbf{B}_1 = \begin{bmatrix} 2 & 1.5 \\ 3 & 2.5 \end{bmatrix}, \qquad (\mathbf{B}_1)^{-1} = \begin{bmatrix} 5 & -3 \\ -6 & 4 \end{bmatrix},$$

Násobíme-li soustavu $\mathbf{A}\mathbf{x} = \mathbf{b}$ inverzní matici $(\mathbf{B}_1)^{-1}$, budou bazické vektory jednotkové a navíc přímo dostaneme základní řešení, to je krajní bod množiny přípustných řešení, který označíme $\mathbf{x}^{(1)}$. Zde

$$(\mathbf{B}_1)^{-1}\,\mathbf{b} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \qquad \text{a proto} \qquad \mathbf{x}^{(1)} = \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix}.$$

První a druhá složka krajního vektoru $\mathbf{x}^{(1)}$ je rovna složkám vektoru $(\mathbf{B}_1)^{-1}\,\mathbf{b}$, protože jsme za bázové vektory vybrali první dva sloupce matice $\mathbf{A}$. Ostatní složky vektoru $\mathbf{x}^{(1)}$ jsou nulové. Samozřejmě složky vektoru $(\mathbf{B}_1)^{-1}\,\mathbf{b}$ musí být nezáporné, aby krajní bod splňoval podmínku nezápornosti.

Nyní pro zjištění dalšího krajního bodu množiny $\mathcal{X}$ vybereme první a třetí sloupec matice $\mathbf{A}$ a z těchto sloupců vytvoříme matici $\mathbf{B}_2$. Pak platí

$$\mathbf{B}_2 = \begin{bmatrix} 2 & -1.5 \\ 3 & 2.2 \end{bmatrix}, \qquad (\mathbf{B}_2)^{-1} = \begin{bmatrix} -22 & 15 \\ -30 & 20 \end{bmatrix}, \qquad (\mathbf{B}_2)^{-1}\,\mathbf{b} = \begin{bmatrix} 10 \\ 10 \end{bmatrix},$$

Proto druhý krajní bod množiny $\mathcal{X}$ je $\mathbf{x}^{(2)} = \begin{bmatrix} 10 & 0 & 10 \end{bmatrix}^T$.

Nyní vybereme poslední možnou kombinaci sloupců matice $\mathbf{A}$ a sice její druhý a třetí sloupec a z nich vytvoříme matici $\mathbf{B}_3$. Potom platí

$$\mathbf{B}_3 = \begin{bmatrix} 1.5 & -1.5 \\ 2.5 & 2.2 \end{bmatrix}, \qquad (\mathbf{B}_3)^{-1} = \begin{bmatrix} -4.9 & 3.33 \\ -5.55 & 3.33 \end{bmatrix}, \qquad (\mathbf{B}_3)^{-1}\,\mathbf{b} = \begin{bmatrix} 2.22 \\ -1.11 \end{bmatrix},$$

Protože některé prvky matice $(\mathbf{B}_3)^{-1}\,\mathbf{b}$ jsou záporné, netvoří druhý a třetí sloupec matice $\mathbf{A}$ bázi.

Pro výpočet krajních paprsků vypočteme vektory

$$(\mathbf{B}_1)^{-1}\,\mathbf{a}^{(3)} = \begin{bmatrix} -0.9 \\ 0.2 \end{bmatrix}, \qquad (\mathbf{B}_2)^{-1}\,\mathbf{a}^{(2)} = \begin{bmatrix} 4.5 \\ 5 \end{bmatrix},$$

kde $\mathbf{a}^{(3)}$ je třetí sloupec matice $\mathbf{A}$, to je ten, který není v matici $\mathbf{B}_1$ a obdobně $\mathbf{a}^{(2)}$ je druhý sloupec matice $\mathbf{A}$, to je opět ten, který není v matici $\mathbf{B}_2$. Protože všechny prvky v obou vektorech nejsou nekladné, neexistuje krajní paprsek dané množiny $\mathcal{X}$. Ve vícerozměrném případě musíme pro výpočet krajních paprsků ověřit všechny vektory $\mathbf{a}^{(i)}$, které nejsou v bázi.

Množina $\mathcal{X}$ je tedy určena

$$\mathcal{X} = \left\{ \mathbf{x} \;\middle|\; \mathbf{x} = k_1 \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} + k_2 \begin{bmatrix} 10 \\ 0 \\ 10 \end{bmatrix};\;\; k_1 + k_2 = 1,\;\; k_1 \geq 0,\; k_2 \geq 0 \right\}.$$

Příklad: Nyní vyšetříme krajní body a krajní paprsky množiny $\mathcal{X} = \{\mathbf{x} \;:\; \mathbf{A}\mathbf{x} = \mathbf{b}\}$, kde

$$\mathbf{A} = \begin{bmatrix} 2 & 1.5 & -1.5 \\ 3 & 2.5 & -2.3 \end{bmatrix}, \qquad \mathbf{b} = \begin{bmatrix} 5 \\ 8 \end{bmatrix}.$$

Protože první dva sloupce matice $\mathbf{A}$ jsou stejné jako v předchozím příkladě, bude první krajní bod $\mathbf{x}^{(1)}$ stejný jako v předchozím příkladě. Snadno si ověříte, že množina přípustných řešení nemá další krajní bod.

Pro určení krajního paprsku vypočteme vektor

$$(\mathbf{B}_1)^{-1}\,\mathbf{a}^{(3)} = \begin{bmatrix} -0.6 \\ -0.2 \end{bmatrix}.$$

Protože složky tohoto vektoru jsou nekladné, existuje krajní paprsek $\mathbf{y}$, jehož první dvě složky jsou rovny záporně vzatým složkám vektoru $(\mathbf{B}_1)^{-1}\,\mathbf{a}^{(3)}$ a třetí složka krajního vektoru je rovna jedné (ostatní složky, kdyby existovaly, by byly nulové). Proto reprezentant krajního paprsku je vektor $\mathbf{y} = \begin{bmatrix} 0.6 & 0.2 & 1 \end{bmatrix}^T$. Množina přípustných řešení $\mathcal{X}$ je v tomto případě určena

$$\mathcal{X} = \left\{ \mathbf{x} \;\middle|\; \mathbf{x} = \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} + \alpha \begin{bmatrix} 0.6 \\ 0.2 \\ 1 \end{bmatrix}; \quad \alpha \geq 0, \right\}.$$

4.7 Maticový zápis simplexové metody

Abychom odvodili metodu nalezení primární i duální úlohy lineárního programování, provedeme maticový zápis řešení úlohy lineárního programování. Mějme tedy primární úlohu, kterou je standardní úloha lineárního programování

$$\max \{ f(\mathbf{x}) = \mathbf{c}^T\mathbf{x} \;:\; \mathbf{A}\mathbf{x} \leq \mathbf{b},\; \mathbf{x} \geq \mathbf{0} \} \tag{4.8}$$

Její kanonický tvar je

$$\max \{ f(\mathbf{x}) = \mathbf{c}^T\mathbf{x} \;:\; \mathbf{A}\mathbf{x} + \mathbf{I}\mathbf{u} = \mathbf{b},\; \mathbf{x} \geq \mathbf{0},\; \mathbf{u} \geq \mathbf{0} \} \tag{4.9}$$

Základní řešení je tvaru - viz simplexová tabulka

$$\begin{array}{rrrrrrr} & \mathbf{A}_{11}\mathbf{x}_1 & +\mathbf{A}_{12}\mathbf{x}_2 & +\mathbf{I}\mathbf{u}_1 & & = & \mathbf{b}_1 \\ & \mathbf{A}_{21}\mathbf{x}_1 & +\mathbf{A}_{22}\mathbf{x}_2 & & +\mathbf{I}\mathbf{u}_2 & = & \mathbf{b}_2 \\ f(\mathbf{x}) & -\mathbf{c}_1^T\mathbf{x}_1 & -\mathbf{c}_2^T\mathbf{x}_2 & & & = & \mathbf{0} \end{array} \tag{4.10}$$

kde $\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix}$, $\mathbf{x} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix}$, $\mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix}$, $\mathbf{c} = \begin{bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \end{bmatrix}$, $\mathbf{u} = \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \end{bmatrix}$ jsou dělené matice a vektory. Dělení matic a vektorů je provedeno tak, abychom jedním krokem získali přímo optimální řešení. Bázické proměnné jsou zřejmě $\mathbf{u}_1$, $\mathbf{u}_2$ a přípustné řešení je tedy $\mathbf{u}_1 = \mathbf{b}_1$, $\mathbf{u}_2 = \mathbf{b}_2$, $\mathbf{x}_1 = \mathbf{0}$, $\mathbf{x}_2 = \mathbf{0}$ a potom kritérium je rovno $f(\mathbf{x}) = \mathbf{0}$.

Nechť nové bázické proměnné jsou $\mathbf{x}_2$, $\mathbf{u}_2$, místo bázové proměnné $\mathbf{u}_1$ zavádíme tedy novou bázovou proměnnou $\mathbf{x}_2$. Z první rovnice omezení (4.10) plyne

$$\mathbf{x}_2 = \mathbf{A}_{12}^{-1}\left(\mathbf{b}_1 - \mathbf{I}\mathbf{u}_1 - \mathbf{A}_{11}\mathbf{x}_1\right). \tag{4.11}$$

Rovnice omezení upravíme do tvaru

$$\begin{bmatrix} \mathbf{A}_{11} & \mathbf{I} \\ \mathbf{A}_{21} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{u}_1 \end{bmatrix} + \begin{bmatrix} \mathbf{A}_{12} & \mathbf{0} \\ \mathbf{A}_{22} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x}_2 \\ \mathbf{u}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix}, \tag{4.12}$$

Zavedeme matici nových bázických vektorů, kterou označíme $\mathbf{B}$. Platí

$$\mathbf{B} = \begin{bmatrix} \mathbf{A}_{12} & \mathbf{0} \\ \mathbf{A}_{22} & \mathbf{I} \end{bmatrix}, \qquad \mathbf{B}^{-1} = \begin{bmatrix} \mathbf{A}_{12}^{-1} & \mathbf{0} \\ -\mathbf{A}_{22}\mathbf{A}_{12}^{-1} & \mathbf{I} \end{bmatrix}.$$

Násobíme-li předchozí soustavu (4.12) zleva maticí $\mathbf{B}^{-1}$ dostaneme

$$\begin{bmatrix} \mathbf{A}_{12}^{-1} & \mathbf{0} \\ -\mathbf{A}_{22}\mathbf{A}_{12}^{-1} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{A}_{11} & \mathbf{I} \\ \mathbf{A}_{21} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{u}_1 \end{bmatrix} + \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x}_2 \\ \mathbf{u}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{12}^{-1}\mathbf{b}_1 \\ -\mathbf{A}_{22}\mathbf{A}_{12}^{-1}\mathbf{b}_1 + \mathbf{b}_2 \end{bmatrix} \tag{4.13}$$

Rovnice pro kritérium (4.10c) se po dosazení (4.11) změní na

$$f(\mathbf{x}) - \mathbf{c}_1^T\mathbf{x}_1 - \mathbf{c}_2^T\left[\mathbf{A}_{12}^{-1}\left(\mathbf{b}_1 - \mathbf{I}\mathbf{u}_1 - \mathbf{A}_{11}\mathbf{x}_1\right)\right] = \mathbf{0}$$

čili po úpravě

$$f(\mathbf{x}) - \left(\mathbf{c}_1^T - \mathbf{c}_2^T\mathbf{A}_{12}^{-1}\mathbf{A}_{11}\right)\mathbf{x}_1 + \left(\mathbf{c}_2^T\mathbf{A}_{12}^{-1}\right)\mathbf{u}_1 = \mathbf{c}_2^T\mathbf{A}_{12}^{-1}\mathbf{b}_1. \tag{4.14}$$

Rovnice (4.13) a (4.14) zapíšeme společně ve tvaru

$$\begin{array}{rcccc} & \mathbf{A}_{12}^{-1}\mathbf{A}_{11}\mathbf{x}_1 + \mathbf{A}_{12}^{-1}\mathbf{u}_1 & +\mathbf{I}\mathbf{x}_2 & = & \mathbf{A}_{12}^{-1}\mathbf{b}_1 \\ & \left(-\mathbf{A}_{22}\mathbf{A}_{12}^{-1}\mathbf{A}_{11} + \mathbf{A}_{21}\right)\mathbf{x}_1 + \mathbf{A}_{22}\mathbf{A}_{12}^{-1}\mathbf{u}_1 & +\mathbf{I}\mathbf{u}_2 & = & \mathbf{b}_2 - \mathbf{A}_{22}\mathbf{A}_{12}^{-1}\mathbf{b}_1 \\ f(\mathbf{x})+ & \left(-\mathbf{c}_1^T + \mathbf{c}_2^T\mathbf{A}_{12}^{-1}\mathbf{A}_{11}\right)\mathbf{x}_1 + \mathbf{c}_2^T\mathbf{A}_{12}^{-1}\mathbf{u}_1 & & = & \mathbf{c}_2^T\mathbf{A}_{12}^{-1}\mathbf{b}_1. \end{array} \tag{4.15}$$

Bázové proměnné jsou nyní $\mathbf{x}_2$ a $\mathbf{u}_2$. Přípustné řešení

$$\begin{aligned} \mathbf{x}_2^* &= \mathbf{A}_{12}^{-1}\mathbf{b}_1 \geq \mathbf{0} && (4.16) \\ \mathbf{u}_2^* &= \mathbf{b}_2 - \mathbf{A}_{22}\mathbf{A}_{12}^{-1}\mathbf{b}_1 \geq \mathbf{0} \\ \mathbf{x}_1^* &= \mathbf{0}, \qquad \mathbf{u}_1^* = \mathbf{0} \\ f(\mathbf{x}^*) &= \mathbf{c}_2^T\mathbf{A}_{12}^{-1}\mathbf{b}_1 && (4.17) \end{aligned}$$

je optimální řešení, pokud jsou nezáporné koeficienty v posledním řádku (v kritériu), čili

$$\begin{aligned} -\mathbf{c}_1^T + \mathbf{c}_2^T\mathbf{A}_{12}^{-1}\mathbf{A}_{11} &\geq \mathbf{0} && (4.18) \\ \mathbf{c}_2^T\mathbf{A}_{12}^{-1} &\geq \mathbf{0} \end{aligned}$$

Z posledního řádku simplexové tabulky (4.15) můžeme přímo získat řešení duální úlohy. Poslední řádek simplexové tabulky je totiž roven

$$f(\mathbf{x}) + \mathbf{v}_1^T\mathbf{x}_1 + \mathbf{v}_2^T\mathbf{x}_2 + \boldsymbol{\lambda}_1^T\mathbf{u}_1 + \boldsymbol{\lambda}_2^T\mathbf{u}_2 = f^*(\mathbf{x}) \tag{4.19}$$

kde $\boldsymbol{\lambda} = \begin{bmatrix} \boldsymbol{\lambda}_1 \\ \boldsymbol{\lambda}_2 \end{bmatrix}$ je řešení duální úlohy. Porovnáním s (4.15) tedy platí $\boldsymbol{\lambda}_1^T = \mathbf{c}_2^T\mathbf{A}_{12}^{-1}, \boldsymbol{\lambda}_2^T = \mathbf{0}$, neboli

$$\boldsymbol{\lambda}_1 = \left(\mathbf{A}_{12}^T\right)^{-1}\mathbf{c}_2, \qquad \boldsymbol{\lambda}_2 = \mathbf{0} \tag{4.20}$$

Předchozí tvrzení snadno prokážeme, budeme-li stejným postupem řešit duální úlohu lineárního programování. Duální úlohu $\min\{\mathbf{b}^T\boldsymbol{\lambda} \ : \ \mathbf{A}^T\boldsymbol{\lambda} \geq \mathbf{c} \ , \ \boldsymbol{\lambda} \geq \mathbf{0}\}$, upravíme do kanonického tvaru

$$-\max\{f(\boldsymbol{\lambda}) = -\mathbf{b}^T\boldsymbol{\lambda} \ : \ -\mathbf{A}^T\boldsymbol{\lambda} + \mathbf{I}\mathbf{v} = -\mathbf{c} \ , \ \boldsymbol{\lambda} \geq \mathbf{0} \ , \ \mathbf{v} \geq \mathbf{0}\} \tag{4.21}$$

Základní řešení opět zapíšeme ve tvaru

$$\begin{array}{rcccccc} & -\mathbf{A}_{11}^T\boldsymbol{\lambda}_1 & -\mathbf{A}_{21}^T\boldsymbol{\lambda}_2 & +\mathbf{I}\mathbf{v}_1 & & = & -\mathbf{c}_1 \\ & -\mathbf{A}_{12}^T\boldsymbol{\lambda}_1 & -\mathbf{A}_{22}^T\boldsymbol{\lambda}_2 & & +\mathbf{I}\mathbf{v}_2 & = & -\mathbf{c}_2 \\ -f(\boldsymbol{\lambda}) & +\mathbf{b}_1^T\boldsymbol{\lambda}_1 & +\mathbf{b}_2^T\boldsymbol{\lambda}_2 & & & = & \mathbf{0} \end{array} \tag{4.22}$$

Bázické proměnné jsou tedy $\mathbf{v}_1$, $\mathbf{v}_2$ a přípustné řešení je tedy $\mathbf{v}_1 = -\mathbf{c}_1$, $\mathbf{v}_2 = -\mathbf{c}_2$, $\boldsymbol{\lambda}_1 = \mathbf{0}$, $\boldsymbol{\lambda}_2 = \mathbf{0}$ a potom kritérium je rovno $f(\boldsymbol{\lambda}) = \mathbf{0}$.

Nechť nové bázické proměnné jsou $\boldsymbol{\lambda}_1$, $\mathbf{v}_1$, místo bázové proměnné $\mathbf{v}_2$ zavádíme tedy novou bázovou proměnnou $\boldsymbol{\lambda}_1$. Z druhé rovnice omezení (4.22b) plyne

$$\boldsymbol{\lambda}_1 = \left(\mathbf{A}_{12}^T\right)^{-1}\left(\mathbf{c}_2 + \mathbf{I}\mathbf{v}_2 - \mathbf{A}_{22}^T\boldsymbol{\lambda}_2\right). \tag{4.23}$$

Rovnice omezení upravíme do tvaru

$$\begin{bmatrix} -\mathbf{A}_{21}^T & \mathbf{0} \\ -\mathbf{A}_{22}^T & \mathbf{I} \end{bmatrix}\begin{bmatrix} \boldsymbol{\lambda}_2 \\ \mathbf{v}_2 \end{bmatrix} + \begin{bmatrix} -\mathbf{A}_{11}^T & \mathbf{I} \\ -\mathbf{A}_{12}^T & \mathbf{0} \end{bmatrix}\begin{bmatrix} \boldsymbol{\lambda}_1 \\ \mathbf{v}_1 \end{bmatrix} = \begin{bmatrix} -\mathbf{c}_1 \\ -\mathbf{c}_2 \end{bmatrix}, \tag{4.24}$$

Zavedeme matici nových bázických vektorů, kterou označíme $\overline{\boldsymbol{B}}$. Platí

$$\overline{\boldsymbol{B}} = \begin{bmatrix} -\mathbf{A}_{11}^T & \mathbf{I} \\ -\mathbf{A}_{12}^T & \mathbf{0} \end{bmatrix}, \qquad \overline{\boldsymbol{B}}^{-1} = \begin{bmatrix} \mathbf{0} & -\left(\mathbf{A}_{12}^T\right)^{-1} \\ \mathbf{I} & -\mathbf{A}_{11}^T\left(\mathbf{A}_{12}^T\right)^{-1} \end{bmatrix}.$$

Opět násobíme předchozí soustavu (4.24) zleva maticí $\overline{\boldsymbol{B}}^{-1}$. Po úpravě rovnice (4.24) a po dosazení (4.23) do kritéria (4.22c) dostaneme

$$\begin{array}{rcccc} \mathbf{A}_{12}^{-T}\mathbf{A}_{22}\boldsymbol{\lambda}_2 - \mathbf{A}_{12}^{-T}\mathbf{v}_2 & & +\mathbf{I}\boldsymbol{\lambda}_1 & = & \mathbf{A}_{12}^{-T}\mathbf{c}_2 \\ \left(\mathbf{A}_{11}^T\mathbf{A}_{12}^{-T}\mathbf{A}_{22} - \mathbf{A}_{12}^T\right)\boldsymbol{\lambda}_2 - \mathbf{A}_{11}^T\mathbf{A}_{12}^{-T}\mathbf{v}_2 & & +\mathbf{I}\mathbf{v}_1 & = & -\mathbf{c}_1 + \mathbf{A}_{11}^T\mathbf{A}_{12}^{-T}\mathbf{c}_2 \\ -f(\mathbf{x}) + \left(\mathbf{b}_2^T - \mathbf{b}_1^T\mathbf{A}_{12}^{-T}\mathbf{A}_{22}^T\right)\boldsymbol{\lambda}_2 + \mathbf{b}_1^T\mathbf{A}_{12}^{-T}\mathbf{v}_2 & & & = & -\mathbf{b}_1^T\mathbf{A}_{12}^{-T}\mathbf{c}_2. \end{array} \tag{4.25}$$

Přípustné řešení duální úlohy

$$\begin{aligned} \boldsymbol{\lambda}_1^* &= \mathbf{A}_{12}^{-T}\mathbf{c}_2 \geq \mathbf{0} && (4.26) \\ \mathbf{v}_1^* &= -\mathbf{c}_1 + \mathbf{A}_{11}^T\mathbf{A}_{12}^{-T}\mathbf{c}_2 \geq \mathbf{0} \\ \boldsymbol{\lambda}_2^* &= \mathbf{0}, \qquad \mathbf{v}_2^* = \mathbf{0} \\ f(\boldsymbol{\lambda}^*) &= \mathbf{b}_1^T\mathbf{A}_{12}^{-T}\mathbf{c}_2 = \mathbf{c}_2^T\mathbf{A}_{12}^{-1}\mathbf{b}_1 && (4.27) \end{aligned}$$

je optimální řešení, pokud jsou nezáporné koeficienty v posledním řádku (v kritériu), čili

$$\begin{aligned} -\mathbf{b}_2 - \mathbf{A}_{22}\mathbf{A}_{12}^{-1}\mathbf{b}_1 &\geq \mathbf{0} \\ \mathbf{A}_{12}^{-1}\mathbf{b}_1 &\geq \mathbf{0} \end{aligned}$$

Skutečně tedy platí, že řešení (4.26) duální úlohy je totožné se vztahem (4.20). Řešením primární úlohy (ve tvaru zde uvedeném) dostaneme tedy nejen primární, ale i duální řešení.

4.8 Speciální případy

U většiny praktických příkladů je počet iterací simplexového algoritmu roven (1.5 až 3)m, kde m je počet omezení.

V předchozím odstavci jsme popsali simplexovou metodu v nejzákladnějším tvaru. Jednalo se o normální úlohu (koeficienty $b_i > 0$) a omezení ve tvaru nerovnosti, které jsme pomocí přídavných proměnných převedli na omezení ve tvaru rovnosti. Tím jsme přímo získali základní řešení, které je nezbytné pro start simplexového algoritmu. Dále úloha měla jediné a omezené řešení.

4.8.1 Alternativní optimální řešení

V tomto případě množina optimálních řešení není tvořena jedním bodem, ale je to konvexní polyedrická množina. Proto je třeba nalézt všechny její krajní body (a případně i všechny její krajní paprsky).

Krajní body množiny optimálních řešení nalezneme následujícím postupem. Pokud v posledním řádku (který reprezentuje kritérium) jsou všechny koeficienty nezáporné, ale některé jsou nulové i u nebázových proměnných, existují **alternativní optimální řešení**.

Jedno alternativní optimální řešení jsme tedy nalezli a označíme jej $\mathbf{x}_1^*$ (to je jeden krajní bod množiny optimálních řešení). Je-li tedy v posledním řádku nula u j-té nebázové proměnné, pak ji známým způsobem zahrneme do báze a získáme jiné optimální řešení, které označíme $\mathbf{x}_2^*$. To provedeme pro všechny sloupce j, u kterých je nula u nebázových proměnných. Tím získáme všechny krajní body $\mathbf{x}_i^*$ množiny optimálních řešení. Jejich konvexní kombinace určuje množinu optimálních alternativních řešení

$$\mathbf{x}^* = \sum_{\forall i} k_i \mathbf{x}_i^*, \qquad \sum k_i = 1, \qquad k_i \geq 0. \tag{4.28}$$

Příklad: Řešte úlohu lineárního programování

$$\max \ \{2\ x_1\ +\ 4\ x_2\ \ :\ x_1 + x_2 \leq 4,\ x_1 + 2x_2 \leq 6,\ x_i \geq 0\}$$

Sestavíme simplexovou tabulku a provedeme první iteraci známým způsobem

x_1	x_2	x_3	x_4	**b**
1	1	1	0	4
1	2	0	1	6
-2	-4	0	0	0
0.5	0	1	-0.5	1
0.5	1	0	0.5	3
0	0	0	2	12

Ze simplexové tabulky určíme jedno řešení $\mathbf{x}_1^* = [0,\ 3]^T$. Protože v simplexové tabulce existuje v posledním řádku nulový prvek i u nebázové proměnné (je to první prvek v

posledním řádku), zahrneme příslušnou proměnnou do báze a určíme druhé řešení. To je uvedeno v následující tabulce

x_1	x_2	x_3	x_4	**b**
1	0	2	-1	2
0	1	-1	1	2
0	0	0	2	12

Druhé řešení (druhý krajní bod množiny optimálních řešení) je rovno $\mathbf{x}_2^* = [2,\ 2]^T$. Množina optimálních řešení je rovna

$$\mathbf{x}^* = \lambda \mathbf{x}_1^* + (1-\lambda)\mathbf{x}_2^* = [(1-\lambda)2\ ;\ 3\lambda + 2(1-\lambda)]^T\ ; \quad \lambda \in (0;1)$$

4.8.2 Neomezená řešení

Je-li některý koeficient v posledním řádku simplexové tabulky záporný, značí to, že jeho zvětšení (z nuly na nenulovou hodnotu) povede ke zlepšení (zvětšení) kritéria. Pokud ale v jemu odpovídajícím sloupci jsou všechny koeficienty $a_{i,j}$ záporné, znamená to, že můžeme odpovídající proměnnou zvětšovat nade všechny meze. Kritérium neomezeně roste a přitom jsou všechna omezení splněna. Tento jev ilustrujeme v následujícím příkladě.

Příklad : Mějme následující úlohu lineárního programování

$$\max\ \{x_1 + 3x_2\ :\ -2x_1 + x_2 \leq 4;\ x_i \geq 0\ \}$$

Zavedeme jednu přídatnou proměnnou y a sestavíme simplexovou tabulku

x_1	x_2	y	**b**
-2	1	1	4
-1	**-3**	0	0

Opět jsme tučně označili nejzápornější prvek v posledním řádku, který nám určuje klíčový sloupec (proměnnou, která způsobí největší růst kritéria). V rámečku je klíčový prvek, který určuje, která proměnná bude nebázová. Po standardních úpravách dostaneme simplexovou tabulku ve tvaru

x_1	x_2	y	**b**
-2	1	1	4
-7	0	3	12

V posledním řádku simplexové tabulky je záporný prvek (-7) ve sloupci odpovídajícím proměnné x_1. Protože všechny prvky v příslušném sloupci jsou záporné (zde pouze jeden prvek -2) je zřejmé, že proměnná x_1 může růst nade všechny meze, omezení jsou splněna a kritérium stále roste. Řešení této úlohy není tedy omezené.

4.8.3 Jiná omezení a jejich převod na kanonický tvar

Pokud jsou omezení v úloze lineárního programování ve tvaru $\mathbf{Ax} \leq \mathbf{b}$, kde $\mathbf{b} \geq 0$, pak zavedením přídatné proměnné $\mathbf{y} \geq 0$ upravíme omezení do tvaru $\mathbf{Ax} + \mathbf{y} = \mathbf{b}$. Protože se přídatná proměnná $\mathbf{y}$ nevyskytuje v kritériu, pak získáme přímo základní řešení ve tvaru $\mathbf{y} = \mathbf{b}$.

Pokud jsou některá omezení ve tvaru rovnosti $\mathbf{Ax} = \mathbf{b}$, případně neplatí, že $\mathbf{b} \geq 0$, nelze nalézt přímo základní řešení. Proto je třeba použít následující postup řešící tento případ.

Nelze-li nalézt přímo základní řešení, pak musíme upravit omezení do tvaru $\mathbf{Ax} = \mathbf{b}$, kde $\mathbf{b} \geq 0$. To vždy je možné, potom ale ve vektoru $\mathbf{x}$ mohou být i nové přídatné proměnné.

Nyní uvažujeme **umělý problém**

$$\max \{ -\sum y_i \ : \ \mathbf{Ax} + \mathbf{y} = \mathbf{b}, \ \mathbf{x} \geq \mathbf{0} \ , \ \mathbf{y} \geq 0 \ \} \tag{4.29}$$

kde $\mathbf{y}$ jsou umělé proměnné. Základní řešení pro tuto úlohu je zřejmě $\mathbf{y} = \mathbf{b}$.

□

Simplexovou metodou řešíme tuto úlohu. Přitom nejprve musíme z kritéria vyloučit proměnné y_i. To snadno uděláme, neboť platí $y_i = b_i - \sum_j a_{i,j} x_j$. Řešení, pokud existuje, je zřejmě max $f(.) = 0$, pro $\mathbf{y} = 0$. Přitom nalezneme základní bázi pro původní úlohu. Je-li ve výsledku umělé úlohy některé y_i nenulové, pak původní úloha nemá řešení.

Řešení původní úlohy se tak rozpadá do dvou fází. V první fázi řešíme umělou úlohu zavedením umělé proměnné $\mathbf{y}$ a jiného kritéria. Pokud dostaneme řešení umělé úlohy $\mathbf{y} = 0$, pokračujeme druhou fází, ve které vypustíme umělé proměnné y_i a vrátíme se k původnímu kritériu a simplexovou metodou řešíme původní úlohu. Uvedený postup ilustrujeme v následujícím příkladě.

Příklad: Nalezněte řešení jednoduché úlohy

$$\max \{ -3x_1 - 2x_2 \ : \ x_1 + x_2 = 10; \quad x_1 \geq 4; \quad x_i \ \geq 0 \}$$

Omezení si upravíme do tvaru

$$\begin{aligned} x_1 + x_2 &= 10 \\ x_1 - x_3 &= 4, \qquad x_i \geq 0 \end{aligned}$$

Protože není na první pohled zřejmé základní řešení, budeme řešit umělý problém

$$\max \{ -y_1 - y_2 \ : \ \mathbf{Ax} + \mathbf{y} = \mathbf{b}; \ \mathbf{x} \geq 0; \ \mathbf{y} \geq 0, \ \}$$

kde matice $\mathbf{A}$ a vektor $\mathbf{b}$ plynou z předchozí soustavy. Simplexová tabulka pro umělý problém je následující

x_1	x_2	x_3	y_1	y_2	$\mathbf{b}$
1	1	0	1	0	10
1	0	-1	0	1	4
0	0	0	1	1	0

Nyní musíme vynulovat koeficienty u proměnných y_i v kritériu (v posledním řádku předchozí tabulky). Potom dostaneme simplexovou tabulku ve tvaru

x_1	x_2	x_3	y_1	y_2	**b**
1	1	0	1	0	10
1	0	-1	0	1	4
-2	-1	1	0	0	-14

Odtud plyne základní řešení $x_1 = x_2 = x_3 = 0$, $y_1 = 10$, $y_2 = 4$. Kritérium je rovno -14. Řešení není optimální, neboť poslední řádek (kritérium) obsahuje záporné prvky. V předchozí tabulce je tučně vyznačen prvek, který určuje klíčový sloupec a v rámečku je klíčový prvek. Standardním postupem upravíme simplexovou tabulku tak, aby proměnná x_1 byla v bázi a naopak proměnná y_2 z báze vypadla. Dostaneme tedy novou simplexovou tabulku pro další krok řešení umělé úlohy

x_1	x_2	x_3	y_1	y_2	**b**
0	1	1	1	-1	6
1	0	-1	0	1	4
0	**-1**	-1	0	2	-6

Základní řešení je zřejmě $y_1 = 6$, $x_1 = 4$. Řešení$f(.) = -6$ opět není optimální, neboť v posledním řádku simplexové tabulky nejsou všechny prvky nezáporné. Opět je tučně vyznačen klíčový prvek, který nám říká, že proměnná x_2 bude v nové bázi a naopak proměnná y_1 z báze vypadne. Po úpravách dostaneme konečně řešení umělé úlohy, jak je patrné z následující simplexové tabulky

x_1	x_2	x_3	y_1	y_2	**b**
0	1	1	1	-1	6
1	0	-1	0	1	4
0	0	0	1	1	0

Z předchozí tabulky plyne konečně řešení umělé úlohy $y_1 = 0$, $y_2 = 0$, $x_1 = 4$, $x_2 = 6$, $x_3 = 0$. Hodnota kritéria je $f(.) = 0$. Umělé proměnné jsou nulové, kritérium je nulové také, z čehož plyne, že původní úloha má řešení.

Nyní jsme dostali základní bázi $x_1 = 4$, $x_2 = 6$, $x_3 = 0$ pro naši původní úlohu. Sestavíme tedy novou simplexovou tabulku pro původní úlohu tak, že vypustíme z předchozí simplexové tabulky sloupce příslušející uměle zavedeným proměnným y_i a do posledního řádku simplexové tabulky napíšeme koeficienty původního kritéria (záporně vzaté). Nová simplexová tabulka má nyní tvar

x_1	x_2	x_3	**b**
0	1	1	6
1	0	-1	4
3	2	0	0

Dále je třeba vynulovat prvky v posledním řádku příslušejícím bázovým proměnným x_1 a x_2. Po provedení této úpravy dostaneme simplexovou tabulku ve tvaru

x_1	x_2	x_3	**b**
0	1	1	6
1	0	-1	4
0	0	1	-24

Protože v posledním řádku simplexové tabulky jsou nezáporné prvky, nalezli jsme konečně řešení naší původní úlohy. Řešení je zřejmě $x_1 = 4$, $x_2 = 6$ a optimální hodnota kritéria je $f(\mathbf{x}) = -24$.

4.9 Příklady

1. Nalezněte pomocí simplexového algoritmu všechny krajní body (vrcholy) konvexní polyedrické množiny určené nerovnostmi

$$-x_1 + x_2 \leq 2;\ x_1 + 2x_2 \leq 5;\ x_1 - 3x_2 \leq 1;\ x_i \geq 0$$

2. Určete, zda existuje vektor $\mathbf{x}$, který splňuje omezení ve tvaru $\mathbf{Ax} = \mathbf{b}$, případně ve tvaru $\mathbf{Ax} \leq \mathbf{b}$, pro kladné, záporné, případně nulové prvky vektoru $\mathbf{b}$. Rozmyslete si podrobně tento problém.

3. Je dána soustava rovnic $\mathbf{Ax} = \mathbf{b}$, dim $\mathbf{A} = m \times n$, kde m, n jsou libovolné. Jakou musí mít hodnost matice $\mathbf{A}$, aby množina $\mathbf{X} = \{\mathbf{x} : \mathbf{Ax} = \mathbf{b}\}$, byla a) neprázdou množinou, b) jednoprvkovou množinou, c) konvexní množinou. Jak nalezneme její krajní body a krajní paprsky.

4. Sestavte program v jazyce Matlab pro řešení úlohy lineárního programování simplexovým algoritmem. Program musí nalézt primární i duální řešení, neomezené řešení i alternativní řešení.

5. Jak by se změnil simplexový algoritmus, kdybychom nekladli omezení na nezápornost proměnných? Proveďte úpravu.

Kapitola 5

Úvod do teorie her

Teorie her se zabývá řešením matematických modelů konfliktních situací. Podle toho, zda se jedná o reálnou rozhodovací situaci nebo její model - hru, či o obecnou optimalizační úlohu, používáme různé názvosloví, které je uvedeno v následující tabulce.

Reálná rozhodovací situace	Matematický model této situace - hra	Teorie optimálního řízení
rozhodovací situace	hra v normálním tvaru	optimalizační problém
účastník	hráč	řídicí veličina (řízení)
rozhodnutí	strategie	hodnoty řídicích veličin
množina rozhodnutí	prostor strategií	množina přípustných hodnot říd. veličin
důsledky rozhodnutí	výplatní funkce	kritérium jakosti řízení
důsledek	výhra	hodnota kritéria

Hra v normálním tvaru je definována množinou

$$\{Q,\ X_1, \ldots, X_n,\ J_1(x_1, \ldots, x_n), \ldots, J_n(x_1, \ldots, x_n)\ \} \tag{5.1}$$

kde $Q = \{1,\ 2, \ldots, n\}$ jsou hráči, množiny X_1 ažX_n jsou množiny strategií hráčů 1 až n; $x_1 \in X_1$ až $x_x \in X_n$ jsou strategie hráčů a $J_i(x_1, \ldots, x_n)$ jsou výhry hráče i.

Konečná hra je taková hra, v níž prostory strategií tvoří konečnou množinu. Hra s konstantním součtem je taková hra, v níž při všech strategiích hráčů je součet výher všech hráčů konstantní. Platí tedy

$$\sum_{i=1}^{n} J_i(x_1, \ldots, x_n) = K, \quad \forall\ x_i \in X_i \tag{5.2}$$

Součet K není ve hře s konstantním součtem závislý na strategiích. Hra s nulovým součtem je při $K = 0$.

Ve hrách existují problémy dvojího druhu:

- Jak by se v rozhodovací situaci choval průměrný jedinec - tzv. deskriptivní hledisko.

- Jaké je v dané situaci objektivně nejlepší rozhodnutí - tzv. normativní hledisko.

Teoreticky zajímavé je normativní hledisko. Deskriptivní hledisko někdy nelze pominout, například při rozhodování při riziku - viz dále. Rozdíl mezi normativním a deskriptivním hlediskem vyplyne názorně z tohoto příkladu. Při rozboru konfliktní situace - střetnutí dvou nepřátelských letounů, bychom mohli při hodnocení výsledků souboje oceňovat stejně situace, že a) obě letadla se navzájem zničí a b) ani jeden letoun není poškozen. To bychom mohli označit za normativní hledisko. Zcela jiné, deskriptivní hledisko, by patrně měl ten, kdo v letadle sedí.

5.1 Antagonistický konflikt

5.1.1 Hry s konstantním součtem

Budeme se zabývat antagonistickým konfliktem dvou účastníků. Matematickým modelem tohoto antagonistického konfliktu je hra dvou hráčů v normálním tvaru s konstantním součtem.

Strategii prvního hráče místo x_1 budeme dále značit u a strategii druhého hráče budeme místo x_2 značit v. Modelem antagonistického konfliktu je tedy hra definovaná množinami

$$\{Q = \{1,\ 2\}\ ;\ U,\ V\ ;\ J_1(u,v),\ J_2(u,v)\} \tag{5.3}$$

Je rozumné definovat **optimální strategii** jako takovou strategii, od níž žádná odchylka nemůže přinést hráči výhody, za předpokladu, že druhý hráč zachová svoji optimální strategii. Optimální strategii prvního hráče označíme $u^* \in U$ a optimální strategii druhého hráče označíme $v^* \in V$, potom tedy platí

$$\begin{aligned} J_1(u,v^*) &\leq J_1(u^*,v^*), \quad \forall\ u \in U,\ v \in V \\ J_2(u,v^*) &\leq J_2(u^*,v^*). \end{aligned} \tag{5.4}$$

Ve hře s nulovým součtem je $J_2 = -J_1$ a proto stačí uvažovat pouze jedinou výplatní funkci $J = J_1$. Potom hra s nulovým součtem je určena množinami $\{Q = \{1,2\}; U, V; J(u,v)\}$. Optimální strategie u^*, v^* prvního a druhého hráče v této hře jsou strategie, pro které platí

$$J(u,v^*)\ \leq\ J(u^*,v^*)\ \leq\ J(u^*,v). \tag{5.5}$$

Je zřejmé, že první hráč se strategiemi u se snaží maximalizovat výplatní funkci, zatímco druhý hráč se volbou strategií v snaží výplatní funkci minimalizovat, protože výhra prvního hráče je rovna ztrátě druhého hráče.

Dále je zřejmé, že i ve hře s konstantním součtem lze použít jedinou výplatní funkci, neboť $J_2 = K - J_1$. Optimální strategie se nezmění, přičteme-li k výplatní funkci J_1 libovolnou konstantu nezávislou na strategiích a podobně pro J_2. Proto každou hru s konstantním součtem lze přeměnit na hru s nulovým součtem, ve které výplatní funkce je $J = J_1 - J_2$, nebo také pouze $J = J_1$.

Změna hry s konstantním součtem na hru s nulovým součtem má názornou interpretaci. V konfliktu, v němž se hráči dělí o pevně danou částku (hra s konstantním

součtem), je lhostejné, snaží-li se hráč získat z této částky pro sebe co nejvíce, nebo se snaží první hráč maximalizovat rozdíl svého zisku a zisku protihráče a druhý hráč se tento rozdíl snaží minimalizovat. Případně se první hráč snaží svůj zisk maximalizovat a druhý hráč se snaží minimalizovat zisk prvního hráče (pak na něj z konstantní částky zbude nejvíce).

5.1.2 Maticové hry

Konečný antagonistický konflikt dvou hráčů popisujeme **maticovou hrou**. V ní je počet strategií obou hráčů konečný. Konečný počet strategií $u \in U$ prvního hráče očíslujeme přirozenými čísly 1 až m. Podobně konečný počet strategií druhého hráče $v \in V$ očíslujeme přirozenými čísly 1 až n. Potom konečná hra s nulovým součtem je **maticová hra**

$$\{Q = \{1, 2\}; U = \{1, \ldots, m\}, V = \{1, \ldots, n\}; J(i, j) = a_{i,j}; i \in U; j \in V\}, \tag{5.6}$$

kde matice $\mathbf{A}$ rozměru $m \times n$ s prvky $a_{i,j}$ je matice hry.

Poznámka: První hráč vybírá řádek i v matici hry a druhý hráč vybírá sloupec j v matici hry. První hráč chce maximalizovat a druhý hráč minimalizovat prvek $a_{i,j}$ v matici hry. □

Situace je úplně jasná, existuje-li v matici hry $\mathbf{A}$ prvek, který je současně nejmenší na řádku a největší ve sloupci. Uvedeme si triviální příklad, který názorně ilustruje naše úvahy. Mějme matici hry $\mathbf{A}$ ve tvaru

$$\begin{array}{rcc} & & \varphi_i \\ \mathbf{A} = & \begin{bmatrix} 2 & 3 & 4 \\ 3 & 4 & 4 \\ 2 & 1 & 6 \end{bmatrix} & \begin{matrix} (2) \\ (3) \\ (1) \end{matrix} \\ \psi_j & \begin{matrix} (3) & (4) & (6) \end{matrix} & \end{array}$$

První hráč, který volí řádek i matice $\mathbf{A}$, chce maximalizovat a proto se podívá po jednotlivých řádcích a v nich nalezne nejmenší prvek (pro něho nejhorší případ). Minimální hodnoty $\min_j a_{ij} = \varphi_i$ jsou v závorce připsány k jednotlivým řádkům matice $\mathbf{A}$. První hráč vybere pochopitelně ten řádek i^*, v němž je minimální prvek největší $\varphi_{i^*} = \max_i \varphi_i$. V našem případě je to druhý řádek, tedy $i^* = 2$ a $\varphi_{i^*} = 3$.

Druhý hráč, který volí sloupce j matice $\mathbf{A}$, chce minimalizovat a proto se podívá po jednotlivých sloupcích matice $\mathbf{A}$ a v nich nalezne největší prvek (pro něho nejhorší případ). Maximální hodnoty $\max_i a_{ij} = \psi_j$ jsou v závorce připsány k jednotlivým sloupcům matice $\mathbf{A}$. Druhý hráč vybere pochopitelně ten sloupce j^*, v němž je maximální prvek nejmenší $\psi_{j^*} = \min_j \psi_j$. V našem případě je to první sloupec, tedy $j^* = 1$ a $\psi_{j^*} = 3$.

Protože se v obou případech jedná o stejný prvek $a_{i^*,j^*} = a_{2,1} = 3$, je tento prvek sedlovým prvkem hry a strategie $i^* = 2$, $j^* = 1$ je optimální strategie. Prvek $a_{i^*,j^*} = J(i^*, j^*)$ se nazývá **cena hry**.

Všimněme si, že odchýlí-li se libovolný hráč od takové optimální strategie, poškodí se tím a protihráč získá. Dále je důležité si uvědomit, že hráči svá rozhodnutí nemusí v tomto případě tajit. Každý hráč může své rozhodnutí zveřejnit před rozhodnutím druhého hráče a nic netratí. Pro sedlový bod totiž platí

$$\max_i \min_j a_{i,j} = \min_j \max_i a_{i,j}$$

První hráč určuje podle výkladu nejprve pro všechna i veličinu φ_i rovnu $\varphi_i = \min_j a_{i,j}$ a potom z těchto prvků hledá prvek maximální, tedy $\max_i \varphi_i = \max_i \min_j a_{i,j}$.

Podobně druhý určuje nejprve pro všechna j veličinu ψ_j rovnu $\psi_j = \max_i a_{i,j}$ a potom z těchto prvků hledá prvek minimální, tedy $\min_j \psi_j = \min_j \max_i a_{i,j}$.

V teorii her veličina $\max_i \min_j a_{i,j}$ je **dolní cena hry**. Je to zaručená výhra prvního hráče. Naopak veličina $\min_j \max_i a_{i,j}$ je **horní cena hry**. Je to zaručená výhra druhého hráče. Označení horní a dolní cena plyne z toho, že obecně platí $\max_i \min_j a_{i,j} \leq \min_j \max_i a_{i,j}$.

Příklad: Vyřešíme si problém nákupu uhlí na zimu. Problém je v tom, že uhlí v létě je laciné, ale v létě nevím, jaká bude následující zima. Bude-li zima mírná, tak v zimě spotřebuji 10q uhlí, bude-li zima normální, pak spotřebuji 15q uhlí a bude-li zima krutá, pak spotřebuji 20q uhlí. V létě 1q uhlí stojí 100Kč a v zimě jsou ceny uhlí 100Kč, 150Kč a 200Kč podle toho, bude-li zima mírná, normální či krutá. Budeme předpokládat, že zbytek uhlí nevyužiji, např. se budu po zimě stěhovat. Moje rozhodování spočívá v tom, jaké množství uhlí koupit v létě, abych minimalizoval své celkové vydání za nákup uhlí na zimu (včetně dokoupení uhlí v zimě, pokud letní nákup nestačí). Příroda, jako můj protihráč, má tři možnosti a sice, že zima bude mírná, normální nebo krutá. Moje rozhodování spočívá v tom, zda v létě koupit 10q, 15q nebo 20q. Vydání při všech možnostech je shrnuto v následující tabulce

		Zima		
		mírná	normální	krutá
koupeno v létě [q]	10	1000	1750	3000
	15	1500	1500	2500
	20	2000	2000	2000

V tabulce jsou uvedena vydání, která se pochopitelně budeme snažit minimalizovat. Naše rozhodování spočívá ve volbě řádku a strategie prvního hráče, který volí řádky, dle zvyklosti je taková, že první hráč hledá nejvyšší výhru. Proto změníme znaménka u všech prvků matice hry v předchozí tabulce. Dostaneme proto následující tabulku, ze které nalezneme obvyklým postupem horní a dolní cenu hry.

			φ_i(min. v řádku)
$\begin{bmatrix} -1000 & -1750 & -3000 \\ -1500 & -1500 & -2500 \\ -2000 & -2000 & \boxed{-2000} \end{bmatrix}$			$\begin{matrix}(-3000)\\(-2500)\\(-2000)\end{matrix}$
ψ_j(max. ve sloupci)	(-1000)	(-1500)	(-2000)

Z předchozí tabulky je zřejmé, že hra má sedlový bod v -2000. Optimální vydání je tedy 2000Kč a optimální strategie je nakoupit v létě 20q uhlí a vydat tedy 2000Kč.

Je zřejmé, že druhý hráč (příroda) není inteligentní hráč, protože nevolí svá rozhodnutí tak, aby maximálně poškodil prvního hráče. V tom není tento příklad realistický. O takových hrách pojednáme později.

5.1.3 Smíšené rozšíření maticové hry

Situace však může být poněkud komplikovanější. Mějme na příklad hru s výplatní maticí

$$\begin{array}{cc} & \varphi_i \\ \mathbf{A} = \begin{bmatrix} 11 & 5 \\ 7 & 9 \end{bmatrix} & \begin{matrix}(5)\\(7)\end{matrix} \\ \psi_j \quad (11) \quad (9) & \end{array} \tag{5.7}$$

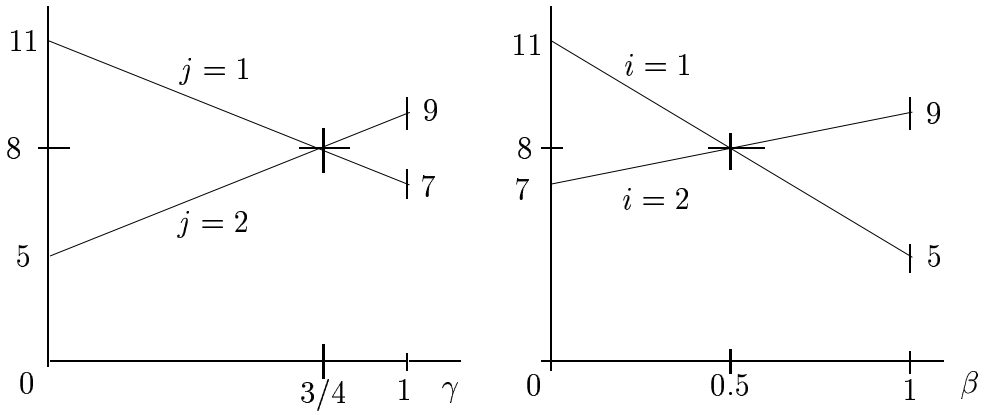
Podle předchozího postupu vybere první hráč druhý řádek (volí strategii $i = 2$), neboť v něm je minimální prvek $\varphi_i = \min_j a_{i,j}$ největší (je to prvek $a_{2,1} = 7$). Druhý hráč obdobně vybere druhý sloupec (strategii $j = 2$), neboť v něm je maximální prvek $\psi_j = \max_i a_{i,j}$ nejmenší (je to prvek $a_{2,2} = 9$).

První hráč má zaručenou v nejhorším případě výhru 7, což je dolní cena hry. Naopak druhý hráč má zaručenou výhru 9, což je horní cena hry.

Hráči nyní mají důvod svá rozhodnutí tajit, neboť dozví-li se hráč o rozhodnutí protivníka, může z této informace získat pro sebe výhodu. V této hře totiž dolní cena hry není rovna horní ceně hry - hra nemá sedlový bod. Platí $7 = \max_i \min_j a_{i,j} < \min_j \max_i a_{i,j} = 9$.

Pokud tedy volí hráči pevně nějaké strategie, již taková jednoduchá hra nemá řešení. Pevné strategie nazýváme **ryzí strategie**. Pokud hra nemá sedlový bod, nemá tedy řešení na množině ryzích strategií.

Abychom mohli řešit každou maticovou hru, budeme předpokládat, že každý hráč bude své strategie vybírat náhodně s určitou pravděpodobností. Strategie vybírané s určitou pravděpodobností nazýváme **smíšené strategie**. Situace, při které volí první hráč



Obrázek 5.1: Smíšené strategie 1. hráče Smíšené strategie 2. hráče

smíšené strategie ve hře s výplatní matici (5.7) je znázorněna na obr. 5.1a. Na obrázku jsou úsečkami znázorněny ceny hry. Druhý hráč volí ryzí strategie $j = 1$ nebo $j = 2$ a

první hráč volí smíšené strategie. Souřadnice γ určuje pravděpodobnost volby strategií prvního hráče. Pro $\gamma = 0$ volí první hráč ryzí strategii $i = 1$ a pro $\gamma = 1$ volí ryzí strategii $i = 2$. Číslo $p_2 = \gamma$ značí pravděpodobnost volby druhé strategie, podobně $p_1 = 1 - \gamma$ je pravděpodobnost volby první strategie. Platí samozřejmě $p_1 + p_2 = 1$ a $\gamma \in [0, 1]$. Pokud druhý hráč volil ryzí strategii $j = 1$, bude cena hry

$$p_1 a_{1,1} + p_2 a_{2,1} = (1 - \gamma)11 + 7\gamma$$

Tato funkce je znázorněna na obr. 5.1a úsečkou spojující body o pořadnicích 11 a 7. Podobně, volí-li druhý hráč strategii $j = 2$, bude cena hry v závislosti na γ vyjádřena

$$p_1 a_{1,2} + p_2 a_{2,2} = (1 - \gamma)5 + 9\gamma$$

Tato funkce je znázorněna na obr. 5.1a úsečkou spojující body o pořadnicích 5 a 9.

Obě úsečky se protínají v bodě o souřadnicí $\gamma = 3/4$, to znamená, že první hráč volí strategii $i = 1$ s pravděpodobností $p_1 = 1 - \gamma = 1/4$ a druhou strategii $i = 2$ s pravděpodobností $p_2 = \gamma = 3/4$. Cena hry je potom rovna 8. Tuto cenu hry nemůže druhý hráč ovlivnit.

Volí-li první hráč smíšené strategie s pravděpodobnostmi jinými než odpovídá průsečíku úseček na obr. 5.1a, poškodí se, neboť druhý hráč může svou volbou strategie cenu hry snížit. Zřejmě pro $\gamma \in (0, 3/4)$ bude tedy druhý volit ryzí strategii $j = 2$ a pro $\gamma \in (3/4, 1)$ volí druhý hráč ryzí strategii $j = 1$.

Podobně lze nalézt smíšené strategie druhého hráče podle obr. 5.1b . Zde také $q_2 = \beta$ je pravděpodobnost, s níž volí druhý hráč druhou strategii a $q_1 = 1 - \beta$ je pravděpodobnost volby první strategie $j = 1$.

Volí-li první hráč strategii $i = 1$, je cena hry v závislosti na pravděpodobnosti volby strategie druhého hráče znázorněna úsečkou spojující body 11 a 5. Cena hry je určena rovnicí

$$q_1 a_{1,1} + q_2 a_{1,2} = (1 - \beta)a_{1,1} + \beta a_{1,2} = (1 - \beta)11 + 5\beta.$$

Podobná lineární závislost platí pro ryzí strategii prvního hráče $i = 2$. Obě úsečky se protínají v bodě o souřadnici $\beta = 0.5$, cena hry je opět rovna 8.

Volíme-li pravděpodobnosti volby strategie druhého hráče $q_1 = 0.5$, $q_2 = 0.5$ a pravděpodobnosti volby strategie prvního hráče $p_1 = 0.25$ a $p_2 = 0.75$, pak tato rozšířená hra má sedlový bod a optimální cena hry je rovna 8.

Hledáme-li optimální strategie mezi pravděpodobnostními předpisy, dostaneme nekonečnou hru, kterou označujeme jako **smíšené rozšíření** původní hry.

Množinu strategií této hry budeme značit tak, že původní symbol pro množinu strategií dáme do okrouhlé závorky. Prostory strategií prvního a druhého hráče jsou tedy

$$\begin{aligned} (\mathbf{U}) &= \left\{\mathbf{u} = [u_1, \ldots, u_m]^T, \sum u_i = 1; u_i \geq 0\right\} \\ (\mathbf{V}) &= \left\{\mathbf{v} = [v_1, \ldots, v_n]^T, \sum v_j = 1; v_j \geq 0\right\} \end{aligned} \tag{5.8}$$

Potom u_i, v_j jsou rovny pravděpodobnostem s nimiž první hráč volí strategii i resp. druhý hráč volí strategii j. Výplatní funkce této hry je

$$J(u, v) = \sum_{i=1}^{m} \sum_{j=1}^{n} u_i a_{i,j} v_j = \mathbf{u}^T \mathbf{A} \mathbf{v} \tag{5.9}$$

O smíšeném rozšíření maticových her platí základní věta teorie maticových her, která tvrdí, že **smíšené rozšíření každé maticové hry má řešení**. Provedeme konstruktivní důkaz předchozího tvrzení, z něhož získáme návod na řešení smíšeného rozšíření maticové hry lineárním programováním.

Uvažujme nejprve maticovou hru, jejíž matice má všechny prvky kladné. Označíme cenu hry jako γ, pak ($\gamma = J(\mathbf{u}^*, \mathbf{v}^*)$). Pro neoptimální strategie platí

$$\mathbf{u}^T \mathbf{A} \mathbf{v}^* \ \leq \ \gamma \ \leq \ (\mathbf{u}^*)^T \mathbf{A} \mathbf{v} \tag{5.10}$$

pro všechny $\mathbf{u} \in (\mathbf{U})$; $\mathbf{v} \in (\mathbf{V})$. Protože $\mathbf{u} \geq 0$; $\mathbf{v} \geq 0$ a dle předpokladu také $a_{i,j} \geq 0$, pak $\gamma \geq 0$.

Aby pravá nerovnost v (5.10) platila pro všechny $\mathbf{v} \in (\mathbf{V})$ stačí, aby platila pro všechny $\mathbf{v}$ ve tvaru $[1, 0, \ldots, 0]^T$, $[0, 1, 0, \ldots, 0]^T$, až $[0, 0, \ldots, 1]^T$, to je pro ryzí strategie. Pro ryzí strategie dostaneme soustavu nerovností

$$\begin{aligned} a_{11}u_1^* + a_{21}u_2^* + \ldots + a_{m1}u_m^* &\geq \gamma \quad \text{pro} \quad \mathbf{v} = [1, 0, \ldots, 0]^T \\ &\vdots \\ a_{1n}u_1^* + a_{2n}u_2^* + \ldots + a_{mn}u_m^* &\geq \gamma \quad \text{pro} \quad \mathbf{v} = [0, 0, \ldots, 1]^T \end{aligned} \tag{5.11}$$

Z levé nerovnosti v (5.10) dostaneme při volbě ryzích strategií $\mathbf{u}$ obdobné nerovnosti ve tvaru

$$\begin{aligned} a_{11}v_1^* + a_{12}v_2^* + \ldots + a_{1n}v_n^* &\leq \gamma \quad \text{pro} \quad \mathbf{u} = [1, 0, \ldots, 0]^T \\ &\vdots \\ a_{m1}v_1^* + a_{m2}v_2^* + \ldots + a_{mn}v_n^* &\leq \gamma \quad \text{pro} \quad \mathbf{u} = [0, 0, \ldots, 1]^T \end{aligned} \tag{5.12}$$

Zavedeme nové nezáporné proměnné

$$x_i = \frac{u_i^*}{\gamma}; \qquad y_j = \frac{v_j^*}{\gamma} \tag{5.13}$$

Potom soustava nerovností (5.11) přejde do tvaru

$$\begin{aligned} a_{11}x_1 + a_{21}x_2 + \ldots + a_{m1}x_m^* &\geq 1 \\ &\vdots \\ a_{1n}x_1 + a_{2n}x_2 + \ldots + a_{mn}x_m^* &\geq 1 \end{aligned} \tag{5.14}$$

a podobně soustava (5.12) přejde do tvaru

$$a_{11}y_1 + a_{12}y_2 + \ldots + a_{1n}y_n \leq 1 \tag{5.15}$$

$$\vdots$$

$$a_{m1}y_1 + a_{m2}y_2 + \ldots + a_{mn}y_n \leq 1 \tag{5.16}$$

Potom soustava (5.14) je omezení úlohy lineárního programování s funkcí

$$x_1 + x_2 + \ldots + x_m = \frac{1}{\gamma} \tag{5.17}$$

kterou máme minimalizovat.

Je tomu vskutku tak, neboť hledáme u_i^* tak, aby γ bylo maximální. Podle (5.13) platí $u_i^* = \gamma x_i$. Sečteme-li předchozí rovnice pro všechna i, pak $\sum u_i^* = 1 = \gamma \sum x_i$ což je kritétium (5.17). Je-li γ výběrem u_i^* maximalizováno, pak x_i hledáme tak, aby (5.17) bylo minimalizováno.

Obdobně nerovnosti (5.15) jsou omezení úlohy lineárního programování, která je duální k předchozí úloze. Kritérium je zde rovno

$$y_1 + y_2 + \cdots + y_n = \frac{1}{\gamma} \tag{5.18}$$

kterou máme výběrem y_j maximalizovat. Je to tedy úloha lineárního programování ve tvaru

$$\max\left\{\mathbf{c}^T\mathbf{y} : \mathbf{A}\mathbf{y} \leq \mathbf{b}; y_j \geq 0; j = 1, \ldots, n; \mathbf{c}^T = [1, 1, \ldots, 1]; \mathbf{b}^T = [1, 1, \ldots, 1]\right\} \tag{5.19}$$

Cena hry je podle (5.18)

$$J(\mathbf{u}^*, \mathbf{v}^*) = \gamma = \frac{1}{\mathbf{c}^T\mathbf{y}} = \frac{1}{\sum y_i} \tag{5.20}$$

a smíšené strategie druhého hráče jsou

$$\mathbf{v}^* = [y_1\gamma, \ldots, y_n\gamma]^T \tag{5.21}$$

Duální řešení této úlohy určí smíšené strategie prvního hráče

$$\mathbf{u}^* = [x_1\gamma, \ldots, x_m\gamma]^T \tag{5.22}$$

neboť podle (5.14) a (5.17) je duální úloha lineárního programování

$$\min\left\{\mathbf{b}^T\mathbf{x} : \mathbf{A}^T\mathbf{x} \geq \mathbf{c}; x_i \geq 0; i = 1, \ldots, m; \mathbf{c}^T = [1, \ldots, 1]; \mathbf{b}^T = [1, \ldots, 1]\right\} \tag{5.23}$$

Povšimněme si, že primární úloha v obvyklém tvaru určí jako primární řešení smíšené strategie druhého hráče.

Předpoklad o kladnosti prvků matice $\mathbf{A}$ není podstatný, neboť optimální strategie smíšeného rozšíření se nezmění, přičteme-li ke každému prvku této matice stejné libovolné číslo k. Pokud tedy matice $\mathbf{A}$ nemá všechny prvky kladné (nezáporné), pak ke všem jejím prvkům přičteme takové kladné číslo k, aby prvky nové matice již nezáporné byly. Cenu původní hry získáme odečtením konstanty k od ceny upravené hry.

Příklad: Pomocí lineárního programování vyřešíme maticovou hru, jejíž matice $\mathbf{A}$ je dle (5.7). Podle (5.19) se tedy jedná o úlohu lineárního programování ve tvaru

$$\max\left\{y_1 + y_2 \ : \ \begin{array}{rcl} 11y_1 + 5y_2 & \leq & 1 \\ 7y_1 + 9y_2 & \leq & 1 \end{array} \ ; \ y_i \geq 0\right\}$$

Ze simplexové tabulky plyne řešení $y_1 = 1/16$, $y_2 = 1/16$ a cena hry je rovna $\gamma = J^* = 1/\sum y_i = 8$.

Tabulka 5.1: Simplexová tabulka pro řešení maticové hry

y_1	y_2	y_3	y_4	**b**
11	5	1	0	1
7	9	0	1	1
-1	-1	0	0	0
64/9	0	1	-5/9	4/9
7/9	1	0	1/9	1/9
-2/9	0	0	1/9	1/9
1	0	9/64	-5/64	1/16
0	1	-7/64	11/64	1/16
0	0	1/32	3/32	1/8

Podle (5.21) jsou optimální strategie druhého hráče

$$\mathbf{v}^* = [\gamma y_1 \ , \ \gamma y_2]^T = [0.5 \ , \ 0.5]^T$$

Duální řešení čteme v posledním řádku simplexové tabulky $x_1 = 1/32$, $x_2 = 3/32$. Smíšené strategie prvního hráče jsou podle (5.22)

$$\mathbf{u}^* = [\gamma x_1 \ , \ \gamma x_2]^T = [1/4 \ , \ 3/4]^T$$

Výsledek souhlasí s předchozím grafickým odvozením. Jednoduché grafické řešení lze použít pouze pro hry typu 2×2.

Příklad: Návrh kvalitního zesilovače
Kvalita nějakého zesilovače závisí na vlastnostech jednoho kondenzátoru. Tento kondenzátor v běžném provedení stojí 1Kč, ale 10 Kč stojí záruční oprava, je-li vadný. Můžeme však použít kondenzátor kvalitnější, který stojí 6Kč, za nějž výrobce nese záruku v tom smyslu, že uhradí náklady na opravu, je-li kondenzátor vadný. Můžeme si také zvolit mimořádně kvalitní kondenzátor, který stojí 10Kč, za nějž výrobce ručí tak, že v případě, že je vadný, nejen uhradí záruční opravu, ale ještě vrátí částku za jeho nákup.

Tuto reálnou rozhodovací situaci můžeme popsat maticovou hrou. První hráč má tři možné strategie a sice koupit normální, lepší nebo nejkvalitnější kondenzátor. Druhý hráč - příroda - rozhodne o tom, zda bude kondenzátor dobrý či vadný. Výplatní matice této hry je v následující tabulce

	Příroda	
	Kondenzátor dobrý	Kondenzátor špatný
Koupě normálního kondenzátoru	-1	-10
Koupě kvalitnějšího kondenzátoru	-6	-6
Koupě nejkvalitnějšího kond.	-10	0

Prvky výplatní matice **A** jsou vydání, mají záporné znaménko, neboť první hráč podle konvence volí řádky a maximalizuje výhru - tedy minimalizuje vydání.

Matice **A** nesplňuje podmínku nezápornosti prvků a proto ji upravíme tak, že ke každému jejímu prvku přičteme +10. Úloha lineárního programování, odpovídající smíšenému rozšíření maticové hry, je potom

$$\max \left\{ y_1 + y_2 \ : \ \begin{array}{rcrcl} 9y_1 & & & \leq & 1 \\ 4y_1 & + & 4y_2 & \leq & 1 \\ & & 10y_2 & \leq & 1 \end{array} \ ; \ y_i \geq 0 \right\}$$

Řešení provedeme simplexovým algoritmem. Uvedeme zde pouze řešení. Řešení primární úlohy je $y_1 = 1/9$, $y_2 = 1/10$. Cena upravené hry je tedy $\gamma = 1/(y_1 + y_2) = 90/19$. Cena původní hry je potom

$$J(\mathbf{u}^*, \mathbf{v}^*) = \frac{90}{19} - 10 = -\frac{100}{19} = -5,26$$

Optimální strategie druhého hráče je

$$\mathbf{v}^* = [\gamma y_1 \ , \ \gamma y_2]^T = [10/19 \ \ 9/19]^T$$

Optimální strategie prvního hráče je

$$\mathbf{u}^* = [\gamma x_1 \ , \ \gamma x_2 \ , \ \gamma x_3]^T = [10/19 \ \ 0 \ \ 9/19]^T$$

Opět zde platí, že druhý hráč - přiroda - není inteligentní protihráč a proto tento model situace není zcela přesný.

Rozmysleme si, jak v praxi realizovat smíšené strategie. Je třeba si zvolit nějaký náhodový mechanismus, který generuje čísla se stejnou pravděpodobností jako jsou pravděpodobnosti smíšených strategií. Při realizaci náhodného pokusu dostaneme řešení.

Mějme například smíšené strategie rovné $\mathbf{u}^* = [1/6, 1/3, 1/2]$. Náhodný generátor, který generuje výstupy s pravděpodobnostmi 1/6, 1/3 a 1/2 můžeme realizovat například náhodným pohledem na vteřinovou ručičku hodinek. Pokud bude ukazatel v poloze 0 až 10s - volíme první strategii, bude-li v poloze 10s až 30s = volíme druhou strategii a bude-li konečně ukazatel v poloze 30s až 60s = zvolíme třetí strategii.

Je zřejmé, že pesimisticky založení hráči nebudou chtít své rozhodnutí svěřit nějakému náhodovému mechanismu. Náhodový mechanismus může zvolit strategii značně nevýhodnou, mající třeba malou pravděpodobnost. Nic nám nepomůže útěcha, že toto rozhodnutí je objektivně nejlepší a při opakovaných pokusech by se jeho výhoda projevila.

V tom je ošidnost celého pojetí pravděpodobnosti. Víme, že pravděpodobnost pádu tašky se střechy domu je velmi malá. Znalost této pravděpodobnosti nám ale vůbec nepomůže, když konkrétně na nás ta taška spadne.

5.2 Rozhodování při riziku a neurčitosti

Nyní budeme předpokládat, že pouze jeden (podle konvence první) účastník konfliktu je inteligentní. Druhý je lhostejný ke své výhře a volí své strategie náhodně.

K popisu takové konfliktní situace použijeme hry dvou hráčů v normálním tvaru. Výplatní funkce druhého hráče nemusíme uvažovat, neboť on je lhostejný ke své výhře a jeho výplatní funkce nemá na jeho rozhodování vliv.

Druhý hráč se rozhoduje tak, že volí své strategie $\mathbf{v}$ náhodně podle nějakého rozložení pravděpodobnosti $p(\mathbf{v})$ na množině $\mathbf{V}$. Chování prvního hráče závisí na tom, zda rozložení $p(\mathbf{v})$ zná či nezná. Zná-li první hráč rozložení $p(\mathbf{v})$, jedná se o **rozhodovánání při riziku**, nezná-li první hráč rozložení $p(\mathbf{v})$, jedná se o **rozhodování při neurčitosti**.

5.2.1 Rozhodování při riziku

Při rozhodování s rizikem spočteme pro každou strategii $\mathbf{u}$ prvního hráče střední hodnotu výhry

$$\mathcal{E}(\mathbf{u}) = \int_{\mathbf{V}} J(\mathbf{u}, \mathbf{v}) p(\mathbf{v}) d\mathbf{v} \tag{5.24}$$

Strategie prvního hráče je optimální, když je střední hodnota výhry maximální

$$\mathbf{u}^* = \arg \max_{\mathbf{u} \in \mathbf{U}} \mathcal{E}(\mathbf{u}) \tag{5.25}$$

Jedná-li se o konečnou hru s maticí hry $\mathbf{A}$, pak, volí-li druhý hráč sloupec j s pravděpodobností p_j, volí první hráč takovou strategii i^*, pro kterou je střední hodnota výhry maximální. Platí tedy

$$i^* = \arg \max_{i} \sum_{j=1}^{n} a_{i,j} p_j \tag{5.26}$$

Každou výhru ve sloupci násobím pravděpodobností p_j příslušného sloupce a první hráč volí takovou strategii (takový řádek), pro kterou je řádkový součet takto vyvážených prvků maximální.

5.2.2 Rozhodování při neurčitosti

V případě neurčitosti vznikají obtíže s tím, jak definovat optimální strategie. Neexistuje jediná volba optimální strategie, existují různé definice optimální strategie, které se nazývají principy. Uvedeme některé z nich.

Princip minimaxu
Pesimisticky založení hráči volí takovou strategii, aby se zajistili proti nejhoršímu případu. Podle principu minimaxu volíme tedy takovou strategii, abychom si zajistili dolní cenu hry. Volíme tedy i^* takové, aby

$$i^* = \arg \max_i \min_j a_{i,j}$$

U her se sedlovým bodem je to samozřejmě optimální řešení.

Princip nedostatečné evidence
Neznáme-li jak druhý hráč volí své strategie, můžeme předpokládat, že všechny strategie druhého hráče jsou rovnocenné, volí je tedy se stejnou pravděpodobností.

Je-li množina $\mathbf{V}$ nekonečná, pak předpokládáme u druhého hráče rozložení s nejmenším obsahem informace. Je-li množina $\mathbf{V}$ interval, pak rozložení s nejmenším obsahem informace je rovnoměrné rozložení. Je-li množina $\mathbf{V}$ interval $[\gamma\ ,\ \infty)$, pak rozložení s nejmenším obsahem informace neexistuje. Předpokládáme-li znalost střední hodnoty μ daného rozložení, pak rozložení s nejmenším obsahem informace je exponencielní rozložení. Je-li konečně množina $\mathbf{V} = E^1$, pak pro existenci rozložení s nejmenším obsahem informace musíme předpokládat znalost rozptylu a střední hodnoty rozložení. Potom rozložení s nejmenším obsahem informace je normální rozložení $\mathcal{N}(\mu, \sigma^2)$.

Předpokládáme-li znalost rozložení, problém se změní na rozhodování při riziku. Pro konečné hry stanovíme optimální strategii i^* jednoduše tak, že sečteme všechny prvky v řádcích matice $\mathbf{A}$ a vybereme jako optimální ten řádek, v němž je tento součet maximální

$$i^* = \arg \max_i \sum_{j=1}^{n} a_{i,j}$$

Princip minimaxu ztráty
Zde jako optimální volíme to rozhodnutí. které nás zajišťuje proti velkým ztrátám ve srovnání s rozhodnutím, které bychom učinili při znalosti ryzí strategie druhého hráče.

Definujeme si funkci ztrát $Z(\mathbf{u}, \mathbf{v})$

$$Z(\mathbf{u}, \mathbf{v}) = J(\mathbf{u}, \mathbf{v}) - \max_u\ J(\mathbf{u}, \mathbf{v})$$

a optimální strategie je taková strategie $\mathbf{u}^*$, pro níž je maximální výraz $\min_v Z(\mathbf{u}, \mathbf{v})$.

U konečných her místo funkce ztrát počítáme ztrátovou matici $\mathbf{Z}$ s prvky $z_{i,j}$

$$z_{i,j} = a_{i,j} - \max_k\ a_{k,j}$$

Optimální je taková strategie i^*, která maximalizuje řádková minima matice ztrát. Zajistíme si tak dolní cenu hry s maticí hry $Z(\mathbf{u}, \mathbf{v})$.

Použití tohoto principu nás chrání proti námitkám těch, kteří "jsou po bitvě generály".

Princip ukazatele optimismu
Podle tohoto principu vypočteme funkce

$$\begin{aligned} M(\mathbf{u}) &= \max_{\mathbf{v}\in\mathbf{V}} J(\mathbf{u}, \mathbf{v}) \\ m(\mathbf{u}) &= \min_{\mathbf{v}\in\mathbf{V}} J(\mathbf{u}, \mathbf{v}) \end{aligned}$$

a volíme ukazatel optimismu $\alpha \in [0\ ,\ 1]$ prvního hráče. Optimální strategie $\mathbf{u}^*$ podle tohoto principu maximalizuje výraz

$$P(\mathbf{u}) = \alpha M(\mathbf{u}) + (1-\alpha)m(\mathbf{u})$$

Pro $\alpha = 0$ je tento princip totožný s principem minimaxu - zajistíme si dolní cenu hry. Pro $\alpha = 1$ volíme takovou strategii, v jejímž řádku je maximální prvek matice hry. Riziko již zde splývá s hazardem.

Pro konečné hry volíme strategii i^* pro kterou je maximální výraz

$$\alpha \max_j a_{i,j} + (1-\alpha) \min_j a_{i,j}.$$

Příklad : Mějme hru s výplatní maticí

$$\mathbf{A} = \begin{bmatrix} 1 & 4 & 3 \\ 5 & 8 & 0 \\ 4 & 0 & 6 \end{bmatrix}$$

Spočteme strategie podle různých principů.
Podle principu minimaxu určíme dolní cenu hry, která je rovná 1 a odpovídající strategie je $i^* = 1$. Podle principu nedostatečné evidence je součet prvků v řádku maximální v druhém řádku, proto $i^* = 2$.

Pro určení strategie podle minimaxu ztráty sestrojíme ztrátovou matici $\mathbf{Z}$ tak, že v každém sloupci nalezneme maximální prvek a ten odečteme od všech prvků ve sloupci, pak

$$\begin{array}{cc} & \psi_i \\ \mathbf{Z} = \begin{bmatrix} -4 & -4 & -3 \\ 0 & 0 & -6 \\ -1 & -8 & 0 \end{bmatrix} & \begin{matrix} (-4) \\ (-6) \\ (-8) \end{matrix} \end{array}$$

Řádková minima matice $\mathbf{Z}$ jsou vynesena v závorce vedle matice $\mathbf{Z}$. Maximum z řádkových minim je zřejmě (-4) a nastane při volbě strategie $i^* = 1$.

Nyní určíme strategii podle principu ukazatele optimismu. Ověřte, že pro hru s výplatní maticí podle tohoto příkladu je funkce $P(\mathbf{u})$ určena vztahy

$$P(1) = 3\alpha + 1; \qquad P(2) = 8\alpha; \qquad P(3) = 6\alpha$$

Pro $\alpha = 0$ je optimum podle ukazatele optimismu $i^* = 1$ a pro $\alpha = 1$ je $i^* = 2$, protože v druhém řádku leží maximální prvek. Strategii $i^* = 2$ budeme volit pro $\alpha \geq 1/5$.

Příklad: Volba léku.
Pacient má jedno z pěti možných virových onemocnění. Lékař má k dispozici tři druhy léků, jejichž léčebné účinky jsou závislé na tom, kterou nemoc má pacient. První lék zaručuje z 50% zdolání prvních čtyř nemocí. Druhý lék bezpečně léčí první druh onemocnění. Třetí lék ničí v 50% viry při druhém onemocnění a zaručeně zdolá pátý druh nemoci.

Situaci popíšeme maticovou hrou - strategií lékaře je volba jednoho ze tří léků a příroda volí jeden z pěti druhů onemocnění. Výplatní matice je v následující tabulce

		onemocnění					(minima řádek)
		1	2	3	4	5	
lék	1	50	50	50	50	0	(0)
	2	100	0	0	0	0	(0)
	3	0	50	0	0	100	(0)
(max. sloupců)		(100)	(50)	(50)	(50)	(100)	

Hra zřejmě nemá sedlový bod. Považujeme-li hru za antagonistický konflikt dvou inteligentních hráčů, je optimální řešení ve smíšených strategiích $\mathbf{u}^* = [2/3,\ 0,\ 1/3]^T$. Cena hry je $\gamma = 33\%$. Pokud nemáme žádné informace o pravděpodobnosti výskytu onemocnění, volí lékař s pravděpodobností 2/3 první lék a s pravděpodobností 1/3 třetí lék. Druhý lék nepoužije vůbec. Potom má naději 33%, že jeho léčba bude úspěšná.

Příroda jako druhý hráč ale není škodolibá, nevolí své strategie tak, aby nejvíce poškodila protivníka. Předpokládejme, že jsou známé pravděpodobnosti výskytu jednotlivých onemocnění, které nechť jsou po řadě rovny

$$\frac{1}{14}\ ;\ \frac{3}{14}\ ;\ \frac{3}{14}\ ;\ \frac{2}{14}\ ;\ \frac{5}{14}$$

Nyní se tedy jedná o hru s rizikem. Při použití prvního léku je pravděpodobnost úspěchu

$$50\frac{1}{14} + 50\frac{3}{14} + 50\frac{3}{14} + 50\frac{2}{14} + 0\frac{5}{14} = 50\frac{9}{14} = 32,1\%$$

Pro druhý lék je pravděpodobnost úspěchu obdobně $100\frac{1}{14}$ a pro třetí lék je pravděpodobnost úspěchu

$$50\frac{3}{14} + 100\frac{5}{14} = 50\frac{13}{14} = 48\%$$

Největší naději na úspěch je při použití třetí strategie - úspěch bude ve 48% případů.

Budou-li se nemoci vyskytovat s uvedenými pravděpodobnostmi, ale lékař bude volit své strategie podle smíšeného rozšíření hry, pak hodnota hry bude pouze

$$p_1\ 50\frac{9}{14} + p_3\ 50\frac{13}{14} = 37\%$$

Spočtěte strategie podle různých principů při hře s neurčitostí.

5.3 Neantagonistický konflikt dvou hráčů

Častější než konflikty s protichůdnými zájmy jsou konflikty, v nichž si každý účastník sleduje své vlastní zájmy.

Matematickým modelem takového konfliktu je hra dvou hráčů s nekonstantním součtem určená množinami (5.3). Zde je již obtížné a nejednoznačné definovat optimální strategii.

Uvažujme například spor dvou podniků. Jejich možné akce jsou:

1. Žalovat druhý podnik - tuto strategii označíme Z.
2. Nabídnout druhému podniku spojení v jedinou organizaci - strategie S.
3. Navrhnout druhému podniku ústupek - strategie U.

Výplatní dvojmatice hry s nekonstantním součtem je následující

$$\begin{array}{cc} & \begin{array}{ccc} Z & S & U \end{array} \\ \begin{array}{c} \\ \boxed{1} \\ \\ \end{array} & \begin{array}{c} Z \\ S \\ U \end{array} \begin{bmatrix} -1;\ -1 & 9;\ -10 & 9;\ -10 \\ -10;\ 9 & -5;\ 100 & 0;\ 0 \\ -10;\ 9 & 0;\ 0 & 5;\ 5 \end{bmatrix} \end{array}$$

V průsečíku strategií je dvojčíslí, na prvním místě je výplata prvního hráče $J_1(\mathbf{u}, \mathbf{v})$ (ten volí řádky) a na druhém místě je výplata druhého hráče $J_2(\mathbf{u}, \mathbf{v})$.

Z výplatní funkce je zřejmé, že žaluje-li jeden podnik druhý, získá pro sebe výhodu. Oboustranná žaloba není výhodná pro žádný podnik a spojení je výhodné pro druhý podnik.

Pro nalezení optimální strategie musíme před zahájením hry vědět, jaké jsou možnosti dohody s protivníkem.

Nelze-li uzavřít žádnou dohodu, budou volit hráči nejspíše strategie (Z, Z), neboť jednostranná odchylka každého hráče od této strategie zmenší jeho výhru.

Existuje-li možnost uzavřít závaznou dohodu, bude výhodné, volí-li hráči strategii (S, S). Jejich společná výhra bude $100 - 5 = 95$. Druhý hráč ale musí ze své výhry dát prvnímu hráči kompenzaci za to, že mu k této výhře dopomohl. Problém, jak společnou výhru rozdělit, skrývá v sobě další konfliktní situaci, kterou nutno vyřešit před hraním této hry.

Je-li možné uzavřít dohodu o volbě strategií, ale není možno přerozdělit výhry - přerozdělení by mělo formu úplatku, který není možno vymáhat ani předem vyplatit, domluví se hráči zřejmě na strategiích (U, U).

Dostáváme tedy tři možnosti řešení neantagonistického konfliktu dvou účastníků:

- nekooperativní teorie
- kooperativní teorie s přenosnou výhrou
- kooperativní teorie s nepřenosnou výhrou

5.3.1 Nekooperativní teorie

Za vyhovující strategii v tomto případě volíme tu, jejíž jednostranné porušení poškodí hráče, který ji porušil. Dvojici $\overline{\mathbf{u}}, \overline{\mathbf{v}}$ nazýváme **rovnovážné strategie**. Pojem rovnovážné strategie u her s nekonstantním součtem je obdobný pojmu optimální strategie u antagonistických her.

Zde se ale může stát, že volí-li jeden hráč nerovnovážnou strategii, poškodí se sice, ale druhého hráče poškodí více. Stupeň vynucení rovnovážného bodu je zde slabší. Potíže navíc vznikají, je-li rovnovážných bodů více než jeden. Některý může být výhodný pro

jednoho hráče a jiný pro druhého. Jejich strategie se pak mohou sejít mimo rovnovážný bod.

Proto se zavádí pojem **dominující rovnovážný bod**, což je takový rovnovážný bod, pro který neexistuje jiný rovnovážný bod, který je výhodnější pro oba hráče. Podobně **záměnné rovnovážné body** jsou takové rovnovážné body, u nichž záměna rovnovážných strategií u libovolného hráče nemění cenu hry.

Je zřejmé, že volba rovnovážných strategií je rozumná pouze tehdy, je-li jediný dominující rovnovážný bod, nebo jsou-li všechny dominující rovnovážné body záměnné. Takový rovnovážný bod nazýváme **optimálním rovnovážným bodem**.

Neantagonistický konflikt dvou účastníků s konečným počtem strategií popisujeme tzv. **dvojmaticovou hrou**. Strategie prvního hráče jsou $\mathbf{U} = \{1, 2, \ldots, m\}$ a druhého hráče jsou $\mathbf{V} = \{1, 2, \ldots, n\}$. Výplatní funkce prvního a druhého hráče jsou

$$J_1(\mathbf{u}, \mathbf{v}) = a_{i,j}, \quad i \in \mathbf{U}; \qquad J_2(\mathbf{u}, \mathbf{v}) = b_{i,j}, \quad j \in \mathbf{V}$$

Prvky $a_{i,j}$, $b_{i,j}$ tvoří výplatní dvojmatici.

Nemá-li dvojmaticová hra rovnovážný bod, můžeme obdobně jako u obyčejné maticové hry zavést smíšené rozšíření dvojmaticové hry. Strategie u_i resp. v_j jsou pravděpodobnosti volby strategií prvního a druhého hráče. Výplatní funkce jsou $J_1(\mathbf{u}, \mathbf{v}) = \mathbf{u}^T\mathbf{A}\mathbf{v}$ resp. $J_2(\mathbf{u}, \mathbf{v}) = \mathbf{u}^T\mathbf{B}\mathbf{v}$, kde matice $\mathbf{A}$ resp. $\mathbf{B}$ mají prvky $a_{i,j}$ resp. $b_{i,j}$.

Smíšené rozšíření každé dvojmaticové hry má řešení. Řešení smíšeného rozšíření dvojmaticové hry vede na úlohu nelineárního (kvadratického) programování ve tvaru

$$\max\left\{\mathbf{x}^T(\mathbf{A}+\mathbf{B})\mathbf{y} - \mathbf{e}^T\mathbf{y} - \mathbf{f}^T\mathbf{x} \ : \mathbf{A}\mathbf{y} \le \mathbf{e}; \mathbf{B}^T\mathbf{x} \le \mathbf{f}; \mathbf{x} \ge 0; \mathbf{y} \ge 0\right\} \tag{5.27}$$

kde vektory $\mathbf{e}$, $\mathbf{f}$ jsou jednotkové vektory kompatibilních rozměrů. Optimální strategie jsou rovny

$$\mathbf{u}^* = \frac{\mathbf{x}^*}{\sum x_i^*}; \qquad \mathbf{v}^* = \frac{\mathbf{y}^*}{\sum y_j} \tag{5.28}$$

Matice $\mathbf{A}$ i $\mathbf{B}$ musí mít opět kladné prvky, pokud je nemají, upravíme je tak, že přičteme k matici $\mathbf{A}$ nebo $\mathbf{B}$ libovolné kladné číslo. Přitom v předchozí úloze nelineárního programování platí, že její maximum je rovno nule.

Příklad: Problém prestiže
Při jednání o nějakém problému je často důležitější otázka prestiže účastníků. Každý z účastníků může projevit buď neústupnost nebo ústupnost. Jednostranná ústupnost vede ke ztrátě prestiže, oboustranná neústupnost vede k důsledkům nepříznivým pro oba hráče. Modelem tohoto konfliktu je dvojmaticová hra, viz následující tabulka

$$\begin{array}{cc} & \boxed{2} \\ & \begin{array}{cc} U & N \end{array} \\ \boxed{1} \begin{array}{c} U \\ N \end{array} & \left[\begin{array}{cc} 0\ ;\ 0 & -10\ ;\ 10 \\ 10\ ;\ -10 & -100\ ;\ -100 \end{array}\right] \end{array}$$

kde U a N jsou dvě možné strategie každého hráče, značící ústupnost a neústupnost.

Rovnovážné body jsou dva a to strategie $(U,\ N)$ nebo $(N,\ U)$. Pro prvního hráče je výhodnější druhý rovnovážný bod (při něm on volí neústupnost) a pro druhého hráče je

výhodnější první rovnovážný bod (při něm on volí také neústupnost). Při tom neústupnost obou hráčů končí nepříznivě pro oba. Hra by měla rozumné řešení, kdyby ústupnost byla hodnocena výše než zisk prestiže.

Podobný případ existuje při uzavírání dohod. Jestliže existuje možnost získat výhodu jednostranným porušením dohody, má konflikt řešení porušit dohodu pro obě strany, což je ve svých důsledcích nevýhodné pro oba. Dohodu, jejíž porušení přináší výhodu pro toho, kdo ji porušil, je tedy lépe neuzavírat.

Příklad: Vězňovo dilema
Dva spolupachatelé mají možnost přiznání (P), nebo zapření (Z) své viny. Jednostranné zapírání by mohlo obviněnému značně uškodit. Jednostranné přiznání přináší výhodu tomu, kdo se přiznal, a proto se přiznají oba. Navrhněte ocenění v této dvojmaticové hře tak, aby hra měla buď jediný rovnovážný bod ($P,\ P$), nebo dva rovnovážné body ($Z,\ Z$) a ($P,\ P$).

5.3.2 Kooperativní teorie - přenosná výhra

Uvažujme nyní neantagonistický konflikt, v němž je možné uzavírat závazné smlouvy o volbě strategií i o přerozdělení výhry. V těchto hrách musíme vyřešit tři problémy:

- Kdy má smysl smlouvu uzavírat
- Jaké volit potom strategie
- Jak rozdělit výhru

Odpověď na první dva problémy je jednoduchá. Smlouvu má smysl uzavřít, získají-li hráči spoluprací více než při samostatném rozhodování.

Každý hráč si spočte svoji zaručenou výhru (dolní cenu hry). Pro prvního hráče je to výhra, kterou označíme J_u a pro druhého hráče je to výhra, kterou označíme J_v, pak

$$J_u = \max_u \min_v J_1(\mathbf{u}, \mathbf{v}), \qquad J_v = \max_v \min_u J_2(\mathbf{u}, \mathbf{v})$$

Hráči si spočtou zisk, budou-li spolupracovat, který je zřejmě roven

$$J_s = \max_{\mathbf{u},\mathbf{v}} [J_1(\mathbf{u}, \mathbf{v}) + J_2(\mathbf{u}, \mathbf{v})]$$

Je zřejmé, že má význam uzavírat dohodu pouze tehdy, je-li $J_s > J_u + J_v$. Hráči si pak zajistí minimální výhru a ještě mohou něco získat z přebytku.

Říkáme, že hra je podstatná, platí-li $J_s > J_u + J_v$, je-li $J_s = J_u + J_v$, pak je hra nepodstatná a v tom případě nemá smysl uzavírat dohody. Opačná nerovnost není možná, ověřte. Při podstatné hře volí hráči strategie u_i, v_j, zajišťující společnou maximální výhru.

Problém, jak rozdělit výhru, není již tak jednoduchý a jednoznačný. Označme a_1 částku, kterou dostane první hráč, a_2 je pak částka, kterou dostane druhý hráč. Dvojice ($a_1,\ a_2$) je **rozdělení**. Přijatelná omezení jsou zřejmě omezena vztahy

$$a_1 + a_2 = J_s, \qquad a_1 \geq J_u, \qquad a_2 \geq J_v$$

Omezení vyplývají z toho, že dělit se může pouze společná výhra, přičemž každý hráč chce získat nejméně tolik, kolik je jeho zaručená výhra. Množinu rozdělení $(a_1,\ a_2)$ splňující předchozí vztahy nazýváme **jádrem hry**. Uvedeme si některá možná rozdělení.

Tzv. **charitativní rozdělení**

$$a_1 = \frac{J_s}{2}, \qquad a_2 = \frac{J_s}{2}$$

je často nepřijatelné a mnohdy není ani jádrem hry.

Spravedlivé rozdělení je takové rozdělení, kdy se výhry rozdělí v poměru přínosu obou hráčů

$$a_1 \ : \ a_2 \ = \ (J_s - J_v) \ : \ (J_s - J_u)$$

Optimální rozdělení můžeme definovat jako rozdělení $(a_1^*,\ a_2^*)$, kde a_1^*, a_2^* jsou souřadnice těžiště jádra. Případně uspořádáme náhodný pokus, v němž na jádře zvolíme rovnoměrné rozdělení. Střední hodnota tohoto rozdělení určuje těžiště jádra a tím optimální rozdělení. Platí

$$\begin{aligned} a_1^* &= J_u + \frac{1}{2}\left(J_s - J_u - J_v\right) \\ a_2^* &= J_v + \frac{1}{2}\left(J_s - J_u - J_v\right)) \end{aligned} \tag{5.29}$$

Hráči si při optimálním rozdělení ponechají zaručenou výhru a o zbytek se rozdělí rovným dílem.

Rozdělení společné výhry není tedy jednoznačné. Všechny naše úvahy zde vycházely ze zaručených výher J_u a J_v. Někdy ale není opodstatněný ani předpoklad zaručené výhry. Například jednoduchá dvojmaticová hra s výplatní dvojmaticí

$$[0\ ,\ -1000\ \ ;\ \ 10\ ,\ 2]$$

má zřejmě $J_u = 0\ ,\ J_v = 2;\ J_s = 12;\ a_1^* = 5\ ,\ a_2^* = 7$. Těžko druhý hráč (volící sloupce) přinutí prvního hráče, aby mu ze společné výhry něco dal, neboť pohrůžka volby první strategie (prvního sloupce) u druhého hráče znamená především pro něj katastrofu.

5.3.3 Kooperativní teorie - nepřenosná výhra

Hráči zde mohou spolupracovat v tom smyslu, že mohou uzavírat závazné smlouvy o volbě strategie, ale nikoliv o přerozdělení společné výhry. Spolupráce je legální, přenos výhry má povahu úplatku. Zůstávají zde tedy pouze dva problémy

- Kdy má smysl uzavírat smlouvu o volbě strategií.
- Jaké strategie potom volit.

Dohodu o volbě strategií má smysl uzavírat, je-li alespoň pro jednoho hráče výhra v případě dohody vyšší než zaručená. **Dosažitelné rozdělení** je dvojice $(a_1\ ,\ a_2)$, pro kterou platí

$$\begin{aligned} a_1 = J_1(\mathbf{u},\mathbf{v}), \quad a_2 &= J_2(\mathbf{u},\mathbf{v}) \quad \text{pro některé } \mathbf{u}\in\mathbf{U},\ \mathbf{v}\in\mathbf{V} \\ a_1 \geq J_u; \qquad\quad a_2 &\geq J_v \end{aligned} \tag{5.30}$$

kde J_u a J_v jsou dolní ceny hry obou hráčů. Dohodu je tedy výhodné uzavřít, je-li

$$(a_1\ ,\ a_2)\ \neq\ (J_u\ ,\ J_v)$$

Množinu dosažitelných rozdělení můžeme dále zúžit zavedením tzv. **paretovského rozdělení**. Jde o takové dosažitelné rozdělení $a = (a_1\ ,\ a_2)$, že neexistuje jiné dosažitelné rozdělení, v němž by jeden hráč získal více a druhý netratil. Množinu paretovského rozdělení označíme $\mathcal{P}$.

Abychom nalezli jediné rozdělení, které bychom mohli označit za optimální, budeme hledat rozdělení, které je nejbližší ke střední hodnotě paretovských rozdělení. Nechť tedy $b = (b_1\ ,\ b_2)$ je střední hodnota rovnoměrného rozložení pravděpodobnosti na $\mathcal{P}$ (je-li $\mathcal{P}$ ohraničená množina). Rozdělení $a^* = (a_1^*\ ,\ a_2^*)$ nazveme optimálním rozdělením, jestliže

$$\|a^*\ ,\ b\| = \min_{\mathcal{P}} \|a\ ,\ b\| \tag{5.31}$$

Příklad : V tomto jednoduchém příkladu ukážeme, jaké různé výsledky dají řešení neantagonistického konfliktu dvou hráčů. Uvažujme konečnou dvojmaticovou hru s výplatní dvojmaticí

$$\begin{array}{cc} & (j) \\ (i) & \left[\begin{array}{ccc} 6\ ;\ 10 & 8\ ;\ 5 & 4\ ;\ 2 \\ 1\ ;\ 3 & 5\ ;\ 6 & 6\ ;\ 6 \\ 0\ ;\ 0 & 15\ ;\ 4 & 5\ ;\ 5 \end{array}\right] \end{array}$$

Podle **nekooperativní teorie** nalezneme nejprve všechny rovnovážné body. Jsou to zřejmě body $(i\ ,\ j)$ o souřadnicích $(1\ ,\ 1)$ a $(2\ ,\ 3)$ s výhrami $(6\ ,\ 10)$ a $(6\ ,\ 6)$. První z nich je sice výhodnější pro druhého hráče, ale není dominující. Proto neexistuje optimální rovnovážný bod. Pokud druhý hráč volí strategii $j = 1$, aby dosáhl pro něho výhodnější rovnovážný bod, ale první hráč volí strategii $i = 2$ nutnou pro dosažení druhého rovnovážného bodu, pak jejich výhra je $(1\ ,\ 3)$, což je nevýhodné pro oba. Hra při nekooperativní teorii nemá řešení.

Podle **kooperativní teorie s přenosnou výhrou** nalezneme nejprve zaručené výhry obou hráčů (dolní ceny her obou hráčů). Zřejmě platí $J_u = 4$, $J_v = 4$. Při spolupráci získají největší společnou výhru $J_s = 19$. Spolupráce je tedy výhodná, optimální strategie je pak $(i^*\ ,\ j^*) = (3\ ,\ 2)$ a optimální rozdělení společné výhry je podle (5.29) zřejmě $a_1^* = 9.5$, $a_2^* = 9.5$.

Při **kooperaci a nepřenosné výhře** určíme nejprve množinu dosažitelných rozdělení. Jsou to zřejmě rozdělení s výhrami $(6\ ,\ 10)$; $(8\ ,\ 5)$; $(5\ ,\ 6)$; $(6\ ,\ 6)$; $(5\ ,\ 5)$; $(15\ ,\ 4)$, tedy s výhrami alespoň $(J_u\ ,\ J_v) = (4\ ,\ 4)$. Dohodu je tedy výhodné uzavřít. Paretovská rozdělení jsou potom rozdělení $(6\ ,\ 10)$; $(8\ ,\ 5)$; $(15\ ,\ 4)$. Střední hodnota paretovských rozdělení je

$$b = \left(\frac{1}{3}(6+8+15)\ ;\ \frac{1}{3}(10+5+4)\right) = (9.6\ ;6.3)$$

Nejbližší rozdělení ke střední hodnotě je rozdělení $(8\ ;\ 5)$, které je tedy optimálním rozdělením a jemu odpovídá optimální strategie $(i^*\ ,\ j^*) = (1\ ,\ 2)$. □

Z tohoto jednoduchého příkladu je zřejmé, že druhý hráč vlastně uzavíráním dohod získal méně, získal pouze o jednotku více než je jeho dolní cena hry. Naopak první hráč

získal uzavřením dohody o čtyři více než je jeho dolní cena hry. Je to způsobeno vysokou výhrou prvního hráče při strategii $(i\ ,\ j) = (3\ ,\ 2)$.

Vidíme tedy, že volba optimální strategie v neantagonistickém konfliktu má silný subjektivní motiv. Je zřejmé, že složitější než získat řešení hry, je správné sestavit matematický model konfliktu. Tyto úvahy ústí v tzv. **teorii užitku** či **teorii rizika**. Zde se těmito zajímavými problémy nebudeme zabývat.

5.4 Příklady

1. Řešte graficky úlohu smíšeného rozšíření maticové hry s maticí hry $\mathbf{A}$ rozměru $2 \times n$. Jak poznáme z grafického řešení, že hra má sedlový bod?

2. Platí vždy pro cenu hry při smíšeném rozšíření

$$\begin{aligned} \text{Cena hry} &> \text{Dolní cena hry} \\ \text{Cena hry} &< \text{Horní cena hry} \\ \text{Cena hry} &= \frac{1}{2}(\text{Dolní cena hry} + \text{Horní cena hry}) \end{aligned}$$

3. V kterém kroku odvození algoritmu na řešení smíšeného rozšíření maticové hry pomocí lineárního programování je nutný předpoklad nezápornosti prvků matice hry $\mathbf{A}$?

4. Rozdělení nákladů na propagaci.
   Dva výrobci - hráči 1 a 2 - se zajímají o dva trhy A, B. Na trhu A lze získat zakázky se ziskem 150 jednotek. Na trhu B lze získat zakázky se ziskem 90 jednotek. Každý výrobce má prostředky buď na velkou propagaci na jednom trhu nebo malou propagaci na obou trzích. Každý hráč má tedy možnost volit jednu ze tří strategií - velká propagace na trhu A, velká propagace na trhu B, nebo konečně malá propagace na obou trzích.

   Zakázky a tím i zisk se dělí podle následujících pravidel:
   a) Vede-li na určitém trhu propagaci pouze jeden výrobce, získá celou zakázku.
   b) Vedou-li oba výrobci na určitém trhu stejnou propagaci nebo žádnou, získají polovinu zakázek.
   c) Vede-li jeden výrobce na určitém trhu velkou propagaci a druhý propagaci malou, získá první výrobce dvě třetiny zakázek a druhý pouze jednu třetinu zakázek.

   Sestavte matici hry a řešte ji. Jak je třeba změnit pravidla hry, aby hra neměla sedlový bod? Jeden výrobce volí pouze velké propagace, ale nevíme, na kterém trhu. Jak této informace může využít druhý hráč?

5. Příprava na zkoušku:
   Student se může připravit na zkoušku velmi důkladně, normálně, nebo vůbec nestudovat. Při tom neví, zda zkoušející bude přísný nebo ne a zda bude mít čas na podrobnou zkoušku či ne. Sestavte hru, zvolte ocenění jednotlivých strategií a řešte ji jako hru dvou inteligentních hráčů nebo jako hru s rizikem a neurčitostí.

6. Při řešení smíšeného rozšíření maticové hry můžeme někdy některý sloupec či řádek ve výplatní matici vypustit, protože u něho vždy vyjde nulová pravděpodobnost volby. Jaké vlastnosti musí mít příslušný sloupec či řádek?

7. Rozmyslete si algoritmus nalezení všech rovnovážných strategií ve dvojmaticové hře. Jak z nich vybereme dominující a záměnné rovnovážné body? Jak určíme optimální rovnovážné body?

Kapitola 6

Numerické metody

V této kapitole uvedeme některé numerické metody řešení problémů nelineárního programování. Všechny metody jsou založeny na iteračních postupech (algoritmech). Proto si nejprve stručně zavedeme potřebné pojmy a tvrzení.

6.1 Algoritmy a jejich konvergence

Definice:
Algoritmus $\mathcal{P}$ *je zobrazení definované na prostoru* $\mathbf{X}$*, které každému bodu* $\mathbf{x} \in \mathbf{X}$ *přiřazuje podmnožinu z* $\mathbf{X}$. □

Nemusí to tedy být zobrazení bodu na bod, ale bodu na množinu. Algoritmus generuje posloupnost bodů $\mathbf{x}_k \in \mathbf{X}$ tak, že z podmnožiny $\mathcal{P}(\mathbf{x}_k)$ vybere libovolný bod $\mathbf{x}_{k+1}$. Potom, je-li dán počáteční bod $\mathbf{x}_0$, algoritmus generuje posloupnost

$$\mathbf{x}_{k+1} \in \mathcal{P}(\mathbf{x}_k)$$

Obvykle numerickými metodami hledáme minimum t. zv. **hodnotící funkce**. Hodnotící funkce je často přímo kritérium $f(\mathbf{x})$, nebo třeba norma gradientu $|\mathrm{grad} f(\mathbf{x})|$. Algoritmus obvykle zajišťuje v každé iteraci pokles hodnotící funkce.

Zobecnění pojmu spojitosti při zobrazení bodu na bod je pojem uzavřeného zobrazení.

Definice:
Uzavřené zobrazení: *Zobrazení* $\mathcal{P}$ *bodu* $\mathbf{x} \in \mathbf{X}$ *na množinu* $\mathbf{y} \subset \mathbf{X}$ *je uzavřené v bodě* $\mathbf{x} \in \mathbf{X}$*, jestliže z předpokladu*

$$\mathbf{x}_k \longrightarrow \mathbf{x}; \quad \mathbf{x}_k \in \mathbf{X}, \qquad \mathbf{y}_k \longrightarrow \mathbf{y}; \quad \mathbf{y}_k \in \mathcal{P}(\mathbf{x}_k)$$

plyne

$$\mathbf{y} \in \mathcal{P}(\mathbf{x}).$$

Zobrazení $\mathcal{P}$ *je uzavřené, je-li uzavřené v libovolném bodě* $\mathbf{x} \in \mathbf{X}$. □

Při zobrazení bodu na bod vyplývá uzavřenost zobrazení ze spojitosti, to znamená, že pro $\mathbf{x}_{k+1} = \mathcal{P}(\mathbf{x}_k)$ a platí-li $\mathbf{x}_k \longrightarrow \mathbf{x}$, pak $\mathcal{P}(\mathbf{x}_k) \longrightarrow \mathcal{P}(\mathbf{x})$.

Mějme tedy množinu řešení Γ naší úlohy nelineárního programování, posloupnost bodů $\mathbf{x}_k$ je generována algoritmem $\mathbf{x}_{k+1} \in \mathcal{P}(\mathbf{x}_k)$, při čemž každý bod ostře snižuje hodnotící funkci tak dlouho, až dosáhneme Γ. Potom můžeme vyslovit následující větu:

Věta o globální konvergenci:
Nechť $\mathcal{P}$ je algoritmus na $\mathbf{X}$, dále předpokládáme, že je generována posloupnost $\{\mathbf{x}_k\}_{k=0}^{\infty}$ předpisem $\mathbf{x}_{k+1} = \mathcal{P}(\mathbf{x}_k)$. Je dána množina řešení $\Gamma \subset \mathbf{X}$ a předpokládáme, že

1. *všechna $\mathbf{x}_k$ jsou obsažena v kompaktní množině $\mathbf{S} \subset \mathbf{X}$*
2. *existuje spojitá hodnotící funkce Z na $\mathbf{X}$, že pro*

$$\begin{aligned} \mathbf{x} \notin \Gamma, \quad & \textit{pak} \quad Z(\mathbf{y}) < Z(\mathbf{x}), \quad \forall\ \mathbf{y} \in \mathcal{P}(\mathbf{x}) \\ \mathbf{x} \in \Gamma, \quad & \textit{pak} \quad Z(\mathbf{y}) \leq Z(\mathbf{x}), \quad \forall\ \mathbf{y} \in \mathcal{P}(\mathbf{x}) \end{aligned}$$

3. *Zobrazení je uzavřené pro všechny body mimo Γ*

Potom limita libovolné konvergentní posloupnosti $\{\mathbf{x}_k\}$ je řešení. □

Poznámka: Mnoho složitých algoritmů je složeno z několika jednodušších algoritmů, které definují uzavřené zobrazení. Je možno dokázat, že algoritmus složený z řady jednodušších uzavřených algoritmů je také uzavřený.

Globální konvergence nezaručuje nalezení globálního řešení našeho problému. Většina algoritmů nalezne pouze lokální řešení, které je rovno globálnímu řešení pouze tehdy, jsou-li splněny další předpoklady, které se nejčastěji týkají konvexnosti problému.

□

Konvergence iteračních výpočetních postupů

Optimální řešení budeme značit s hvězdičkou - tedy $\mathbf{x}^*$, λ^* a pod.

Iterační postup je **konečný**, jestliže po konečném počtu kroků K platí

$$\mathbf{x}^* = \mathcal{P}(\mathbf{x}_K)$$

Iterační postup je nekonečný, je-li $\mathbf{x}^* = \lim_{k\to\infty} \mathbf{x}_k$.

Konvergenci iteračních algoritmů zaručuje věta o pevném bodě, která ale vyžaduje, aby zobrazení definované algoritmem $\mathcal{P}$ bylo kontrahující zobrazení. Uvedeme si tedy potřebná tvrzení.

Definice:
Nechť S je podmnožina normovaného prostoru $\mathbf{X}$ a nechť P je zobrazení S na S. Pak $\mathcal{P}$ je **kontrahující zobrazení**, *jestliže existuje α, $0 \leq \alpha < 1$ takové, že*

$$\| \mathcal{P}(\mathbf{x}_1) - \mathcal{P}(\mathbf{x}_2) \| \ \leq \ \alpha \| \mathbf{x}_1 - \mathbf{x}_2 \| \quad \forall \mathbf{x}_1,\ \mathbf{x}_2 \in S.$$

□

Věta o pevném bodě:
Je-li P kontrahující zobrazení na uzavřené množině S, pak existuje jediný vektor $\mathbf{x}^* \in S$ splňující $\mathbf{x}^* = \mathcal{P}(\mathbf{x}^*)$. Dále $\mathbf{x}^*$ lze získat metodou postupných aproximací (algoritmem P), začneme-li z libovolného bodu $\mathbf{x}_0 \in S$. □

Často nás více než vlastní konvergence zajímá, jak rychle se iteračním postupem blížíme k řešení. Rychlost konvergence vyjadřujeme tzv. **řádem konvergence**. Přitom vzdálenost dvou bodů $\mathbf{x}$ a $\mathbf{y}$ můžeme hodnotit např. Eukleidovou metrikou

$$\sigma(\mathbf{x},\ \mathbf{y}) = \sum_{i=1}^{n}(x_i - y_i)^2. \tag{6.1}$$

kde x_i, y_i jsou $i$té složky vektoru $\mathbf{x}$ resp. $\mathbf{y}$.

Definice:
Řád konvergence *posloupnosti* $\mathbf{x}_k$ *je roven supremu nezáporných čísel* $p \geq 0$*, pro která platí*

$$0 \leq \lim_{k\to\infty} \frac{\sigma(\mathbf{x}_{k+1},\ \mathbf{x}^*)}{[\sigma(\mathbf{x}_k,\ \mathbf{x}^*)]^p} < \infty \tag{6.2}$$

Při tom předpokládáme, že iterační postup je nekonečný. □

Pokud $p = 1$ zavádíme jemnější míru konvergence.

Definice:
Jestliže

$$\lim_{k\to\infty} \frac{\sigma(\mathbf{x}_{k+1},\ \mathbf{x}^*)}{\sigma(\mathbf{x}_k,\ \mathbf{x}^*)} = \alpha < 1 \tag{6.3}$$

říkáme,že posloupnost x_k *konverguje* **lineárně s poloměrem konvergence** α*. Pro* $\alpha = 0$ *mluvíme o* **superlineární** *konvergenci.*

□

V dalších odstavcích pojednáme o vybraných numerických metodách nelineárního programování.

Začneme metodami jednorozměrové optimalizace, to je optimalizace při jedné nezávisle proměnné. Tyto metody se uplatní i při mnoharozměrové optimalizaci, neboť v prostoru vyšší dimenze volíme směr hledání a optimalizace v daném směru je vlastně úloha jednorozměrového hledání.

Potom uvedeme několik metod optimalizace, které nevyužívají derivace hodnotící funkce. Tyto metody jsou principielně jednoduché, jejich konvergence je mnohdy horší.

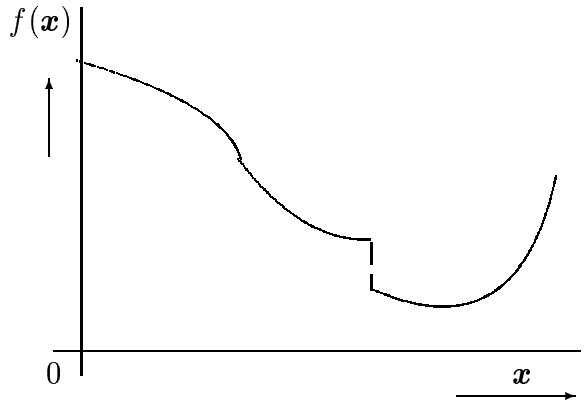
Následovat budou vybrané metody nelineárního programování, nejprve bez omezení a potom budeme diskutovat řešení optimalizačních problémů s omezením.

6.2 Jednorozměrová optimalizace

6.2.1 Fibonacciova metoda

Fibonacciova metoda řeší problém nalezení extrému nějaké funkce $f(x)$ jedné proměnné x na daném intervalu. Budeme předpokládat, že daná funkce je na zvoleném intervalu unimodální, to znamená, že má na zvoleném intervalu jeden extrém - např. jedno minimum, viz obr. 6.1.

Je dán počáteční interval d_1 nejistoty nezávisle proměnné, v níž hledané minimum leží. Naším úkolen je co nejvíce zmenšit interval nejistoty, ve kterém leží minimum funkce. Přitom máme k dispozici určitý počet pokusů (volbu bodů $x_i,\ i = 1, 2, \ldots, N$), ve kterých zjišťujeme hodnotu dané funkce.



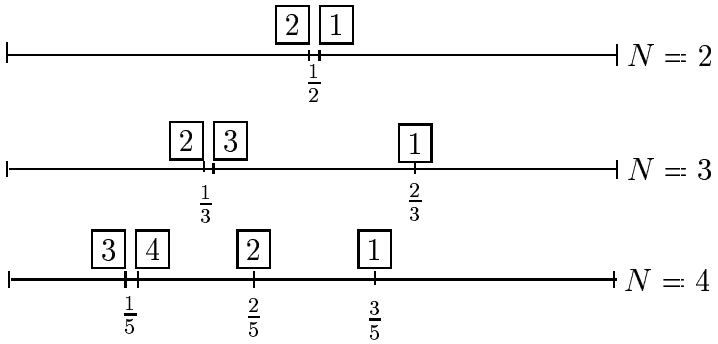
Obrázek 6.1: Unimodální funkce

Ukážeme, že interval nejistoty se v závislosti na počtu pokusů zmenšuje podle Fibonacciho čísel.

Fibonacciho čísla F_i tvoří posloupnost, která je popsaná rekurentním vztahem

$$F_i \;=\; F_{i-1} \;+\; F_{i-2}, \tag{6.4}$$

kde $F_0 = F_1 = 1$. Uvažujme nejprve, že máme k dispozici pouze dva pokusy ($N = 2$), ve kterých můžeme zjistit funkční hodnotu neznámé unimodální funkce. Abychom co nejvíce zmenšili interval nejistoty, ve kterém leží minimum této funkce, umístíme pokusy těsně vedle sebe uprostřed daného intervalu - viz obr. 6.2 pro $N = 2$. Body, ve kterých zjišťujeme hodnotu funkce, jsou v obr. 6.2 v rámečku. Je-li v bodě $\boxed{2}$ hodnota funkce menší než v bodě $\boxed{1}$, pak minimum funkce zřejmě leží v první polovině intervalu. V opačné případě leží v druhém intervalu. Interval nejistoty jsme tak zmenšili na polovinu pomocí dvou pokusů.



Obrázek 6.2: Fibonacciho metoda

Pokud budeme mít k dispozici tři pokusy - viz obr. 6.2 pro $N = 3$, pak naše strategie bude jiná. První dva pokusy - body $\boxed{1}$ a $\boxed{2}$ umístíme do $\frac{1}{3}$ a $\frac{2}{3}$ intervalu nejistoty . Pokud v bodě $\boxed{2}$ bude hodnota funkce menší než v bodě $\boxed{1}$, pak minimum funkce zřejmě leží v prvních dvou třetinách intervalu nejistoty. Třetí pokus umístíme těsně vedle druhého a je-li hodnota funkce v bodě $\boxed{2}$ menší než v bodě $\boxed{3}$, pak minimum zřejmě leží v první třetině intervalu nejistoty.

Pokud v bodě $\boxed{2}$ bude hodnota funkce větší než v bodě $\boxed{1}$, pak minimum funkce zřejmě leží v druhých dvou třetinách intervalu nejistoty. Třetí pokus potom umístíme těsně vedle bodu $\boxed{1}$ a pak můžeme rozhodnout, zda minimum funkce leží v druhé nebo třetí třetině intervalu nejistoty. Tím jsme třemi pokusy zmenšili interval nejistoty na 1/3.

Máme-li k dispozici čtyři pokusy - viz obr. 6.2 pro $N = 4$, pak první dva pokusy - body $\boxed{1}$ a $\boxed{2}$ - umístíme do $\frac{2}{5}$ a $\frac{3}{5}$ intervalu nejistoty . Je-li v bodě $\boxed{2}$ hodnota funkce menší než v bodě $\boxed{1}$, pak minimum funkce zřejmě leží v prvních třech pětinách intervalu nejistoty. Třetí pokus umístíme pak do $\frac{1}{5}$. Pokud bude v bodě $\boxed{3}$ hodnota funkce menší než v bodě $\boxed{2}$, pak minimum funkce zřejmě leží v prvních dvou pětinách intervalu nejistoty. Volbou bodu $\boxed{4}$ těsně vedle bodu $\boxed{3}$ konečně zmenšíme interval nejistoty ještě na polovinu. Minimální interval nejistoty bude tedy při čtyřech pokusech $\frac{1}{5}$ původního intervalu.

Označme tedy počáteční interval nejistoty jako d_1. Pak interval nejistoty po N pokusech bude

$$d_N = \frac{1}{F_N} d_1$$

kde F_N je Fibonacciho číslo.

Počáteční interval nejistoty nechť je interval $[a,\ b]$. Obecný algoritmus volby bodů, ve kterých zjišťujeme hodnotu funkce, je následující:

1. Zvolme $\alpha_1 = a$, $\beta_1 = b$.

2. Vypočtěme pro $i = 1, 2, \ldots, N-1$

$$\begin{aligned} \bar{\alpha}_{i+1} &= \beta_i - \frac{F_{N-i}}{F_{N-i+1}} |\beta_i - \alpha_i| \\ \bar{\beta}_{i+1} &= \alpha_i + \frac{F_{N-i}}{F_{N-i+1}} |\beta_i - \alpha_i| \end{aligned}$$

3. Je-li $f(\bar{\alpha}_{i+1}) \leq f(\bar{\beta}_{i+1})$, pak $\alpha_{i+1} = \alpha_i$, $\beta_{i+1} = \bar{\beta}_{i+1}$, je-li naopak $f(\bar{\alpha}_{i+1}) > f(\bar{\beta}_{i+1})$, pak $\alpha_{i+1} = \bar{\alpha}_{i+1}$, $\beta_{i+1} = \beta_i$. Vždy jedno z čísel α_i, β_i je rovno jednomu z čísel α_{i+1}, β_{i+1}.

□

Tento algoritmus optimálním způsobem redukuje interval nejistoty, ve kterém leží minimum unimodální funkce v závislosti na počtu pokusů.

Nyní prozkoumáme rychlost konvergence Fibonacciovy metody. Platí

$$\frac{d_{N+1}}{d_N} = \frac{F_N}{F_{N+1}}$$

Generátor Fibonacciových čísel je popsán diferenční rovnicí (6.4). Řešení této diferenční rovnice můžeme psát ve tvaru

$$F_N = c_1 \lambda_1^N + c_2 \lambda_2^N$$

kde λ_1 a λ_2 jsou kořeny charakteristické rovnice

$$\lambda^2 = \lambda + 1$$

Jeden kořen $\lambda_1 = \frac{1+\sqrt{5}}{2} \doteq 1.618$ je nestabilní a druhý kořen $\lambda_2 = \frac{1-\sqrt{5}}{2}$ je stabilní. Pak

$$\lim_{N\to\infty} \frac{d_{N+1}}{d_N} = \lim_{N\to\infty} \frac{\lambda_1^N}{\lambda_1^{N+1}} = \frac{1}{\lambda_1} = 0.618$$

Proto Fibonacciova metoda konverguje lineárně s konvergenčním poloměrem 0.618. Magické číslo $\lambda_1 = 1.618$ je známé jako **poměr zlatého řezu**. Tento poměr byl ve starém Řecku považován za nejestetičtější poměr dvou stran obdélníku (např ve stavebnictví).

Nevýhodou Fibonacciovy metody je, že naše strategie od začátku závisí na daném počtu pokusů. Pokud například přidáme ještě jeden pokus, pak vlastně musíme začít úplně znovu. Tuto nevýhodu odstraňuje **metoda zlatého řezu**, která je vlastně Fibonacciova metoda pro $N \to \infty$. Podle metody metody zlatého řezu redukujeme nejistotu každým pokusem o $\frac{1}{\lambda_1} = 0.618$. V metodě zlatého řezu tedy platí

$$d_N = \left(\frac{1}{\lambda_1}\right)^N d_1 = 0.618^N d_1.$$

Proto pokusy vždy volíme v bodech 0.618 a $(1 - 0.618) = 0.382$ intervalu nejistoty. Metoda zlatého řezu konverguje stejně jako Fibonacciova metoda lineárně s konvergenčním poloměrem 0.618. Rozdíl mezi Fibonacciovou metodou a metodou zlatého řezu je patrný pouze pro menší počet pokusů, jak plyne z následující tabulky, která ukazuje redukci intervalu v závislosti na počtu pokusů

N	2	3	4	5	6	8	10
Fibonacciova metoda	0.5	0.333	0.2	0.125	0.077	0.0294	0.0111
Zlatý řez	0.618	0.382	0.236	0.146	0.09	0.0344	0.0131

Fibonacciova metoda nevyžaduje znalost derivací funkce, jejíž extrém hledáme. Za předpokladu unimodality funkce je dokonce optimální. Často zkoumaná funkce je hladká a tuto její vlastnost lze využít pro konstrukci efektivnějších metod. Existuje celá řada těchto metod, které se liší podle toho, kolik hodnot funkce máme k dispozici a zda známe derivace hledané funkce. Všechny tyto metody mají řád konvergence větší než jedna.

6.2.2 Newtonova metoda

Je to jedna z nejstarších metod. Newtonova metoda vyžaduje znalost první a druhé derivace zkoumané funkce. V bodě x_k, který je výsledkem i-té iterace, aproximujeme zkoumanou funkci kvadratickou funkcí $q(x)$, pak

$$f(x) \doteq q(x) = f(x_k) + f'(x_k)(x - x_k) + \frac{1}{2}f''(x_k)(x - x_k)^2 \tag{6.5}$$

Nový odhad x_{k+1} bodu, ve kterém nastává extrém funkce $f(x)$, položíme do bodu, ve kterém nastává extrém aproximační funkce $q(x)$, to je do bodu, ve kterém je nulová první derivace funkce $q(x)$. Pak

$$0 = q'(x_{k+1}) = f'(x_k) + f''(x_k)(x_{k+1} - x_k)$$

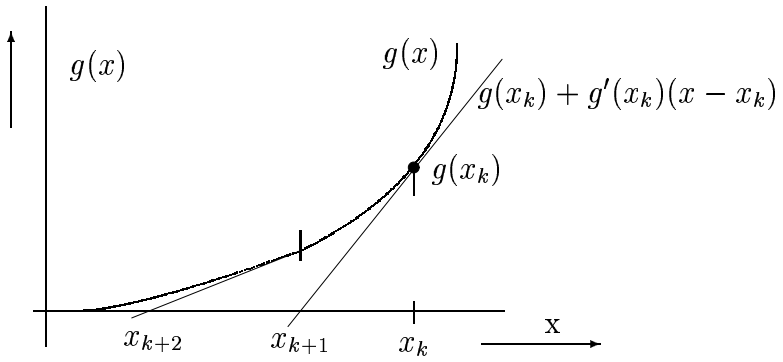
Odtud

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)}. \tag{6.6}$$

Metoda je totožná s iteračním řešením rovnice $g(x) = 0$, (pak $g(x) = f'(x)$). Proto

$$x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$$

Podle předchozího vztahu je funkce $g(x)$ aproximována v bodě x_k tečnou a proto se tato metoda nazývá také **metoda tečen** - viz obr. 6.3.



Obrázek 6.3: Newtonova metoda

Příklad: Nalezněme iteračním postupem $\sqrt{2}$.
Pro hledané $x = \sqrt{2}$ tedy platí $g(x) = x^2 - 2 = 0$. Odtud

$$x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)} = x_k - \frac{x_k^2 - 2}{2x_k}$$

Zvolíme-li počáteční odhad $x_0 = 1$, pak snadno spočteme, že $x_1 = 1.5$, $x_2 = 1.41666$, $x_3 = 1.41421$. Konvergence algoritmu je vskutku velmi rychlá.

□

Globální konvergence Newtonovy metody není zaručena. Pokud ale Newtonova metoda konverguje, pak konverguje s řádem konvergence alespoň dva. Platí následující tvrzení:

Věta:
Nechť funkce $g(x) = f'(x)$ má spojité druhé derivace a nechť ve stacionárním bodě, který označíme x^* platí $g(x^*) = f'(x^*) = 0$, $g'(x^*) = f''(x^*) \neq 0$. Za těchto předpokladů, je-li x_0 dostatečně blízko x^*, posloupnost x_k generovaná Newtonovou metodou konverguje k x^* s řádem konvergence alespoň dva.

□

Důkaz: Nechť pro body ξ, které leží v okolí optimálního bodu x^* platí, že $|g''(\xi)| < r$ a $|g'(\xi)| > s$. Pak v okolí bodu x^* platí

$$g(x^*) = 0 = g(x_k) + g'(x_k)(x^* - x_k) + \frac{1}{2}g''(\xi)(x^* - x_k)^2$$

kde ξ je nějaký bod mezi x^* a x_k. Odtud

$$x^* = x_k - \frac{g(x_k)}{g'(x_k)} - \frac{1}{2}\frac{g''(\xi)}{g'(x_k)}(x^* - x_k)^2.$$

Z Newtonovy metody plyne, že první dva členy na pravé straně předchozího vztahu jsou rovny x_{k+1}. Proto platí

$$x^* - x_{k+1} = -\frac{1}{2}\frac{g''(\xi)}{g'(x_k)}(x^* - x_k)^2$$

Protože jsme dle předpokladu omezili hodnoty prvních a druhých derivací funkce $g(x)$ v okolí bodu x^*, pak v okolí tohoto bodu platí

$$|x_{k+1} - x^*| \leq \frac{r}{2s}|x_k - x^*|^2$$

Odtud dle definice řádu konvergence plyne, že Newtonova metoda vskutku konverguje s řádem konvergence alespoň dva. To znamená, že blízko řešení každá iterace zpřesňuje výsledek o jeden řád. □

Metoda Regula falsi

Newtonova metoda vyžadovala znalost první a druhé derivace zkoumané funkce. Metoda Regula falsi aproximuje druhou derivaci funkce pomocí první diference jejích prvních derivací

$$f''(x_k) \doteq \frac{f'(x_k) - f'(x_{k-1})}{x_k - x_{k-1}}$$

Potom iterační předpis metody Regula falsi je

$$x_{k+1} = x_k - f'(x_k)\frac{x_k - x_{k-1}}{f'(x_k) - f'(x_{k-1})}$$

Věta:
Za stejných předpokladů jako u Newtonovy metody je řád konvergence metody Regula falsi roven 1.618. □

Protože důkaz řádu konvergence metody Regula falsi je zajímavý, uvedeme jej podrobně.

Důkaz: Budeme používat jednodušší značení $g_k = g(x_k) = f'(x_k)$, $g^* = g(x^*) = f'(x^*)$ (ale v minimu samozřejmě $g(x^*) = 0$) a obdobně $g_{k-1} = g(x_{k-1})$. Zřejmě platí

$$\begin{aligned}
x_{k+1} - x^* &= x_k - x^* - g_k\frac{x_k - x_{k-1}}{g_k - g_{k-1}} \\
&= (x_k - x^*)\left[1 - \frac{g_k\frac{1}{x_k - x^*}}{\frac{g_k - g_{k-1}}{x_k - x_{k-1}}}\right] \\
&= (x_k - x^*)\left[\frac{\frac{g_k - g_{k-1}}{x_k - x_{k-1}} - \frac{g^* - g_k}{x^* - x_k}}{\frac{g_k - g_{k-1}}{x_k - x_{k-1}}}\right] \\
&= (x_k - x^*)(x_{k-1} - x^*)\left[\frac{\frac{\frac{g_k - g_{k-1}}{x_k - x_{k-1}} - \frac{g^* - g_k}{x^* - x_k}}{x_{k-1} - x^*}}{\frac{g_k - g_{k-1}}{x_k - x_{k-1}}}\right]
\end{aligned}$$

Při tom platí

$$\frac{g_k - g_{k-1}}{x_k - x_{k-1}} = g'(\xi_k), \qquad \frac{\frac{g_k - g_{k-1}}{x_k - x_{k-1}} - \frac{g^* - g_k}{x^* - x_k}}{x_{k-1} - x^*} = \frac{1}{2} g''(\eta_k)$$

kde ξ_k je konvexní kombinace bodů x_k, x_{k-1} a η_k je konvexní kombinace bodů x_k, x_{k-1}, x^*. Pak platí

$$x_{k+1} - x^* = \frac{g''(\eta_k)}{2g'(\xi_k)}(x_k - x^*)(x_{k-1} - x^*)$$

Z předchozího vztahu plyne, že metoda Regula falsi konverguje, startujeme-li dostatečně blízko optima x^*.

Abychom určili řád konvergence, pak pro velké k můžeme psát

$$x_{k+1} - x^* = M(x_k - x^*)(x_{k-1} - x^*)$$

kde

$$M = \frac{g''(x^*)}{2g'(x^*)}$$

Potom pro $\varepsilon_k = x_k - x^*$ platí

$$\varepsilon_{k+1} = M\varepsilon_k\varepsilon_{k-1}$$

Logaritmováním předchozí rovnice dostaneme pro $z_k = \log M\varepsilon_k$ diferenční rovnici pro z_k

$$z_{k+1} = z_k + z_{k-1},$$

což je Fibonacciho rovnice (6.4). Pro ni platí

$$\lim_{k\to\infty} \frac{z_{k+1}}{z_k} = \lambda_1$$

kde $\lambda_1 = 1.618$. Odtud

$$z_{k+1} - \lambda_1 z_k \to 0$$

Tudíž

$$\log M\varepsilon_{k+1} - \lambda_1 \log M\varepsilon_k \to 0, \qquad \text{a proto} \qquad \log \frac{M\varepsilon_{k+1}}{(M\varepsilon_k)^{\lambda_1}} \to 0$$

Proto konečně platí

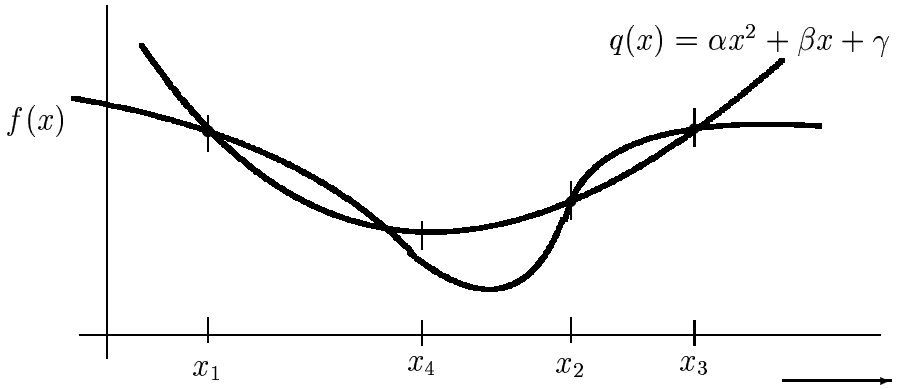
$$\frac{\varepsilon_{k+1}}{(\varepsilon_k)^{\lambda_1}} \to M^{(\lambda_1 - 1)} = konst.$$

Tím je dokázán řád konvergence $p = \lambda_1 = 1.618$ metody Regula falsi.

6.2.3 Metoda kvadratické interpolace

Výhodou této metody je, že nevyžaduje znalost derivací zkoumané funkce. Vycházíme ze znalosti funkčních hodnot ve třech bodech x_1, x_2 a x_3. Body $f_1 = f(x_1)$, $f_2 = f(x_2)$, $f_3 = f(x_3)$ proložíme kvadratickou funkci - viz obr. 6.4.

$$q(x) = f_1 \frac{(x - x_2)(x - x_3)}{(x_1 - x_2)(x_1 - x_3)} + f_2 \frac{(x - x_1)(x - x_3)}{(x_2 - x_1)(x_2 - x_3)} + f_3 \frac{(x - x_1)(x - x_2)}{(x_3 - x_1)(x_3 - x_2)}$$



Obrázek 6.4: Metoda kvadratické interpolace

Nyní určíme bod x_4, ve kterém je nulová první derivace funkce $q(x)$. Po úpravě dostaneme

$$x_4 = \frac{1}{2}\frac{(x_2^2 - x_3^2)f(x_1) + (x_3^2 - x_1^2)f(x_2) + (x_1^2 - x_2^2)f(x_3)}{(x_2 - x_3)f(x_1) + (x_3 - x_1)f(x_2) + (x_1 - x_2)f(x_3)}$$

Ze čtyř bodů, které nyní máme k dispozici vynecháme ten, ve kterém je zkoumaná funkce maximální a ze zbylých tří bodů opakujeme algoritmus.

Metoda kvadratické interpolace konverguje s řádem konvergence 1.3.

Modifikace této metody podle Daviese, Swanna a Campeyho spočívá v tom, že bod x_2 se volí uprostřed bodů x_1 a x_3 Pak pro minimum kvadratické aproximace platí

$$x_4 = x_2 + \frac{\Delta x}{2}\frac{f(x_1) - f(x_3)}{f(x_1) - 2f(x_2) + f(x_3)},$$

kde $\Delta x = x_2 - x_1$. V dalším kroku se vypočtený bod x_4 zvolí středem nového intervalu a krajní body se určí ve stejné vzdálenosti na obě strany od bodu x_4 a výpočet se opakuje.

Metoda kubické interpolace

Metoda kubické aproximace vychází ze znalosti funkčních hodnot spolu s derivacemi ve dvou bodech zkoumané funkce. Známe tedy v bodech x_k a x_{k-1} hodnoty $f(x_k)$, $f'(x_k)$ a $f(x_{k-1})$, $f'(x_{k-1})$. Volíme aproximační polynom třetího řádu, který má funkční hodnoty a derivace v bodech x_k a x_{k-1} stejné jako zkoumaná funkce.

Minimum aproximačního polynomu $q(x) = ax^3 + bx^2 + cx + d$ nastává v bodě

$$x = \frac{-2b + \sqrt{4b^2 - 12ac}}{6a} \quad \text{pro} \quad 4b^2 - 12ac \geq 0$$

Minimum kubického interpolačního polynomu určené z $f_k = f(x_k)$, $f'_k = f'(x_k)$ a $f_{k-1} = f(x_{k-1})$, $f'_{k-1} = f'(x_{k-1})$ je v bodě

$$x_{k+1} = x_k - g$$

kde

$$\begin{aligned} e &= x_k - x_{k-1} \\ z &= 3\frac{f_{k-1} - f_k}{e} + f'_{k-1} + f'_k \end{aligned}, \qquad \begin{aligned} w &= \sqrt{z^2 - f'_{k-1} f'_k} \\ g &= e\frac{f'_k + w - z}{f_k - f_{k-1} + 2w} \end{aligned}.$$

Řád konvergence metody kubické interpolace je pouze dva. Globální konvergence při kvadratické nebo kubické aproximaci je zaručena, pokud $f(x_{k+1}) < f(x_k)$. Aproximační funkce mohou být různé, jako aproximační funkce můžeme volit také např. spliny.

6.2.4 Nepřesné algoritmy jednorozměrové optimalizace

Protože algoritmy jednorozměrového hledání jsou často dílčími algoritmy vícerozměrové optimalizace, je jim v literatuře věnována velká pozornost. Je zřejmé, že iteračním postupem, který musíme ukončit po konečném počtu iterací, nedosáhneme přesně hledané minimum. Je-li takový algoritmus součástí nějakého složitějšího algoritmu, je třeba zaručit, že se celý algoritmus nezhroutí, když použijeme nepřesný dílčí algoritmus jednorozměrového hledání.

Nejprve uvedeme podmínku, kdy je jednorozměrový algoritmus uzavřený. Pro danou funkci $f(\mathbf{x})$ máme dány dva vektory - počáteční vektor $\mathbf{x}$ a směr $\mathbf{s}$. Předpokládáme, že z bodu $\mathbf{x}$ provádíme jednorozměrové hledání na polopřímce ve směru $\mathbf{s}$. Dále předpokládáme, že minimum v daném směru skutečně existuje.

Definujme zobrazení z prostoru R^{2n} do prostoru R^n

$$S(\mathbf{x}, \mathbf{s}) = \left\{ \mathbf{y} \ : \ \mathbf{y} = \mathbf{x} + \alpha\mathbf{s} \quad \text{pro nějaké } \alpha \geq 0, \quad f(\mathbf{y}) = \min_{0 \leq \alpha \leq \infty} f(\mathbf{x} + \alpha\mathbf{s}) \right\}$$

Platí, že předchozí zobrazení je uzavřené v $(\mathbf{x}, \mathbf{s})$ je-li funkce f spojitá a $\mathbf{s} \neq 0$.

Dále uvedeme několik kritérií pro ukončení jednorozměrového hledání.

Procentní test

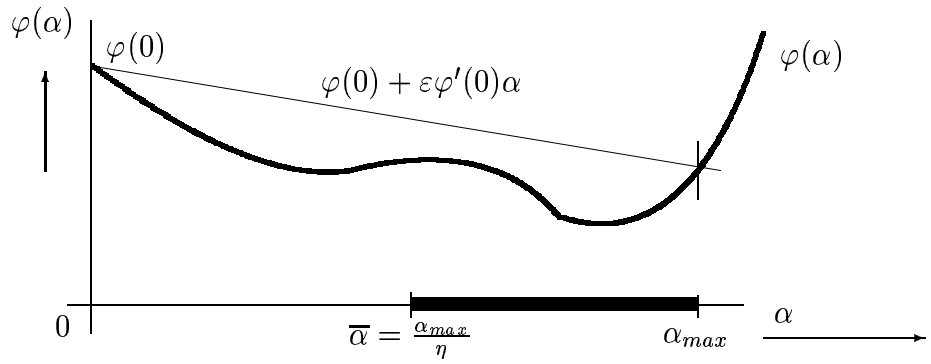
Při tomto testu určujeme hledaný parametr α pouze s určitou přesností vzhledem k jeho správné hodnotě. Volíme např. konstantu c, $0 < c < 1$ (vhodná hodnota je $c = 0.1$) a parametr α pro jednorozměrovou optimalizaci splňuje omezení $|\alpha - \alpha^*| \leq c\alpha^*$, kde α^* je jeho hodnota minimalizující funkci ve směru $\mathbf{s}$.

Tento algoritmus je uzavřený. Toto kritérium můžeme použít při kvadratické nebo kubické interpolaci.

Armiův test

Základní myšlenka tohoto testu je, že při jednorozměrové optimalizaci hledané α nesmí být ani příliš malé, ani příliš velké. Definujme funkci

$$\varphi(\alpha) = f(\mathbf{x}_k + \alpha\mathbf{s}_k)$$



Obrázek 6.5: Armiův test

Uvažujme nyní lineární funkci $\varphi(0) + \varepsilon\varphi'(0)\alpha$ pro určité ε z intervalu $0 < \varepsilon < 1$ (vhodná volba je $\varepsilon = 0.2$). Tato funkce je v obr. 6.5 jako polopřímka vycházející z bodu $\varphi(0)$ se sklonem $\varepsilon\varphi'(0)$. Hodnota α není příliš velká, když

$$\varphi(\alpha) \leq \varphi(0) + \varepsilon\varphi'(0)\alpha \tag{6.7}$$

Aby α nebylo naopak příliš malé, volíme $\eta > 1$ (vhodná volba je $\eta = 2$) a pak parametr α není považován za příliš malý, je-li

$$\varphi(\eta\alpha) > \varphi(0) + \varepsilon\varphi'(0)\eta\alpha$$

To znamená, že když α zvětšíme η-krát, pak už není splněn test (6.7). Oblast přijatelných α je v obr. 6.5 znázorněna tučně.

Při Armiovu testu vycházíme z libovolného α. Splňuje-li test (6.7), zvětšujeme ho η-kráte tak dlouho, až test (6.7) není splněn. Pak vhodné α je to největší (předposlední), které test (6.7) splnilo. Pokud prvně zvolené α nesplňuje test (6.7), pak je opakovaně děleno η tak dlouho, až nalezneme takové α, které test (6.7) splňuje.

Goldsteinův test

Jiný test pro přesnost jednorozměrové optimalizace je Goldsteinův test. Vycházíme opět z Armiova testu (6.7), který určuje, zda hledané α není příliš velké. Zde volíme ε v intervalu $1 < \varepsilon < 0.5$. Hodnota hledaného parametru α není naopak považována za příliš malou, je-li splněn Goldsteinův test

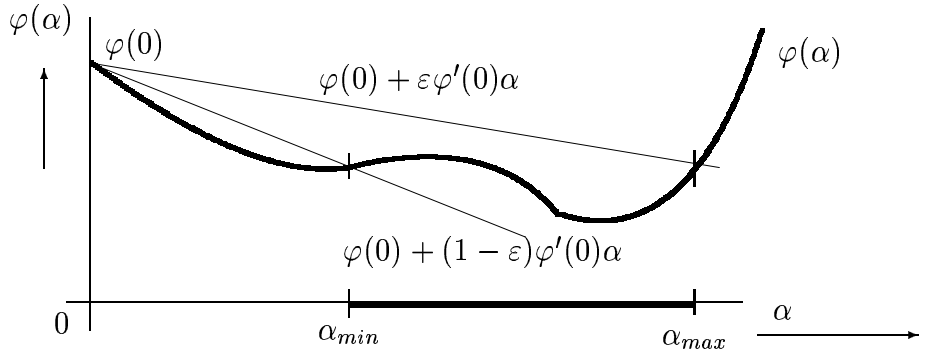
$$\varphi(\alpha) > \varphi(0) + (1 - \varepsilon)\varphi'(0)\alpha$$

To znamená, že $\varphi(\alpha)$ musí ležet nad přímkou se směrnicí $(1 - \varepsilon)\varphi'(0)$ - viz obr. 6.6. Oblast přijatelných α je v obr. 6.6 znázorněna tučně. Goldsteinův test je uzavřený jednorozměrový algoritmus.

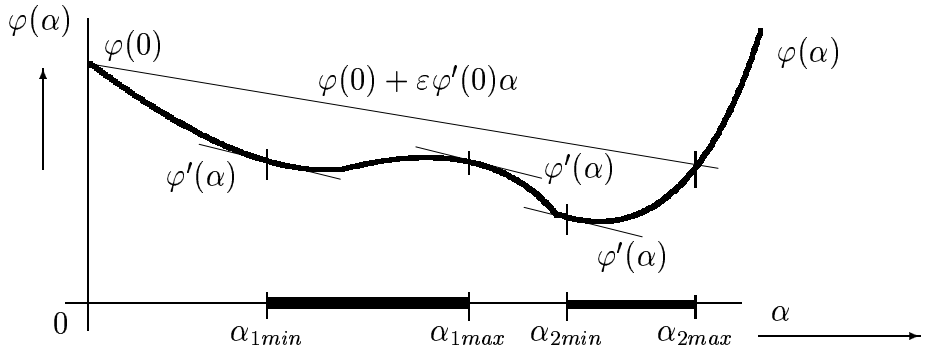
Wolfeho test

Tento test je výhodný, můžeme-li snadno určit derivace kriteriální funkce. Zde je ε zvoleno v intervalu $1 < \varepsilon < 0.5$ a α pak musí vyhovovat Armiovu testu (6.7) a také

$$\varphi'(\alpha) \geq (1 - \varepsilon)\varphi'(0)$$



Obrázek 6.6: Goldsteinův test



Obrázek 6.7: Wolfeho test

Tento test je znázorněn na obr. 6.7. Oblast přijatelných α je také v obr. 6.7 znázorněna tučně.

6.3 Numerické metody bez omezení

V této kapitole uvedeme některé algoritmy hledání minima nelineární mnoharozměrové funkce $f(\mathbf{x})$. Nejprve stručně uvedeme metody, které nevyužívají derivace zkoumané funkce. Tyto metody se nazývají **metody přímého hledání** nebo **komparativní metody**.

V dalších odstavcích pojednáme o více používaných metodách, které využívají derivace kriteriální funkce.

6.3.1 Komparativní metody

Gaussova-Seidlova metoda

Tato jednoduchá metoda je v podstatě **metoda cyklické záměny proměnných**. Jak sám název napovídá vícerozměrovou optimalizaci převedeme na posloupnost jednorozměrových optimalizací ve směru jednotlivých souřadnic.

To znamená, že z výchozího bodu $\mathbf{x}$ provádíme minimalizaci ve směru první souřadnice x_1. Po nalezení lokálního extrému funkce $f(\mathbf{x})$ ve směru $\mathbf{s}_1 = [1, 0, \ldots, 0]^T$ pokračujeme

ve směru druhé souřadnice atd..

Gaussova-Seidlova metoda je jednoduchá, nevýhodou je pomalý výpočet. Při tom není ani zaručena konvergence metody. Výhodou je možnost paralelního výpočtu.

Metody přímé komparace

V této jednoduché metodě volíme počáteční bod $\mathbf{x}^{(0)}$ a přírůstek $\Delta\mathbf{x}$. Nalezneme bod $\mathbf{x}^{(1)}$ tak, že k první souřadnici počátečního vektoru $\mathbf{x}^{(0)}$ přičteme Δx_1. Není-li tento bod úspěšný, pak Δx_1 odečteme. Pokud ani tento bod není úspěšný, pak ponecháme původní bod. Toto provedeme postupně ve všech souřadnicích. Dostaneme pak posloupnost bodů

$$\mathbf{x}^{(1)} = \begin{bmatrix} x_1^{(0)} \pm \Delta x_1 \\ x_2^{(0)} \\ \vdots \\ x_n^{(0)} \end{bmatrix}, \quad \mathbf{x}^{(2)} = \begin{bmatrix} x_1^{(1)} \\ x_2^{(1)} \pm \Delta x_2 \\ \vdots \\ x_n^{(1)} \end{bmatrix}, \cdots \mathbf{x}^{(n)} = \begin{bmatrix} x_1^{(n-1)} \\ x_2^{(n-1)} \\ \vdots \\ x_n^{(n-1)} \pm \Delta x_n \end{bmatrix},$$

kde při neúspěchu může být přírůstek Δx_i i nulový. Tento postup můžeme opakovat, nebo můžeme provést modifikaci metody pomocí vektoru úspěšného směru
$\mathbf{s} = [\Delta x_1,\ -\Delta x_2,\ 0, \ldots, \Delta x_n]$, kde znaménka, případně nuly, jsou podle úspěšného či neúspěšného směru.

Potom provedeme expanzi ve směru vektoru $\mathbf{s}$. To znamená, že postupně pro $\alpha = 1, 2, \ldots$ počítáme $f(\mathbf{x} + \alpha\mathbf{s})$ tak dlouho, pokud nastává pokles kriteriální funkce. Pokud přírůstek $\Delta\mathbf{x}$ je příliš velký, pak provedeme redukci, to znamená, že $\Delta\mathbf{x}$ zmenšíme na polovinu. Po provedené minimalizaci ve směru $\mathbf{s}$ dostaneme nový bod a pro něho zjistíme nový směr a opakujeme minimalizaci v novém směru.

Metoda pravidelného a flexibilního simplexu

Tato jednoduchá metoda startuje vytvořením simplexu v prostoru $\mathbf{X}$.
Simplex je konvexní polyedr tvořený jako konvexní obal $n+1$ bodů $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(n+1)}$. Simplex v rovině je trojúhelník, v třírozměrném prostoru je to čtyřstěn atd.. Z těchto bodů vynecháním nejhoršího a konstrukcí nového bodu vytvoříme nový simplex, to je další iteraci simplexové metody. Simplexová metoda je metoda heuristická, která je ale velmi jednoduchá, názorná a snadno implementovatelná.

Pravidelný simplex vytvoříme z následujících bodů

$$\mathbf{x}^{(1)} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \quad \mathbf{x}^{(2)} = \mathbf{x}^{(1)} + \begin{bmatrix} e \\ d \\ \vdots \\ d \end{bmatrix}, \quad \cdots \quad \mathbf{x}^{(n+1)} = \mathbf{x}^{(1)} + \begin{bmatrix} d \\ d \\ \vdots \\ e \end{bmatrix},$$

kde $\mathbf{x}^{(1)}$ je libovolně zvolený bod v prostoru R^n a konstanty d a e určíme z následujících vztahů

$$d = a\frac{\sqrt{(n+1)}+1}{n\sqrt{2}}, \qquad e = d + \frac{a}{\sqrt{2}},$$

kde a je délka hrany pravidelného simplexu. Má-li minimalizovaná funkce největší (to znamená nejhorší) hodnotu v bodě $\mathbf{x}^{(k)}$

$$k = \arg\max_i f(\mathbf{x}^{(i)})$$

nahradíme tento bod v množině vrcholů simplexu $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n+1)}$ novým bodem

$$\widehat{\mathbf{x}}^{(k)} = \mathbf{x}^{(k)} + 2\left[\mathbf{c} - \mathbf{x}^{(k)}\right] = 2\mathbf{c} - \mathbf{x}^{(k)}$$

kde

$$\mathbf{c} = \frac{1}{n}\left(-\mathbf{x}^{(k)} + \sum_{i=1}^{n+1} \mathbf{x}^{(i)}\right).$$

Bod $\mathbf{c}$ je těžiště bodů $\mathbf{x}^{(i)}$ s vynecháním bodu nejhoršího $\mathbf{x}^{(k)}$. Bod $\widehat{\mathbf{x}}^{(k)}$ je v podstatě reflexní bod, ležící na polopřímce vycházející z bodu $\mathbf{x}^{(k)}$ a procházející středem $\mathbf{c}$. Tento bod je umístěn symetricky k nejhoršímu bodu $\mathbf{x}^{(k)}$ vzhledem k $\mathbf{c}$. Tím dostaneme opět množinu $n+1$ bodů, které tvoří nový simplex.

Rychlost a přesnost nalezení optimálního bodu záleží na velikosti simplexu. Čím větší je simplex, tím se rychleji blížíme k optimu. Přesnost výpočtu naopak vyžaduje malou velikost simplexu. Proto při výpočtu je třeba měnit délku hrany simplexu. Podle Neldera a Meada je to možno dělat pomocí expanze, kontrakce nebo redukce základního simplexu. Dostaneme tak metodu **proměnného (flexibilního) simplexu**.

- **Expanze:** Je-li v novém bodě $\widehat{\mathbf{x}}^{(k)}$, který vznikl reflexí, minimální hodnota kriteriální funkce
$$f(\widehat{\mathbf{x}}^{(k)}) \leq \min_{i=1,\ldots,n+1} \left\{f(\mathbf{x}^{(i)})\right\},$$
pak směr reflexe je úspěšný a proto provedeme expanzi simplexu ve směru reflexe a nový vrchol simplexu volíme v bodě
$$\widehat{\mathbf{x}}^{(k)} = \mathbf{c} + \alpha[\mathbf{c} - \mathbf{x}^{(k)}],$$
kde α je větší než jedna, obvykle 2 nebo 3.

- **Kontrakce:** Je-li naopak novém bodě $\widehat{\mathbf{x}}^{(k)}$, který vznikl reflexí, hodnota kriteriální funkce větší než v ostatních bodech simplexu
$$f(\widehat{\mathbf{x}}^{(k)}) > \max_{i=1,\ldots,n+1,\ i\neq k} \left\{f(\mathbf{x}^{(i)})\right\},$$
je simplex zřejmě příliš velký, neboť starý bod i bod vzniklý reflexí nejsou úspěšné. Proto provedeme kontrakci simplexu podle vztahu
$$\widehat{\mathbf{x}}^{(k)} = \mathbf{c} + \beta[\mathbf{c} - \mathbf{x}^{(k)}],$$
kde β je menší než jedna, obvykle $\beta = 0.5$. Tím zkrátíme směr postupu v neúspěšném směru.

- **Redukce:** Celkovou redukci velikosti simplexu provedeme následujícím postupem. Z bodů tvořících simplex vybereme nejvýhodnější bod $\mathbf{x}^{(j)}$ podle vztahu

$$j = \arg\min_i f(\mathbf{x}^{(i)})$$

Vrcholy nového, redukovaného, simplexu budou rovny

$$\mathbf{x}_{new}^{(i)} = \mathbf{x}^{(j)} + \gamma[\mathbf{x}^{(i)} - \mathbf{x}^{(j)}],$$

kde koeficient redukce $\gamma < 1$ volíme obvykle rovný 0.5.

Iterační výpočet podle simplexového algoritmu s proměnným simplexem ukončíme, když

$$\sum_{i=1, i\neq k}^{n+1} \left(f(\mathbf{x}^{(i)}) - f(\widehat{\mathbf{x}}^{(k)}) \right) \leq \varepsilon,$$

kde ε je zvolená přesnost nalezení minima.

6.3.2 Gradientní metody

Gradientní metoda je jednou z nejstarších a nejoblíbenějších numerických metod. Směr hledání volíme úměrný gradientu zkoumané funkce. Při tom gradient funkce $f(\mathbf{x})$ v bodě $\mathbf{x}_k$, který budeme značit $\mathbf{g}(\mathbf{x}_k) = \text{grad } f(\mathbf{x}_k)$ je sloupcový vektor ($\mathbf{g}(\mathbf{x}_k) = \nabla^T f(\mathbf{x}_k)$). Často budeme používat pro gradient $\mathbf{g}(\mathbf{x}_k)$ jednodušší značení $\mathbf{g}_k$. Při minimalizaci dané funkce $f(\mathbf{x})$ je iterační algoritmus

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha \mathbf{g}_k$$

kde α je nezáporná konstanta.

Podle toho, jak volíme konstantu α, rozlišujeme různé typy gradientních metod:

- Nalezneme α_k optimální. Tato metoda se nazývá **metoda nejrychlejšího sestupu (steepest descent)**.
- Konstanta α_k je závislá na velikosti gradientu, provádíme vlastně jednorozměrovou optimalizaci ve směru gradientu.
- Volíme pevné $\alpha_k = \alpha$, dostaneme potom gradientní metodu s pevným krokem. Tím nemáme vůbec zaručenou konvergenci algoritmu a proto se tato varianta již nepoužívá.

Optimální α_k určíme z podmínky

$$\frac{\partial f(\mathbf{x}_k - \alpha_k \mathbf{g}_k)}{\partial \alpha_k} = 0 \tag{6.8}$$

Z předchozího vztahu vypočteme optimální α_k, když budeme funkci $f(\mathbf{x})$ aproximovat kvadratickou funkcí

$$f(\mathbf{x}) = f(\mathbf{x}^*) + \frac{1}{2}(\mathbf{x} - \mathbf{x}^*)^T \mathbf{H} (\mathbf{x} - \mathbf{x}^*), \tag{6.9}$$

kde matice $\mathbf{H}$ je pozitivně definitní a minimum funkce $f(\mathbf{x})$ je v bodě $\mathbf{x}^*$. První a druhé derivace funkce $f(\mathbf{x})$ jsou rovny

$$\left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}\right)^T = \mathbf{H}(\mathbf{x} - \mathbf{x}^*) = \mathbf{g}(\mathbf{x}), \qquad \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x}^2} = \mathbf{H}$$

Za proměnnou $\mathbf{x}$ dosadíme další iteraci $\mathbf{x} = \mathbf{x}_k - \alpha_k \mathbf{g}_k = \mathbf{x}_{k+1}$. Pak

$$f(\mathbf{x}_{k+1}) = f(\mathbf{x}^*) + \frac{1}{2}(\mathbf{x}_k - \alpha_k \mathbf{g}_k - \mathbf{x}^*)^T \ \mathbf{H}\ (\mathbf{x}_k - \alpha_k \mathbf{g}_k - \mathbf{x}^*) = \Phi(\alpha_k),$$

kde funkce $\Phi(\alpha_k)$ je funkcí α_k pro pevné $\mathbf{x}_k$ a $\mathbf{x}^*$. Pro optimální α_k, musí být derivace funkce $\Phi(\alpha_k)$ podle α_k rovna nule, pak

$$\frac{\partial \Phi(\alpha_k)}{\partial \alpha_k} = \alpha_k \mathbf{g}_k^T \mathbf{H} \mathbf{g}_k - \mathbf{g}_k^T \mathbf{H}(\mathbf{x}_k - \mathbf{x}^*) = 0$$

Odtud

$$\alpha_k = \frac{\mathbf{g}_k^T \mathbf{H}(\mathbf{x}_k - \mathbf{x}^*)}{\mathbf{g}_k^T \mathbf{H} \mathbf{g}_k} = \frac{\mathbf{g}_k^T \mathbf{g}_k}{\mathbf{g}_k^T \mathbf{H} \mathbf{g}_k} \tag{6.10}$$

Algoritmus metody nejrychlejšího sestupu je tedy

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \frac{\mathbf{g}_k^T \mathbf{g}_k}{\mathbf{g}_k^T \mathbf{H} \mathbf{g}_k} \mathbf{g}_k \tag{6.11}$$

□

Optimální α_k můžeme také hledat pomocí algoritmů jednorozměrového hledání jako

$$\alpha_k = \arg \min_{\alpha \geq 0} f(\mathbf{x}_k - \alpha \mathbf{g}_k) \tag{6.12}$$

V reálných problémech nenalezneme přesně optimální hodnotu α_k a ani to není nutné. Je pouze třeba zaručit celkovou konvergenci gradientního algoritmu. Proto v každé iteraci musí nastat pokles hodnotící funkce $f(\mathbf{x}_{k+1}) \leq f(\mathbf{x}_k)$ a zároveň parametr α_k nesmí být příliš malý, aby byla zaručena celková konvergence algoritmu.

Jedna z možností je omezení hledání optimálního α_k na určitý interval $\alpha \in [0,\ s]$, pro zvolený skalár s. Pak tedy

$$\alpha_k = \arg \min_{\alpha \in [0,\ s]} f(\mathbf{x}_k - \alpha \mathbf{g}_k)$$

Abychom se vyhnuli zdlouhavým výpočtům, zavádí se určitá pravidla založená na postupné redukci kroků.

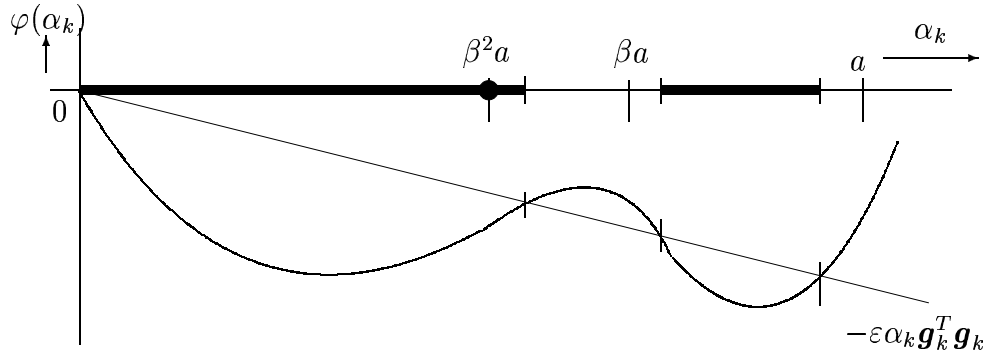
Nejjednodušší je volit pevné číslo $a > 0$ a α_k volit rovné a tak dlouho, dokud $f(\mathbf{x}_k - \alpha_k \mathbf{g}_k) < f(\mathbf{x}_k)$. Pokud nerovnost pro nějaké k přestane platit, pak redukujeme $\alpha_k = \beta a$, $(0 < \beta < 1)$, případně redukujeme krok opakovaně. Tato jednoduchá metoda ale nezaručuje konvergenci algoritmu - redukce kritéria v každém kroku není dostatečná, aby zaručovala konvergenci algoritmu.

Abychom se vyhnuli potížím s konvergencí používá se **Armiovo pravidlo**. Volíme tedy pevné skaláry a, $\beta \in (0,\ 1)$, $\varepsilon \in (0,\ 1)$. Položíme $\alpha_k = \beta^{m_k}\ a$, kde m_k je první celé číslo m, pro které platí

$$\varphi(\alpha_k) = f(\mathbf{x}_k - \alpha_k \mathbf{g}_k) - f(\mathbf{x}_k) \leq -\varepsilon \alpha_k \mathbf{g}_k^T \mathbf{g}_k$$

Volíme tedy α_k postupně rovné $\beta^0 a$, $\beta^1 a$ až $\beta^{m_k} a$. Obvykle volíme $a = 1$, $\beta \in [0.5, 0.1]$, $\varepsilon \in [10^{-5}, 10^{-1}]$. Tímto způsobem nejsme spokojeni pouze s redukcí, ale redukce kritéria musí být dostatečná - úměrná gradientu.

V předchozí rovnici jsme označili $\varphi(\alpha_k) = f(\mathbf{x}_k - \alpha_k \mathbf{g}_k) - f(\mathbf{x}_k)$, pak platí $\varphi(0) = 0$, $\varphi'(0) = \dfrac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \dfrac{d\mathbf{x}}{d\alpha} = -\mathbf{g}_k^T \mathbf{g}_k$. Množina přijatelných řešení podle Armiovy metody je v obr. 6.8 vyznačena tučně. Množina přijatelných řešení netvoří interval, ale α_k ve tvaru $\alpha_k = \beta^i a$ vždy nalezneme. Přijaté řešení je v obr. 6.8 vyznačeno černou tečkou.



Obrázek 6.8: Armiovo pravidlo

Pro nalezení α_k je možno použít i Goldsteinovo pravidlo nebo metodu kvadratické či kubické interpolace.

Gradientní metoda nevede k optimu přímo. Směry hledání pomocí gradientní metody nejrychlejšího sestupu jsou na sebe kolmé. Platí tedy $\mathbf{g}_{k+1}^T \mathbf{g}_k = 0$. Platnost předchozího vztahu prokážeme pro kvadratické funkce (6.9). Zde platí

$$\mathbf{g}_{k+1}^T \mathbf{g}_k = \left[\mathbf{H}\left(\mathbf{x}_k - \alpha_k \mathbf{g}_k - \mathbf{x}^*\right)\right]^T \mathbf{H}\left(\mathbf{x}_k - \mathbf{x}^*\right)$$

Po úpravě předchozího výrazu dostaneme

$$\left(\mathbf{x}_k - \mathbf{x}^*\right)^T \mathbf{H}^T \mathbf{H}\left(\mathbf{x}_k - \mathbf{x}^*\right) - \frac{\mathbf{g}_k^T \mathbf{g}_k}{\mathbf{g}_k^T \mathbf{H} \mathbf{g}_k} \mathbf{g}_k^T \mathbf{H}^T \mathbf{H}\left(\mathbf{x}_k - \mathbf{x}^*\right) = 0.$$

Předchozí výraz je vskutku roven nule, neboť platí $\mathbf{g}(\mathbf{x}) = \mathbf{H}\left(\mathbf{x} - \mathbf{x}^*\right)$.

Nyní se budeme věnovat problému konvergence gradientního algoritmu. Algoritmus gradientní metody ($\mathbf{x}_{k+1} = P(\mathbf{x}_k)$) můžeme dekomponovat do dvou na sebe navazujících algoritmů $P = S \,.\, G$. Algoritmus $G \;:\; E^n \to E^{2n}$ definovaný $G(\mathbf{x}) = (\mathbf{x}; -\mathbf{g}(\mathbf{x}))$ nám dá počáteční bod a směr hledání. Algoritmus S je algoritmus jednorozměrového hledání ve směru negativního gradientu. Je to tedy zobrazení $S : E^{2n} \to E^n$

$$S(\mathbf{x},\ \mathbf{s}) = \left\{\mathbf{y} \;:\; \mathbf{y} = \mathbf{x} + \alpha \mathbf{s} :\ \alpha \geq 0,\ f(\mathbf{y}) = \min_{\alpha \geq 0} f(\mathbf{x} + \alpha \mathbf{s})\right\}$$

Tento algoritmus je uzavřený pokud $\mathbf{s} \neq 0$ a algoritmus G je spojitý. Proto je algoritmus P uzavřený. Algoritmus generuje klesající posloupnost (pro $\mathbf{g}(\mathbf{x}_k) \neq 0$). Protože jsou splněny předpoklady věty o globální konvergenci, algoritmus gradientní metody konverguje.

Vyšetřeme nyní rychlost konvergence gradientní metody. Uvažujme kvadratický případ (funkce $f(\mathbf{x})$ je dle (6.9)) a metodu nejrychlejšího sestupu (6.11). Platí

$$\frac{f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*)}{f(\mathbf{x}_k) - f(\mathbf{x}^*)} = \frac{(\mathbf{x}_k - \alpha_k \mathbf{g}_k - \mathbf{x}^*)^T \mathbf{H} (\mathbf{x}_k - \alpha_k \mathbf{g}_k - \mathbf{x}^*)}{(\mathbf{x}_k - \mathbf{x}^*)^T \mathbf{H} (\mathbf{x}_k - \mathbf{x}^*)}$$

označme $\mathbf{x}_k - \mathbf{x}^* = \mathbf{y}$, $\mathbf{g}_k = \mathbf{g}$, $\alpha_k = \alpha$, pak

$$\frac{f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*)}{f(\mathbf{x}_k) - f(\mathbf{x}^*)} = \frac{(\mathbf{y} - \alpha\mathbf{g})^T \mathbf{H} (\mathbf{y} - \alpha\mathbf{g})}{(\mathbf{y})^T \mathbf{H} (\mathbf{y})} = 1 + \frac{\alpha^2 \mathbf{g}^T\mathbf{H}\mathbf{g} - 2\alpha \mathbf{g}^T\mathbf{H}\mathbf{y}}{\mathbf{y}^T\mathbf{H}\mathbf{y}}$$

Po dosazení za optimální α dostaneme

$$\frac{f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*)}{f(\mathbf{x}_k) - f(\mathbf{x}^*)} = 1 + \frac{\frac{\mathbf{g}^T\mathbf{g}}{\mathbf{g}^T\mathbf{H}\mathbf{g}} \frac{\mathbf{g}^T\mathbf{g}}{\mathbf{g}^T\mathbf{H}\mathbf{g}} \mathbf{g}^T\mathbf{H}\mathbf{g} - 2 \frac{\mathbf{g}^T\mathbf{g}}{\mathbf{g}^T\mathbf{H}\mathbf{g}} \mathbf{g}^T\mathbf{H}\mathbf{y}}{\mathbf{y}^T\mathbf{H}\mathbf{y}}$$

Protože $\mathbf{H}\mathbf{y} = \mathbf{g}$ dostaneme konečně po úpravě

$$\frac{f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*)}{f(\mathbf{x}_k) - f(\mathbf{x}^*)} = 1 - \frac{\mathbf{g}_k^T\mathbf{g}_k}{\mathbf{g}_k^T\mathbf{H}^{-1}\mathbf{g}_k \; \mathbf{g}_k^T\mathbf{H}\mathbf{g}_k} \tag{6.13}$$

Zjednodušení předchozího výrazu dostaneme na základě Kantorovičovy nerovnosti.

Věta: Kantorovičova nerovnost
Nechť $\mathbf{Q}$ *je pozitivně definitní symetrická matice. Pak pro libovolný vektor* $\mathbf{x}$ *platí*

$$\frac{\mathbf{x}^T\mathbf{x}}{\mathbf{x}^T\mathbf{Q}\mathbf{x} \;\; \mathbf{x}^T\mathbf{Q}^{-1}\mathbf{x}} \geq \frac{4aA}{(a+A)^2} \tag{6.14}$$

kde a, A *je nejmenší a největší vlastní číslo matice* $\mathbf{Q}$. □

Důkaz: Nechť $0 \leq a = \lambda_1 \leq \lambda_2 \leq \cdots, \leq \lambda_n = A$, kde λ_i jsou vlastní čísla matice $\mathbf{Q}$. Vhodnou volbou souřadnic lze matici $\mathbf{Q}$ diagonalizovat. Pak ale

$$\mathbf{x}^T\mathbf{x} = \sum x_i^2, \qquad \mathbf{x}^T\mathbf{Q}\mathbf{x} = \sum \lambda_i x_i^2, \qquad \mathbf{x}^T\mathbf{Q}^{-1}\mathbf{x} = \sum \frac{1}{\lambda_i} x_i^2$$

Zavedeme $\theta_i = \dfrac{x_i^2}{\sum x_i^2}$ a definujme funkce

$$\Phi(\theta) = \frac{1}{\sum \theta_i \lambda_i}, \qquad \Psi(\theta) = \sum \frac{\theta_i}{\lambda_i}$$

Pak

$$\frac{\mathbf{x}^T\mathbf{x}}{\mathbf{x}^T\mathbf{Q}\mathbf{x} \; \mathbf{x}^T\mathbf{Q}^{-1}\mathbf{x}} = \frac{\sum x_i^2}{\sum \lambda_i x_i^2 \;\; \sum \frac{x_i^2}{\lambda_i}} = \frac{1}{\frac{\sum \lambda_i x_i^2}{\sum x_i^2}} \cdot \frac{1}{\frac{\sum \frac{x_i^2}{\lambda_i}}{\sum x_i^2}} = \frac{\Phi(\theta)}{\Psi(\theta)}$$

Protože $\sum \theta_i = 1$ jsou funkce $\Phi(\theta)$, $\Psi(\theta)$ konvexní kombinace λ_i, nebo $\dfrac{1}{\lambda_i}$. Minimální hodnota poměru ϕ/Ψ je dosažena pro nějaké $\lambda = \theta_1\lambda_1 + \theta_n\lambda_n$, kde λ_1, λ_n jsou po řadě nejmenší a největší vlastní čísla matice $\mathbf{Q}$ a $\theta_1 + \theta_n = 1$. Platí

$$\frac{\theta_1}{\lambda_1} + \frac{\theta_n}{\lambda_n} = \frac{\lambda_1 + \lambda_n - \theta_1\lambda_1 - \theta_n\lambda_n}{\lambda_1\lambda_n} = \frac{\lambda_1 + \lambda_n - \lambda}{\lambda_1\lambda_n}$$

Pak

$$\frac{\Phi(\theta)}{\Psi(\theta)} \geq \min_{\lambda\in(\lambda_1,\lambda_n)} \frac{\frac{1}{\lambda}}{\frac{\lambda_1+\lambda_n-\lambda}{\lambda_1\lambda_n}} = \frac{\lambda_1\lambda_n}{\lambda\left(\lambda_1+\lambda_n-\lambda\right)}$$

Minimum předchozího výrazu je v bodě $\lambda = \frac{\lambda_1+\lambda_n}{2}$, proto

$$\frac{\Phi(\theta)}{\Psi(\theta)} \geq \frac{4\lambda_1\lambda_n}{\left(\lambda_1+\lambda_n\right)^2}$$

□

Proto platí následující tvrzení:

Věta:
Metoda nejrychlejšího sestupu konverguje v kvadratickém případě pro libovolný počáteční bod $\mathbf{x}_0 \in E^n$ k jedinému minimu $\mathbf{x}^*$ a pro všechna k platí

$$\frac{f(\mathbf{x}_{k+1})-f(\mathbf{x}^*)}{f(\mathbf{x}_k)-f(\mathbf{x}^*)} \leq \frac{(A-a)^2}{(A+a)^2} \tag{6.15}$$

kde a, A jsou po řadě nejmenší a největší vlastní čísla matice $\mathbf{H}$ kvadratické formy (6.9).
□

Předchozí tvrzení platí, neboť podle (6.13) a (6.14)

$$\frac{f(\mathbf{x}_{k+1})-f(\mathbf{x}^*)}{f(\mathbf{x}_k)-f(\mathbf{x}^*)} \leq 1-\frac{4aA}{(A+a)^2} = \frac{A^2+2aA+a^2-4aA}{(A+a)^2} = \frac{(A-a)^2}{(A+a)^2}$$

Proto metoda nejrychlejšího sestupu konverguje **lineárně s poloměrem konvergence** nejvýše $\left(\frac{A-a}{A+a}\right)^2 = \left(\frac{r-1}{r+1}\right)^2$, kde $r = \frac{A}{a} \geq 1$ je číslo podmíněnosti matice $\mathbf{H}$ kvadratické formy (6.9).

Nekvadratický případ řeší následující věta:

Věta:
Předpokládáme, že funkce $f(\mathbf{x})$ má spojité druhé parciální derivace a má relativní minimum v bodě $\mathbf{x}^*$. Dále předpokládáme, že Hessova matice $\mathbf{H}(\mathbf{x}^*)$ má minimální vlastní číslo $a \geq 0$ a maximální vlastní číslo A. Jestliže $\{\mathbf{x}_k\}$ je posloupnost generovaná metodou nejrychlejšího sestupu, která konverguje k $\mathbf{x}^*$, pak posloupnost $\{f(\mathbf{x}_k)\}$ konverguje k $f(\mathbf{x}^*)$ lineárně s poloměrem konvergence, který není větší než $\left(\frac{A-a}{A+a}\right)^2$. □

6.3.3 Newtonova metoda a její modifikace

Opět hledáme minimum funkce $f(\mathbf{x})$ pro $\mathbf{x} \in E^n$. Funkci $f(\mathbf{x})$ aproximujeme kvadratickou funkcí, pak

$$f(\mathbf{x}) \doteq f(\mathbf{x}_k) + \mathbf{g}_k^T(\mathbf{x}-\mathbf{x}_k) + \frac{1}{2}(\mathbf{x}-\mathbf{x}_k)^T\mathbf{H}(\mathbf{x}-\mathbf{x}_k) \tag{6.16}$$

Další iteraci $\mathbf{x}_{k+1}$ dostaneme z podmínky nulovosti první derivace předchozí funkce, pak

$$\left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}\right)^T = \mathbf{g}_k + \mathbf{H}(\mathbf{x}-\mathbf{x}_k) = 0$$

Odtud

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{H}^{-1}\mathbf{g}_k \tag{6.17}$$

V obecném případě nekvadratické funkce $f(\mathbf{x})$ nahradíme matici $\mathbf{H}$ maticí druhých derivací $\nabla^2 f(\mathbf{x}_k)$, pak dostaneme **čistý tvar Newtonovy metody**

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \left[\nabla^2 f(\mathbf{x}_k)\right]^{-1} \mathbf{g}(\mathbf{x}_k) \tag{6.18}$$

V obecném případě nekvadratické funkce $f(\mathbf{x})$, pokud jsme daleko od minima, je kvadratická aproximace nepřesná a Hessova matice může být singulární nebo i negativně definitní. Newtonovou metodou můžeme nalézt i maximum, protože Newtonova metoda hledá stacionární bod. Proto se používá **modifikace Newtonovy metody**, podle následujícího algoritmu:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \left[\nabla^2 f(\mathbf{x}_k)\right]^{-1} \mathbf{g}(\mathbf{x}_k) \tag{6.19}$$

Užijeme tedy Newtonův směr hledání $\mathbf{s}_k = -\left[\nabla^2 f(\mathbf{x}_k)\right]^{-1} \mathbf{g}(\mathbf{x}_k)$, pak je iterační postup

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha \mathbf{s}_k$$

Obecnější **modifikované gradientní metodě** ve tvaru

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{S}_k \mathbf{g}(\mathbf{x}_k) \tag{6.20}$$

kde $\mathbf{S}_k$ je nějaká symetrická matice, se budeme věnovat později.

Pro čistou Newtonovu metodu platí věta o lokální konvergenci:

Věta:
Nechť funkce $f(\mathbf{x})$ má spojité třetí parciální derivace, dále předpokládáme, že v lokálním minimu $\mathbf{x}^*$ je Hessova matice $\nabla^2 f(\mathbf{x}^*) = \mathbf{H}(\mathbf{x}^*) > 0$. Potom, je-li počáteční bod dostatečně blízko k $\mathbf{x}$, posloupnost bodů generovaná Newtonovou metodou konverguje k $\mathbf{x}^*$. Řád konvergence Newtonovy metody je alespoň 2.

□

Důkaz : Předpokládejme, že pro body $\mathbf{x}$ a $\mathbf{y}$ v nějakém δ-okolí bodu $\mathbf{x}^*$ platí

$$\|\mathbf{H}(\mathbf{x}) - \mathbf{H}(\mathbf{y})\| \le \beta_1 \|\mathbf{x} - \mathbf{y}\|, \qquad \left\|(\mathbf{H}(\mathbf{x}))^{-1}\right\| \le \beta_2$$

odtud plyne

$$\|\mathbf{H}(\mathbf{x}) - \mathbf{H}(\mathbf{y})\| \|\mathbf{x} - \mathbf{y}\| \le \beta_1 \|\mathbf{x} - \mathbf{y}\|^2.$$

Gradient $\mathbf{g}(\mathbf{x}_k)$ můžeme v δ-okolí bodu $\mathbf{x}^*$ vyjádřit ve tvaru

$$\mathbf{g}(\mathbf{x}_k) = \mathbf{g}(\mathbf{x}^*) + \mathbf{H}(\bar{\mathbf{x}})(\mathbf{x}_k - \mathbf{x}^*) = \mathbf{H}(\bar{\mathbf{x}})(\mathbf{x}_k - \mathbf{x}^*)$$

kde $\bar{\mathbf{x}} \in (\mathbf{x}_k,\ \mathbf{x}^*)$. Poslední úprava v předchozím vztahu platí, neboť $\mathbf{g}(\mathbf{x}^*) = 0$. Pak platí

$$\begin{aligned}
\|\mathbf{x}_{k+1} - \mathbf{x}^*\| &= \left\|\mathbf{x}_k - (\mathbf{H}(\mathbf{x}_k))^{-1}\mathbf{g}(\mathbf{x}_k) - \mathbf{x}^*\right\| \\
&= \left\|(\mathbf{H}(\mathbf{x}_k))^{-1}\left[\mathbf{H}(\mathbf{x}_k)(\mathbf{x}_k - \mathbf{x}^*) - \mathbf{g}(\mathbf{x}_k)\right]\right\| \\
&\le \beta_2 \left\|\mathbf{H}(\mathbf{x}_k)(\mathbf{x}_k - \mathbf{x}^*) - \mathbf{H}(\bar{\mathbf{x}})(\mathbf{x}_k - \mathbf{x}^*)\right\|^2
\end{aligned}$$

Proto platí

$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \leq \beta_1\beta_2 \, ||\mathbf{x}_k - \mathbf{x}^*\|^2$$

a lokální kvadratická konvergence algoritmu je dokázána.

□

S globální konvergencí Newtonovy metody to není moc dobré. Jaké jsou tedy nedostatky čisté Newtonovy metody

- inverze Hessovy matice $\mathbf{H}(\mathbf{x}_k)$ nemusí existovat
- čistá Newtonova metoda nemusí zajišťovat klesání - může nastat $f(\mathbf{x}_{k+1}) > f(\mathbf{x}_k)$
- konečně čistá Newtonova metoda může konvergovat k maximu

Ani modifikovaná Newtonova metoda všechny nedostatky neodstraní. Proto je pro globální konvergenci třeba upravit Newtonovu metodu. Směr hledání $\mathbf{s}_k$ se získá řešením

$$\mathbf{H}_k\mathbf{s}_k = -\mathbf{g}_k$$

Aby řešení existovalo provedeme modifikaci

$$[\mathbf{H}_k + \mathbf{\Delta}_k]\, \mathbf{s}_k = -\mathbf{g}_k$$

kde diagonální matici $\mathbf{\Delta}_k$ volíme tak, aby matice $(\mathbf{H}_k + \mathbf{\Delta}_k)$ byla pozitivně definitní. Tuto modifikaci provedeme vždy, když Newtonův směr neexistuje, nebo to není směr klesání.

Můžeme také provést omezený Newtonův krok, který získáme minimalizací

$$f(\mathbf{x}_k + \mathbf{s}) = f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)\mathbf{s} + \frac{1}{2}\mathbf{s}^T\nabla^2 f(\mathbf{x}_k)\mathbf{s} \tag{6.21}$$

v nějakém okolí nuly. Tomuto okolí se říká **oblast důvěry** (trust region). Pak

$$\mathbf{s}_k = \arg \min_{||\mathbf{s}|| \leq \gamma_k} f(\mathbf{x}_k + \mathbf{s})$$

kde γ_k je nějaká kladná konstanta. Omezený krok zaručuje klesání pro γ_k dostatečně malé.

Problémem je zde volba oblasti důvěry γ_k. Rozumné je zvolit počáteční hodnotu γ_k a zvětšit ji, když iterace postupují úspěšně a naopak ji zmenšit, když jsou iterace neúspěšné. Úspěšnost iterací můžeme hodnotit podle poměru skutečného a predikovaného zlepšení

$$r_k = \frac{f(\mathbf{x}_k) - f(\mathbf{x}_{k+1})}{f(\mathbf{x}_k) - f(\mathbf{x}_k + \mathbf{s}_k)}$$

kde $f(\mathbf{x}_k + \mathbf{s}_k)$ je hodnota získaná kvadratickou aproximací podle (6.21). Pokud $r_k \to 1$ je možno zvětšit γ, proto $\gamma_{k+1} > \gamma_k$. Lze ukázat, že tato omezená Newtonova metoda také řeší problém

$$[\mathbf{H}_k + \delta_k\mathbf{I}]\, \mathbf{s}_k = -\mathbf{g}_k$$

kde $\mathbf{I}$ je jednotková matice a δ_k je nezáporný skalár.

6.3.4 Gaussova-Newtonova metoda

Gaussova-Newtonova metoda řeší problém minimalizace kritéria ve tvaru nejmenších čtverců

$$f(\mathbf{x}) = \frac{1}{2}\left\|\mathbf{h}(\mathbf{x})\right\|^2 = \frac{1}{2}\mathbf{h}^T(\mathbf{x})\mathbf{h}(\mathbf{x}) \tag{6.22}$$

kde $\mathbf{h}(\mathbf{x})$ je nelineární funkce. Jedná se tedy o nelineární nejmenší čtverce. Čistý tvar Gaussovy-Newtonovy metody je založen na linearizaci funkce $\mathbf{h}(\mathbf{x})$ v bodě $\mathbf{x}_k$. Pak

$$\mathbf{h}(\mathbf{x}) \doteq \mathbf{h}(\mathbf{x}_k) + \nabla\mathbf{h}(\mathbf{x}_k)\,(\mathbf{x} - \mathbf{x}_k)$$

Novou iteraci získáme minimalizací normy lineární aproximace (místo $\mathbf{h}(\mathbf{x}_k)$ budeme psát $\mathbf{h}_k$)

$$\begin{aligned}\mathbf{x}_{k+1} &= \arg\min_{\mathbf{x}\in R^n} \frac{1}{2}\left[\mathbf{h}_k + \nabla\mathbf{h}_k\,(\mathbf{x} - \mathbf{x}_k)\right]^T \left[\mathbf{h}_k + \nabla\mathbf{h}_k\,(\mathbf{x} - \mathbf{x}_k)\right] \\ &= \arg\min_{\mathbf{x}\in R^n} \frac{1}{2}\left[\mathbf{h}_k^T\mathbf{h}_k + 2\ (\mathbf{x} - \mathbf{x}_k)^T\nabla\mathbf{h}_k^T\mathbf{h}_k + (\mathbf{x} - \mathbf{x}_k)^T\nabla\mathbf{h}_k^T\nabla\mathbf{h}_k\,(\mathbf{x} - \mathbf{x}_k)\right]\end{aligned}$$

Odtud po doplnění na úplný čtverec dostaneme minimální hodnotu předchozího výrazu

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \left(\nabla\mathbf{h}_k^T\ \nabla\mathbf{h}_k\right)^{-1}\ (\nabla\mathbf{h}_k)^T\,\mathbf{h}_k \tag{6.23}$$

Toto je **čistý tvar Gaussovy - Newtonovy metody**.

Pokud je matice $(\nabla\mathbf{h}_k^T\ \nabla\mathbf{h}_k)$ singulární , pak ji regularizujeme volbou takové diagonální matice Δ_k, aby matice $(\nabla\mathbf{h}_k^T\ \nabla\mathbf{h}_k + \Delta_k)$ byla pozitivně definitní. Pokud diagonální matici Δ_k volíme $\Delta_k = \alpha_k\mathbf{I}$, $(\alpha_k > 0)$ dostaneme tzv. **Levenbergovu - Marquardtovu metodu**. Místo (6.23) je iterační algoritmus podle Levenberga-Marquardta

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \left(\nabla\mathbf{h}_k^T\ \nabla\mathbf{h}_k + \alpha_k\mathbf{I}\right)^{-1}\ (\nabla\mathbf{h}_k)^T\,\mathbf{h}_k \tag{6.24}$$

Pokud je koeficient α_k malý, metoda se blíží Gauss - Newtonově metodě, pokud se koeficient α_k blíží nekonečnu, pak se blížíme metodě nejrychlejšího sestupu. Zde je problémem volba vhodného koeficientu α_k. Nejprve musí být dostatečně veliký, abychom se dostatečně rychle blížíli k optimu a pokud jsme blízko optima, pak naopak musí být malý, abychom zaručili konvergenci algoritmu.

Nelineární nejmenší čtverce mají řadu praktických aplikací specielně v nelineární estimaci a aproximaci.

Příklad: Trénování neuronových sítí
Neuronová síť reprezentuje model nelineárního systému. Neuronová síť se skládá z vícevrstvých perceptronů. Nechť tedy neuronová síť má N vrstev. Každá vrstva, kterou budeme označovat indexem $k = 0, \ldots, N-1$ je tvořena n_k akčními jednotkami - které jsou popsány nelineárním zobrazením mezi jedním vstupním a jedním výstupním signálem - $\phi(\xi)$. Užívané nelineární funkce jsou na příklad

$$\begin{aligned}\phi(\xi) &= \frac{1}{1+e^{-\xi}}, && \text{sigmoidální funkce} \\ \phi(\xi) &= \frac{e^{\xi} - e^{-\xi}}{e^{\xi} + e^{-\xi}}, && \text{funkce hyperbolický tangens}\end{aligned}$$

Vstupem do j-té aktivační jednotky je lineární kombinace signálů $(x_k^1, \ldots, x_k^{n_k})$, které jsou výstupem z předchozí vrstvy. Tyto signály tvoří vektor $\mathbf{x}_k$. Výstup j-té aktivační jednotky v $(k+1)$ vrstvě je signál, který označíme x_{k+1}^j. Platí tedy

$$x_{k+1}^j = \phi\left(\alpha_k^{0j} + \sum_{s=1}^{n_k} \alpha_k^{sj} x_k^s\right), \qquad j = 1, \ldots, n_{k+1},$$

kde váhy α_k^{sj} je třeba určit. Určení vah vede na řešení problému nelineárních nejmenších čtverců. Tento proces se také nazývá učení nebo také trénování neuronové sítě. Váhy ve všech vrstvách a ve všech jednotkách tvoří vektor vah

$$\alpha = \left\{\alpha_k^{sj} \;:\; k = 0, \ldots, N-1,\ s = 0, \ldots, n_k,\ j = 1, \ldots, n_{k+1}\right\}$$

Pro daný vektor vah α a vstupní vektor $\mathbf{u} = \mathbf{x}_0$ do první vrstvy produkuje neuronová síť jediný výstupní vektor $\mathbf{y} = \mathbf{x}_N$ z N-té vrstvy. Celá neuronová síť realizuje tedy nelineární zobrazení vstupního vektoru $\mathbf{u} = \mathbf{x}_0$ na výstupní vektor $\mathbf{y} = \mathbf{x}_N$ a toto zobrazení je parametrizováno vektorem vah α. Pak tedy

$$\mathbf{y} = \mathbf{x}_N = \mathbf{f}(\alpha, \mathbf{x}_0) = \mathbf{f}(\alpha, \mathbf{u}).$$

Předpokládejme, že známe m párů vstupních a výstupních signálů $(\mathbf{u}_1,\ \mathbf{y}_1)$, $\ldots, (\mathbf{u}_m,\ \mathbf{y}_m)$ změřených na reálném objektu. Naším cílem je pomocí neuronové sítě vytvořit model objektu, který by reagoval stejně jako reálný objekt. Chceme tedy natrénovat neuronovou síť tak, aby odchylka $\mathbf{e}_i$ výstupu neuronové sitě a výstupu reálného objektu $\mathbf{e}_i = \mathbf{y}_i - \mathbf{f}(\alpha, \mathbf{u}_i)$ byla co nejmenší pro všechny i-té dvojice vstupního a výstupního signálu. To realizujeme určením vah α, které minimalizují součet kvadrátů chyb

$$\alpha^* = \arg\min_{\alpha} \frac{1}{2} \sum_{i=1}^{m} \|\mathbf{y}_i - f(\alpha, \mathbf{u}_i)\|^2 = \arg\min_{\alpha} \mathbf{e}^T(\alpha)\mathbf{e}(\alpha)$$

kde vektor $\mathbf{e}(\alpha)$ je vektor chyb, který je nelineárně závislý na váhách α. Učení neuronové sítě vede tedy na problém nelineárních nejmenších čtverců.

6.3.5 Metody konjugovaných směrů

Metody konjugovaných směrů byly vyvinuty proto, aby urychlily konvergenci gradientních metod a vyhnuly se potížím spojeným s modifikací Newtonovy metody. Pomocí konjugovaných směrů můžeme vyvinout výpočetní postupy, které konvergují po konečném počtu kroků (samozřejmě při minimalizaci kvadratické funkce).

Nejprve si uvedeme definici konjugovaných vektorů.

Definice:
Vektory $\mathbf{s}_1, \ldots \mathbf{s}_p$ *jsou konjugované k symetrické matici* $\mathbf{Q}$*, když platí*

$$\mathbf{s}_i^T\ \mathbf{Q}\ \mathbf{s}_j = 0, \qquad \forall\ i,\ j,\quad i \neq j \tag{6.25}$$

□

Uvažujme nejprve kvadratickou funkci

$$f(\mathbf{x}) = \mathbf{a}^T\mathbf{x} + \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x}$$

Pak platí následující tvrzení:

Věta:
Jsou-li $\mathbf{s}_0, \mathbf{s}_1, \ldots \mathbf{s}_{n-1}$ *nenulové vektory konjugované vzhledem k matici* $\mathbf{H}$*, pak pro každé* $\mathbf{x}_0 \in R^n$ *a*

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha\mathbf{s}_k,$$

kde $\alpha = \alpha_k$ *je voleno tak, aby*

$$\alpha_k = \arg\min_{\alpha} f(\mathbf{x}_k + \alpha\mathbf{s}_k)$$

platí, že $\mathbf{x}_n = \mathbf{x}^*$.

□

Metoda konjugovaných směrů tedy konverguje v kvadratickém případě nejvýše v n krocích. Provedeme konstruktivní důkaz předchozí věty.

Důkaz: Pokud platí předchozí věta, pak

$$\mathbf{x}^* = \mathbf{x}_0 + \alpha_0\mathbf{s}_0 + \alpha_1\mathbf{s}_1 + \cdots + \alpha_{n-1}\mathbf{s}_{n-1}$$

Je tomu vskutku tak, neboť první dva členy na pravé straně přechozí rovnice jsou rovny $\mathbf{x}_1$, první tři členy jsou rovny $\mathbf{x}_2$, atd.. Proto

$$\mathbf{x}^* - \mathbf{x}_0 = \alpha_0\mathbf{s}_0 + \alpha_1\mathbf{s}_1 + \cdots + \alpha_{n-1}\mathbf{s}_{n-1}$$

Předchozí rovnici budeme zleva násobit vektorem $\mathbf{s}_k^T\mathbf{H}$ a protože jsou směry $\mathbf{s}_i$ konjugované, platí

$$\mathbf{s}_k^T\mathbf{H}\,(\mathbf{x}^* - \mathbf{x}_0) = \alpha_k\mathbf{s}_k^T\mathbf{H}\mathbf{s}_k.$$

Proto

$$\alpha_k = \frac{\mathbf{s}_k^T\mathbf{H}\,(\mathbf{x}^* - \mathbf{x}_0)}{\mathbf{s}_k^T\mathbf{H}\mathbf{s}_k}$$

Pro libovolné k platí také

$$\mathbf{x}_k - \mathbf{x}_0 = \alpha_0\mathbf{s}_0 + \alpha_1\mathbf{s}_1 + \cdots + \alpha_{k-1}\mathbf{s}_{k-1}$$

Předchozí rovnici budeme opět zleva násobit vektorem $\mathbf{s}_k^T\mathbf{H}$, pak

$$\mathbf{s}_k^T\mathbf{H}\,(\mathbf{x}_k - \mathbf{x}_0) = 0, \quad \Rightarrow \quad \mathbf{s}_k^T\mathbf{H}\mathbf{x}_k = \mathbf{s}_k^T\mathbf{H}\mathbf{x}_0$$

Proto

$$\alpha_k = \frac{\mathbf{s}_k^T\mathbf{H}\mathbf{x}^* - \mathbf{s}_k^T\mathbf{H}\mathbf{x}_0}{\mathbf{s}_k^T\mathbf{H}\mathbf{s}_k} = \frac{\mathbf{s}_k^T\mathbf{H}\mathbf{x}^* - \mathbf{s}_k^T\mathbf{H}\mathbf{x}_k}{\mathbf{s}_k^T\mathbf{H}\mathbf{s}_k}$$

Gradient naší kvadratické funkce je

$$\mathbf{g}_k = \mathbf{a} + \mathbf{H}\mathbf{x}_k, \qquad \mathbf{g}(\mathbf{x}^*) = 0 = \mathbf{a} + \mathbf{H}\mathbf{x}^*$$

Proto můžeme upravit výraz pro α_k do konečného tvaru

$$\alpha_k = \frac{\mathbf{s}_k^T (\mathbf{H}\mathbf{x}^* - \mathbf{H}\mathbf{x}_k)}{\mathbf{s}_k^T \mathbf{H}\mathbf{s}_k} = \frac{\mathbf{s}_k^T (-\mathbf{a} - \mathbf{g}_k + \mathbf{a})}{\mathbf{s}_k^T \mathbf{H}\mathbf{s}_k} = -\frac{\mathbf{s}_k^T \mathbf{g}_k}{\mathbf{s}_k^T \mathbf{H}\mathbf{s}_k}$$

Tento výraz pro α_k je ale totožný s optimálním α_k určeným podle $\arg\min_\alpha f(\mathbf{x}_k + \alpha\mathbf{s}_k)$, neboť potom musí platit

$$\frac{\partial f(\mathbf{x}_k + \alpha\mathbf{s}_k)}{\partial \alpha} = 0$$

Po dosazení za $f(\mathbf{x}_k + \alpha\mathbf{s}_k) = \mathbf{a}^T(\mathbf{x}_k + \alpha\mathbf{s}_k) + \frac{1}{2}(\mathbf{x}_k + \alpha\mathbf{s}_k)^T\mathbf{H}(\mathbf{x}_k + \alpha\mathbf{s}_k)$ do předchozího výrazu dostaneme pro α_k stejný vztah jako v předchozím odvození.

□

V obecném případě nekvadratické funkce $f(\mathbf{x})$ místo matice $\mathbf{H}$ uvažujeme matici $\nabla^2 f(\mathbf{x})$.

Pro kvadratický případ platí $\mathbf{g}_k^T\mathbf{s}_i = 0$ pro $i < k$, což znamená, že směry $\mathbf{g}_k$ a $\mathbf{s}_i$ jsou navzájem kolmé. Platnost tohoto vztahu ukážeme ve dvou krocích. Nejprve, je-li i o jednotku menší než k, pak

$$\begin{aligned}\mathbf{g}_{k+1}^T\mathbf{s}_k &= (\mathbf{a} + \mathbf{H}\mathbf{x}_{k+1})^T\mathbf{s}_k = (\mathbf{a} + \mathbf{H}\mathbf{x}_k + \alpha\mathbf{H}\mathbf{s}_k)^T\mathbf{s}_k \\ &= +(\mathbf{a} + \mathbf{H}\mathbf{x}_k)^T\mathbf{s}_k + \alpha\mathbf{s}_k^T\mathbf{H}\mathbf{s}_k = \mathbf{g}_k^T\mathbf{s}_k - \frac{\mathbf{s}_k^T\mathbf{g}_k}{\mathbf{s}_k^T\mathbf{H}\mathbf{s}_k}\mathbf{s}_k^T\mathbf{H}\mathbf{s}_k = 0\end{aligned}$$

V obecném případě pro $i < k$ platí

$$\mathbf{g}_{k+1}^T\mathbf{s}_i = (\mathbf{a} + \mathbf{H}\mathbf{x}_{k+1})^T\mathbf{s}_i + \alpha\mathbf{s}_k^T\mathbf{H}\mathbf{s}_i = \mathbf{g}_k^T\mathbf{s}_i$$

protože směry $\mathbf{s}_i$ jsou konjugované směry. Protože $\mathbf{g}_{k+1}^T\mathbf{s}_i = \mathbf{g}_k^T\mathbf{s}_i$, pak také $\mathbf{g}_k^T\mathbf{s}_i = \mathbf{g}_{k-1}^T\mathbf{s}_i$ až $= \mathbf{g}_{i+1}^T\mathbf{s}_i = 0$.

6.3.6 Metoda konjugovaných gradientů

V předchozím odstavci jsme neřešili problém generování konjugovaných směrů. Ukázali jsme pouze, že pokud máme n konjugovaných směrů, pak jsou-li to směry hledání a v těchto směrech postupujeme v každé iteraci s optimálním krokem, metoda v kvadratickém případě konverguje v konečném počtu iterací. V metodě popisované v tomto odstavci se konjugované směry generují z gradientů a dosavadního směru pohybu. Metoda konjugovaných gradientů je tedy specielní metodou konjugovaných směrů. Konjugované směry se generují sekvenčně podle následujícího algoritmu:

Algoritmus konjugované gradientní metody:

- Startujeme z libovolného bodu $\mathbf{x}_0 \in R^n$. První směr $\mathbf{s}_0$ je ve směru záporného gradientu

$$\mathbf{s}_0 = -\mathbf{g}_0 = -\mathbf{a} - \mathbf{H}\mathbf{x}_0 \tag{6.26}$$

  poslední rovnost platí pro kvadratický případ $f(\mathbf{x}) = \mathbf{a}^T\mathbf{x} + \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x}$.

- Obecně platí

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{s}_k \tag{6.27}$$

kde

$$\alpha_k = \frac{\mathbf{g}_k^T \mathbf{s}_k}{\mathbf{s}_k^T \mathbf{H} \mathbf{s}_k} \tag{6.28}$$

- Nový konjugovaný směr je

$$\mathbf{s}_{k+1} = -\mathbf{g}_{k+1} + \beta_k \mathbf{s}_k \tag{6.29}$$

kde β_k je optimální, $\beta_k = \arg\min_\beta f\left(\mathbf{x}_{k+1} + \alpha_k\left(-\mathbf{g}_{k+1} + \beta \mathbf{s}_k\right)\right)$. Platí

$$\beta_k = \frac{\mathbf{g}_{k+1}^T \mathbf{H} \mathbf{s}_k}{\mathbf{s}_k^T \mathbf{H} \mathbf{s}_k} \tag{6.30}$$

□

Směry $\mathbf{s}_i$ jsou konjugované směry. Platí tedy $\mathbf{s}_{k+1}^T \mathbf{H} \mathbf{s}_i = 0$, pro $i \leq k$. Po dosazení z (6.29) dostaneme

$$\mathbf{s}_{k+1}^T \mathbf{H} \mathbf{s}_i = -\mathbf{g}_{k+1}^T \mathbf{H} \mathbf{s}_i + \beta_k \mathbf{s}_k^T \mathbf{H} \mathbf{s}_i = 0, \quad \text{pro} \;\; i \leq k$$

Pro $i = k$ to platí, neboť podle (6.30) je právě podle tohoto vztahu β_k určeno. Pro $i \leq k$ jsou oba členy nulové, čili $\mathbf{g}_{k+1}^T \mathbf{H} \mathbf{s}_i = 0$ a také $\mathbf{s}_k^T \mathbf{H} \mathbf{s}_i = 0$. Platnost předchozích výrazů lze dokázat úplnou indukcí.

Pro nekvadratický případ nutno počítat gradient i Hessovu matici. Metoda konjugovaných směrů pak bohužel není globálně konvergentní, to znamená, že samozřejmě není ani konvergentní v konečném počtu kroků.

Proto po n nebo $n + 1$ krocích nutno provést reinicializaci, to znamená nastartovat metodu konjugovaných gradientů od počátku znova, neboť metoda startuje s čistým gradientním krokem.

Abychom se vyhnuli nutnosti výpočtu Hessovy matice druhých derivací, která se vyskytuje ve výpočtu velikosti kroku α_k - viz. (6.28), provádí se výpočet α_k pomocí algoritmu jednorozměrového hledání. Výpočet koeficientu β_k se provádí podle jiných vztahů, ve kterých se nevyskytuje Hessova matice.

Jeden takový postup se nazývá **metoda Fletchera a Reevese**. Podle této metody se výpočet α_k provádí algoritmy jednorozměrového hledání a výpočet β_k je dle vztahu

$$\beta_k = \frac{\mathbf{g}_{k+1}^T \mathbf{g}_{k+1}}{\mathbf{g}_k^T \mathbf{g}_k} \tag{6.31}$$

Přitom po n krocích se provádí reinicializace. Předchozí vztah pro β_k je ekvivalentní s (6.30) v kvadratickém případě.

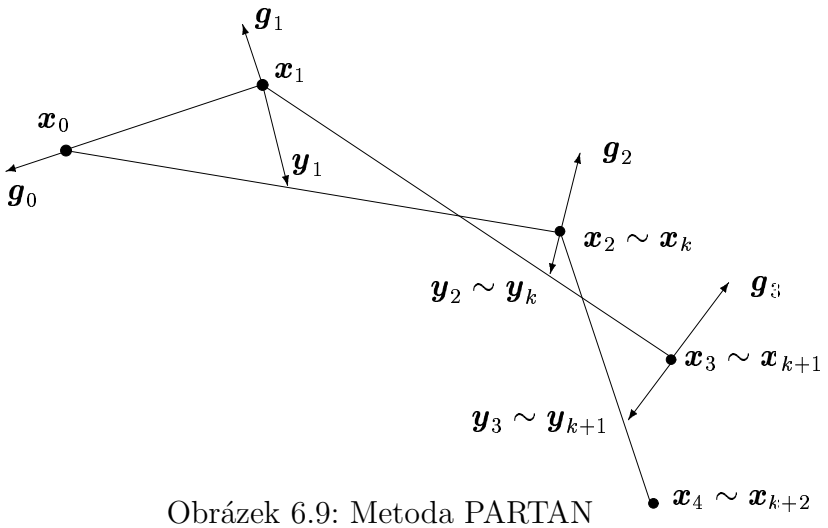
Jiná modifikace je **metoda Polaka a Ribiera**. Podle této metody se koeficient β_k počítá podle vztahu

$$\beta_k = \frac{\left(\mathbf{g}_{k+1} - \mathbf{g}_k\right)^T \mathbf{g}_{k+1}}{\mathbf{g}_k^T \mathbf{g}_k} \tag{6.32}$$

Při tom po n krocích se opět provádí reinicializace. Předchozí vztah pro β_k je opět ekvivalentní s (6.30) v kvadratickém případě. Simulace prokazují, že nejvýhodnější je metoda Polaka a Ribiera.

Jiná metoda konjugovaných směrů je tak zvaná **metoda paralelních tečen**, nebo krátce **metoda PARTAN**.

Motivací k této metodě bylo to, že gradientní metoda nejrychlejšího sestupu generuje směry sestupu, které jsou navzájem kolmé. Někdy se tento jev označuje názorně jako postup "cik - cak". V metodě PARTAN se dělá zrychlený krok, který se odvozuje z gradientu a předchozího směru.



Obrázek 6.9: Metoda PARTAN

Algoritmus metody PARTAN je následující - viz obr. 6.9:

- Startujeme z libovolného bodu $\mathbf{x}_0$. Bod $\mathbf{x}_1$ určíme gradientní metodou nejrychlejšího sestupu

  $$\mathbf{x}_1 = \mathbf{x}_0 + \alpha_0 \mathbf{g}_0$$

  kde α_0 je optimálně určeno algoritmem jednorozměrového hledání.

- Nyní obecně z bodu $\mathbf{x}_k$ určíme nejprve pomocný bod

  $$\mathbf{y}_k = \mathbf{x}_k + \alpha_k \mathbf{g}_k$$

  kde α_k je opět optimálně určeno algoritmem jednorozměrového hledání.

- Potom se provede zrychlený krok

  $$\mathbf{x}_{k+1} = \mathbf{x}_{k-1} + \beta_k \left(\mathbf{y}_k - \mathbf{x}_{k-1}\right)$$

  kde β_k je opět optimálně určeno algoritmem jednorozměrového hledání.

□

V jednom iteračním kroku se tedy provádí dvě jednorozměrové optimalizace. Metoda PARTAN je metoda ekvivalentní metodě konjugovaných gradientů.

6.3.7 Kvazi-newtonovské metody

Kvazi-newtonovské metody jsou gradientní metody, které leží někde mezi metodou nejrychlejšího sestupu a Newtonovou metodou. Snaží se využít přednosti obou metod. Gradientní metody mají zaručenou konvergenci a Newtonova metoda má v okolí optima řád konvergence rovný dvěma. Newtonova metoda ale vyžaduje výpočet Hessovy matice, nebo spíše její inverze.

Minimalizaci funkce $f(\mathbf{x})$ podle **modifikované Newtonovy metody** provedeme podle následujícího algoritmu

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{S}_k \nabla^T f(\mathbf{x}_k) \tag{6.33}$$

kde $\mathbf{S}_k$ je symetrická matice a α_k se volí takové, aby $\mathbf{x}_{k+1}(\alpha)$ bylo minimální.

Je-li matice $\mathbf{S}_k$ rovná inverzi Hessovy matice, pak se jedná o modifikaci Newtonovy metody, je-li $\mathbf{S}_k = \mathbf{I}$, pak se naopak jedná o metodu nejrychlejšího sestupu. Aby algoritmus (6.33) byl klesající algoritmus, je nutné, aby matice $\mathbf{S}_k$ byla pozitivně definitní.

Protože modifikovaná Newtonova metoda se blíží gradientní metodě, bude i její konvergence podobná konvergenci gradientní metody. Pro kvadratický případ minimalizace

$$f(\mathbf{x}) = \frac{1}{2}(\mathbf{x} - \mathbf{x}^*)^T \mathbf{H}(\mathbf{x} - \mathbf{x}^*),$$

můžeme pro algoritmus (6.33) určit optimální krok α_k

$$\alpha_k = \frac{\mathbf{g}_k^T \mathbf{S}_k \mathbf{g}_k}{\mathbf{g}_k^T \mathbf{S}_k \mathbf{H} \mathbf{S}_k \mathbf{g}_k}, \qquad \mathbf{g}_k = \mathbf{H}(\mathbf{x}_k - \mathbf{x}^*)$$

Potom řád konvergence modifikované Newtonovy metody je určen následujícím tvrzením:

Věta: *Konvergence modifikované Newtonovy metody.*
Pro algoritmus (6.33) platí v kvadratickém případě pro každé k

$$f(\mathbf{x}_{k+1}) \leq \left(\frac{A_k - a_k}{A_k - a_k}\right)^2 f(\mathbf{x}_k)$$

kde A_k, a_k jsou po řadě největší a nejmenší vlastní čísla matice $\mathbf{S}_k\mathbf{H}$.

□

Konvergence modifikované Newtonovy metody je tedy lineární a poloměr konvergence závisí na maximálním a minimálním vlastním čísle matice $\mathbf{S}_k\mathbf{H}$.

Uvedeme ještě tzv. **klasickou modifikovanou Newtonovu metodu**, ve které se Hessova matice $\mathbf{H}(\mathbf{x}_k)$ neurčuje znova v každém kroku. Algoritmus je následující

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \left(\mathbf{H}(\mathbf{x}_0)\right)^{-1} \nabla^T f(\mathbf{x}_k) \tag{6.34}$$

Inverze Hessovy matice se podle tohoto algoritmu provádí pouze v počátečním bodě $\mathbf{x}_0$.

Základní myšlenka **kvazi - newtonovských metod** je snaha konstruovat inverzi Hessovy matice nebo její aproximaci užitím informace získávané během iteračního procesu.

Nechť funkce $f(\mathbf{x})$ má spojité druhé derivace. V bodech $\mathbf{x}_{k+1}$ a $\mathbf{x}_k$ určíme gradienty $\mathbf{g}_{k+1} = (\nabla f(\mathbf{x}_{k+1}))^T$, $\mathbf{g}_k = (\nabla f(\mathbf{x}_k))^T$ a definujeme vektor

$$\mathbf{d}_k = \mathbf{x}_{k+1} - \mathbf{x}_k.$$

Potom Hessovu matici můžeme aproximovat podle vztahu

$$\mathbf{g}_{k+1} - \mathbf{g}_k \doteq \mathbf{H}(\mathbf{x}_k)(\mathbf{x}_{k+1} - \mathbf{x}_k) = \mathbf{H}(\mathbf{x}_k)\mathbf{d}_k$$

Je-li Hessova matice konstantní, jak je tomu v kvadratickém případě, pak, zavedeme-li vektor $\mathbf{q}_k = \mathbf{g}_{k+1} - \mathbf{g}_k$, platí $\mathbf{q}_k = \mathbf{H}_k\mathbf{d}_k$.

Máme-li tedy n nezávislých směrů $\mathbf{d}_0$, $\mathbf{d}_1$, …, $\mathbf{d}_{n-1}$ a jim odpovídající gradienty $\mathbf{g}_i$ a z toho plynoucí vektory $\mathbf{q}_i = \mathbf{g}_{i+1} - \mathbf{g}_i$, pak

$$\mathbf{H} = \mathbf{Q}\mathbf{D}^{-1}$$

kde složené matice $\mathbf{Q}$, $\mathbf{D}$ jsou

$$\mathbf{D} = [\mathbf{d}_0, \ldots, \mathbf{d}_{n-1}], \qquad \mathbf{Q} = \left[\mathbf{q}_0, \ldots, \mathbf{q}_{n-1}\right]$$

Vyvinutý algoritmus výpočtu Hessovy matice není iterační. Spíše než Hessovu matici potřebujeme určit její inverzi.

Hledáme-li aproximaci $\mathbf{S}_{k+1}$ inverze Hessovy matice založené na datech z jednotlivých kroků iteračního algoritmu, pak chceme, aby platilo

$$\mathbf{S}_{k+1}\mathbf{q}_i = \mathbf{d}_i, \qquad 0 \leq i \leq k.$$

Pak pro $k = n - 1$ je $\mathbf{S}_n = \mathbf{H}^{-1}$. Iterační postup má mnoho řešení. Uvedeme některé iterační metody výpočtu aproximace Hessovy matice.

Korekce pomocí matice s jednotkovou hodností

Aproximaci inverze Hessovy matice budeme provádět iteračně podle vztahu

$$\mathbf{S}_{k+1} = \mathbf{S}_k + a_k\mathbf{z}_k\mathbf{z}_k^T \tag{6.35}$$

kde a_k je nějaká konstanta a $\mathbf{z}_k$ je nějaký vektor, které je třeba určit. Matice $\mathbf{z}_k\mathbf{z}_k^T$ je čtvercová matice hodnosti jedna, která se také nazývá **dyáda**.

Z $\mathbf{S}_{k+1}\mathbf{q}_i = \mathbf{d}_i$ platí pro $i = k$

$$\mathbf{d}_k = \mathbf{S}_{k+1}\mathbf{q}_k = \mathbf{S}_k\mathbf{q}_k + a_k\mathbf{z}_k\mathbf{z}_k^T\mathbf{q}_k$$

Předchozí vztah násobíme zleva vektorem $\mathbf{q}_k^T$ a dostaneme

$$\mathbf{q}_k^T\mathbf{d}_k - \mathbf{q}_k^T\mathbf{S}_k\mathbf{q}_k = a_k\left(\mathbf{z}_k^T\mathbf{q}_k\right)^2$$

Odtud dostaneme vztah, který později využijeme

$$a_k\left(\mathbf{z}_k^T\mathbf{q}_k\right)^2 = \mathbf{q}_k^T\left(\mathbf{d}_k - \mathbf{S}_k\mathbf{q}_k\right) \tag{6.36}$$

Nyní odvodíme iterační algoritmus následujícími úpravami vztahu (6.35), pak

$$\begin{aligned}
\mathbf{S}_{k+1} &= \mathbf{S}_k + a_k\mathbf{z}_k\mathbf{z}_k^T \\
&= \mathbf{S}_k + \frac{a_k^2\left(\mathbf{z}_k^T\mathbf{q}_k\right)^2}{a_k\left(\mathbf{z}_k^T\mathbf{q}_k\right)^2}\mathbf{z}_k\mathbf{z}_k^T \\
&= \mathbf{S}_k + \frac{a_k^2\mathbf{z}_k\left(\mathbf{z}_k^T\mathbf{q}_k\ \mathbf{q}_k^T\mathbf{z}_k\right)\mathbf{z}_k^T}{\mathbf{q}_k^T\left(\mathbf{d}_k - \mathbf{S}_k\mathbf{q}_k\right)}
\end{aligned}$$

Odtud konečně dostaneme

$$\mathbf{S}_{k+1} = \mathbf{S}_k + \frac{(\mathbf{d}_k - \mathbf{S}_k\mathbf{q}_k)\,(\mathbf{d}_k - \mathbf{S}_k\mathbf{q}_k)^T}{\mathbf{q}_k^T\,(\mathbf{d}_k - \mathbf{S}_k\mathbf{q}_k)} \tag{6.37}$$

Ukázali jsme, že pro $i = k$ platí (6.37). Tento vztah platí i pro $i < k$. Proto, když Hessova matice je konstantní, matice $\mathbf{S}_k$ konverguje k $\mathbf{H}^{-1}$ v nejvýše n krocích. Jediná potíž při aplikaci algoritmu v obecném případě je nutnost zachovat v jednotlivých krocích pozitivní definitnost matice $\mathbf{S}_k$. Proto musí platit $\mathbf{q}_k^T\,(\mathbf{d}_k - \mathbf{S}_k\mathbf{q}_k) > 0$, což ale obecně není garantováno.

Metoda Davidona, Fletchera a Powella

Jiný postup je korekce aproximace inverze Hessovy matice maticí hodnosti dvě. Metodu navrhl Davidon a později rozvinuli Fletcher s Powellem. Metodu označujeme zkráceně podle autorů jako **DFP metodu**.

Metodu startujeme z libovolného počátečního bodu $\mathbf{x}_0$ a libovolné symetrické pozitivně definitní matice $\mathbf{S}_0$. Další iteraci získáme podle

$$\mathbf{x}_{k+1} = \arg\min_{\alpha} f(\mathbf{x}_k + \alpha\mathbf{s}_k) \tag{6.38}$$

kde směr $\mathbf{s}_k$ je

$$\mathbf{s}_k = -\mathbf{S}_k\mathbf{g}_k. \tag{6.39}$$

Vektor $\mathbf{g}_k$ je vektor gradientu a matice $\mathbf{S}_k$ je dle

$$\mathbf{S}_{k+1} = \mathbf{S}_k + \frac{\mathbf{d}_k\mathbf{d}_k^T}{\mathbf{d}_k^T\mathbf{q}_k} - \frac{\mathbf{S}_k\mathbf{q}_k\;\mathbf{q}_k^T\mathbf{S}_k}{\mathbf{q}_k^T\mathbf{S}_k\mathbf{q}_k} \tag{6.40}$$

kde vektor $\mathbf{q}_k = \mathbf{g}_{k+1} - \mathbf{g}_k$.

Při minimalizaci kvadratické funkce jsou vektory $\mathbf{s}_i$ konjugované vzhledem k Hessově matici. Metoda DFP je tedy metoda konjugovaných směrů a pro $\mathbf{S}_0 = \mathbf{I}$ je to metoda konjugovaných gradientů.

Broydenovy metody

Zaměníme-li vektory $\mathbf{d}_k$ a $\mathbf{q}_k$, pak vlastně nehledáme přímo inverzi Hessiánu, ale přímo Hessovu matici. Potom tedy místo matice $\mathbf{S}_k$ aktualizujeme nějakou jinou symetrickou matici, kterou třeba označíme $\mathbf{Q}_k$. Potom po záměně vektorů pro ni platí vztah

$$\mathbf{Q}_{k+1} = \mathbf{Q}_k + \frac{\mathbf{q}_k\mathbf{q}_k^T}{\mathbf{q}_k^T\mathbf{d}_k} - \frac{\mathbf{Q}_k\mathbf{d}_k\;\mathbf{d}_k^T\mathbf{Q}_k}{\mathbf{d}_k^T\mathbf{Q}_k\mathbf{d}_k} \tag{6.41}$$

Tato aktualizace se označuje jako aktualizace Broydena, Fletchera, Goldfarba a Shannoa, zkráceně **aktualizace BFGS**.

Poznámka: V některých pramenech se uvádí jiný vztah pro aktualizaci Hessovy matice podle BFGS

$$\mathbf{Q}_{k+1} = \mathbf{Q}_k + \frac{\mathbf{q}_k\mathbf{q}_k^T}{\mathbf{q}_k^T\mathbf{d}_k} - \frac{\mathbf{Q}_k\;\mathbf{Q}_k}{\mathbf{d}_k^T\mathbf{Q}_k\mathbf{d}_k} \tag{6.42}$$

□

Inverzi matice $\mathbf{Q}_k$ provedeme podle **věty o maticové inverzi**

$$\left[\mathbf{A} + \mathbf{a}\mathbf{b}^T\right]^{-1} = \mathbf{A}^{-1} + \frac{\mathbf{A}^{-1}\mathbf{a}\mathbf{b}^T\mathbf{A}^{-1}}{1 + \mathbf{b}^T\mathbf{A}^{-1}\mathbf{a}} \tag{6.43}$$

kde $\mathbf{A}$ je $n \times n$ matice a $\mathbf{a}$, i $\mathbf{b}$ jsou n-rozměrné vektory. Předchozí formule platí, když všechny inverze existují. Větu o maticové inverzi musíme v našem případě použít dvakrát.

Potom aktualizace inverze Hessovy matice je

$$\mathbf{S}_{k+1} = \mathbf{S}_k + \left(1 + \frac{\mathbf{q}_k^T\mathbf{S}_k\mathbf{q}_k}{\mathbf{q}_k^T\mathbf{d}_k}\right)\frac{\mathbf{d}_k\mathbf{d}_k^T}{\mathbf{d}_k\mathbf{q}_k} - \frac{\mathbf{d}_k\mathbf{q}_k^T\mathbf{S}_k + \mathbf{S}_k\mathbf{q}_k\mathbf{d}_k^T}{\mathbf{q}_k^T\mathbf{d}_k} \tag{6.44}$$

Numerické simulace ukazují, že tato metoda dává lepší výsledky než metoda DFP. Obě metody korigují v další iteraci odhad Hessovy matice maticí hodnosti dvě. Proto je možno definovat celou **třídu metod Broydenových**

$$\mathbf{S}^{\varphi} = (1 - \varphi)\,\mathbf{S}^{DFP} + \varphi\mathbf{S}^{SFGS} \tag{6.45}$$

kde $\mathbf{S}^{DFP}$ je aktualizace inverze Hessovy matice podle metody DFP a obdobně $\mathbf{S}^{BFGS}$ je aktualizace inverze Hessovy matice podle metody BFGS a φ je reálný, obvykle nezáporný, parametr.

V těchto metodách velice záleží na kvalitě algoritmu jednorozměrového hledání parametru α. Metoda se obvykle znovu startuje po m krocích, kde $m < n$. Metody tohoto typu jsou globálně konvergentní a lokálně konvergují superlineárně.

6.4 Numerické metody s omezením

Většina optimalizačních problémů má omezující podmínky. Proto numerické metody jejich řešení jsou velmi důležité. Numerické metody optimalizace s omezením se dělí na dvě základní skupiny - primární a duální metody.

Primární metody jsou metody, které používají původní formulaci problému. Hledají optimum v dovoleném rozsahu proměnných, to znamená, že stále (v každé iteraci) řešení je přípustné. To znamená, že pokud ukončíme výpočet, dosažené řešení je přípustné a leží zřejmě blízko optima. Mezi nevýhody těchto metod počítáme nutnost nalezení přípustného bodu při startu primárních metod a nalezení přípustného směru, abychom neporušili omezení. Tyto metody obyčejně nevyužívají specielní strukturu problému a jsou proto použitelné pro obecné problémy nelineárního programování. Rychlost konvergence těchto metod je srovnatelná s druhými metodami.

Duální metody využívají nutné podmínky optima. Používají Lagrangeovy koeficienty. Řeší problém obvykle v jiné dimenzi.

Mezi primární metody počítáme

- metodu přípustných směrů
- metodu aktivních množin

- metodu projekce gradientu
- metodu redukovaného gradientu
- metody penalizačních a barierových funkcí a s nimi související
- metody vnitřního bodu (interior point methods)

Nejvýznamnější duální metoda je velmi oblíbená a účinná metoda kvadratického programování (SQP - Sequential Quadratic Programming). Někdy se pro tyto metody používá označení přímé a nepřímé metody.

V dalších odstavcích uvedeme základní myšlenky některých metod.

6.4.1 Metody přípustných směrů

Základní myšlenka metody je velmi jednoduchá. Vyjdeme z přípustného bodu $\mathbf{x}_k$. Potom směr $\mathbf{s}_k$ je přípustný směr v bodě $\mathbf{x}_k$, když platí

1. existuje $\bar{\alpha}$ takové, že bod $\mathbf{x}_k + \alpha\mathbf{s}_k$ je přípustný bod pro všechna α v intervalu $0 \leq \alpha \leq \bar{\alpha}$
2. nový bod $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha\mathbf{s}_k$ zajišťuje klesání hodnotící funkce pro nějaké α z určeného intervalu, pak
$$f(\mathbf{x}_k + \alpha\mathbf{s}_k) < f(\mathbf{x}_k)$$

Problémem metody je, že někdy nemusí existovat přípustný směr a metoda v tomto nejjednodušším tvaru není globálně konvergentní.

Pokud jsou ale omezení úlohy lineární, pak přípustný směr lze nalézt pomocí lineárního programování. Tento postup se nazývá **Zoutendijkova metoda**. Mějme tedy problém

$$\min \ \{ \ f(\mathbf{x}) \ : \ \mathbf{A}\mathbf{x} \leq \mathbf{b} \ ; \ \mathbf{x} \geq \mathbf{0} \ \} \tag{6.46}$$

Aby přípustný směr zajišťoval klesání hodnotící funkce, musí svírat tupý úhel s gradientem v příslušném bodě. Proto přípustný směr musí splňovat následující podmínky $\mathbf{A}(\mathbf{x}_k + \mathbf{s}_k) \leq \mathbf{b}$, $\mathbf{x}_k + \mathbf{s}_k \geq \mathbf{0}$, a $\nabla f(\mathbf{x}_k)\ \mathbf{s}_k < 0$. To ale řeší úloha lineárního programování ve tvaru

$$\max_{\mathbf{s}_k} \{ -\nabla f(\mathbf{x}_k)\ \mathbf{s}_k \ ; \ \mathbf{A}\mathbf{s}_k \leq \mathbf{b} - \mathbf{A}\mathbf{x}_k \ , \ \mathbf{s}_k \geq -\mathbf{x}_k \} \tag{6.47}$$

Po nalezení přípustného směru nalezneme další bod pomocí algoritmu jednorozměrového hledání ve směru $\mathbf{s}_k$, tedy řešením úlohy

$$\mathbf{x}_{k+1} = \arg\min_{\alpha} f(\mathbf{x}_k + \alpha\mathbf{s}_k)$$

6.4.2 Metody aktivních množin

Základní myšlenka metod aktivních množin je opět velmi prostá. Omezení ve tvaru nerovnosti rozdělíme na dvě skupiny - aktivní omezení a neaktivní omezení. V aktivních omezeních jsou nerovnosti splněny jako rovnosti a proto musí být brány v úvahu. Neaktivní omezení jsou ta omezení, která naopak nejsou splněna jako rovnosti a mohou proto být úplně ignorována.

Mějme tedy úlohu

$$\min \{ f(\mathbf{x}) \ : \ g_i(\mathbf{x}) \leq 0, \ i = 1, \ldots, m \ \} \tag{6.48}$$

Nechť pro některá i jsou omezení splněna jako rovnosti. Tato i tvoří **množinu aktivních omezení**, kterou označíme A. Ostatní omezení můžeme ignorovat. Naše úloha se změní na

$$\min \{ f(\mathbf{x}) \ : \ g_i(\mathbf{x}) = 0, i \in A \} \tag{6.49}$$

Protože nevíme, která omezení nakonec budou aktivní a která nikoliv, zvolíme si na počátku iteračního řešení nějakou množinu W aktivních omezení, kterou nazveme **pracovní množinou**.

Při řešení předchozího problému vycházíme tedy z nějakého bodu $\mathbf{x}$, který vyhovuje omezením $g_i(\mathbf{x}) = 0, i \in W$ a $g_i(\mathbf{x}) < 0, i \notin W$. Pak hledáme minimum funkce $f(\mathbf{x})$ na pracovním podprostoru určeným množinou aktivních omezení. Přitom mohou nastat dvě možnosti.

1. Při hledání je nutno ověřovat, zda ostatní omezení jsou splněna. Pokud narazíme na hranici určenou omezením, které není v množině aktivních omezení, tak toto omezení je nutno zahrnout do množiny aktivních omezení. Tímto způsobem roste pracovní množina aktivních omezení.

2. Předpokládejme nyní, že jsme nalezli minimum funkce $f(\mathbf{x})$ na pracovní množině aktivních omezení a ostatní omezení, která nejsou v množině aktivních omezení, nejsou porušena. Tento bod označíme $\mathbf{x}_w$, pak

   $$\mathbf{x}_w = \arg\min_{\mathbf{x}} \{ f(\mathbf{x}) \ : \ g_i(\mathbf{x}) = 0, \ i \in W, \ g_i(\mathbf{x}) < 0, \ i \notin W \}$$

   Je třeba nyní rozhodnout, zda jsme v optimu naší původní úlohy, nebo zda vypuštěním některého omezení z množiny aktivních omezení nenalezneme lepší řešení původní úlohy. To můžeme zjistit např. podle znaménka Lagrangeových koeficientů.

   Pokud jsou nezáporné všechny Lagrangeovy koeficienty příslušející pracovní množině aktivních omezení, to znamená $\lambda_i \geq 0$ pro všechny $i \in W$, pak bod $\mathbf{x}_w = \mathbf{x}^*$ je řešením naší původní úlohy (pokud samozřejmě splňuje i ostatní omezení).

   Pokud naopak pro některé $i \in W$ je některý Lagrangeův koeficient $\lambda_i < 0$, pak vypuštěním tohoto i-tého omezení dosáhneme lepší řešení původní úlohy. To plyne z citlivostní věty o významu Lagrangeových koeficientů. Jestliže místo omezení $g_i(\mathbf{x}) = 0$ budeme mít omezení $g_i(\mathbf{x}) = -\alpha < 0$ pro nějaké malé $\alpha > 0$, pak se původní kritérium změní o $(\lambda_i \alpha)$. To znamená, že pro změnu i-tého omezení z aktivního na neaktivní $g_i(\mathbf{x}) = -\alpha < 0$ hodnota kritéria klesne. Tímto způsobem naopak zmenšíme pracovní množinu omezení.

Postupným zvětšováním a zmenšováním pracovní množiny aktivních omezení a minimalizací kritéria na množině pracovních omezení dosáhneme postupně řešení původní úlohy. Minimalizaci na pracovní množině aktivních omezení, to je minimalizaci při omezeních pouze ve tvaru rovnosti, provádíme metodou projekce gradientu nebo metodou redukovaného gradientu. Tyto metody budou popsány v následujících odstavcích.

6.4.3 Metoda projekce gradientu

Uvedeme zde základní myšlenky metody. Přípustný směr se hledá pomocí projekce negativního gradientu na množinu přípustných hodnot určenou omezujícími podmínkami. Tuto metodu publikoval poprvé Rosen v roce 1960.

Nejprve budeme uvažovat pouze lineární omezení, jedná se tedy o problém ve tvaru

$$\min \{f(\mathbf{x}) \ ; \ \mathbf{A}\mathbf{x} \leq \mathbf{b}; \mathbf{x} \geq \mathbf{0}\} \tag{6.50}$$

Předpokládáme, že jsme našli přípustný bod $\mathbf{x}_k$.

Uvažujeme pouze ta omezení, která jsou splněna jako rovnice (pouze aktivní omezení). Lineární omezení můžeme zapsat ve tvaru $\mathbf{a}_i^T\mathbf{x} \leq b_i$, kde vektory $\mathbf{a}_i^T$ jsou rovny i-tým řádkům matice $\mathbf{A}$. Volíme tedy pomocnou matici $\mathbf{Q}$ složenou z těch řádků $\mathbf{a}_i^T$ matice $\mathbf{A}$, ve kterých jsou omezení splněna jako rovnice. V takto vytvořené matici $\mathbf{Q}$ vynecháme lineárně závislé řádky. Matice $\mathbf{Q}$ je pak rozměru $p \times n$ a její hodnost je $p < n$.

Aby nový bod $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha\mathbf{s}$, kde $\alpha > 0$ splňoval omezení, to znamená, aby platilo $\mathbf{A}\mathbf{x}_{k+1} \leq \mathbf{b}$, pak směr $\mathbf{s}$ musí splňovat

$$\mathbf{Q}\mathbf{s} \leq \mathbf{0} \tag{6.51}$$

Dále, aby směr $\mathbf{s}$ byl přípustný (zajišťoval klesání minimalizované funkce $f(\mathbf{x})$), musí svírat tupý úhel s gradientem

$$\nabla f(\mathbf{x}_k)\ \mathbf{s} < 0$$

Obě předchozí podmínky splníme, vezmeme-li za směr $\mathbf{s}$ ortogonální projekci gradientu $\mathbf{g}_k = \nabla f(\mathbf{x}_k)^T$ do lineárního podprostoru generovaného řádky matice $\mathbf{Q}$. Promítáme tedy do ortogonálního doplňku množiny všech vektorů, které lze zapsat ve tvaru $\mathbf{Q}^T\mathbf{q}$.

Záporný gradient rozkládáme na dvě složky - vektory $\mathbf{s}$ a $\mathbf{v}$ - pak

$$-\mathbf{g}_k = \mathbf{s} + \mathbf{v} = \mathbf{s} + \mathbf{Q}^T\mathbf{q} \tag{6.52}$$

Předchozí rovnici násobíme zleva maticí $\mathbf{Q}$, pak

$$-\mathbf{Q}\mathbf{g}_k = \mathbf{Q}\mathbf{s} + \mathbf{Q}\mathbf{Q}^T\mathbf{q}$$

Protože $\mathbf{Q}\mathbf{s} = 0$ (neboť zatím chceme zůstat v určené množině aktivních omezení), pak z předchozí rovnice určíme vektor

$$\mathbf{q} = -\left(\mathbf{Q}\mathbf{Q}^T\right)^{-1}\mathbf{Q}\mathbf{g}_k$$

Odtud hledaný přípustný směr

$$\mathbf{s} = \left[-\mathbf{I} + \mathbf{Q}^T\left(\mathbf{Q}\mathbf{Q}^T\right)^{-1}\mathbf{Q}\right]\mathbf{g}_k = \mathbf{P}\mathbf{g}_k \tag{6.53}$$

kde matice

$$\mathbf{P} = \left[-\mathbf{I} + \mathbf{Q}^T \left(\mathbf{Q}\mathbf{Q}^T\right)^{-1} \mathbf{Q}\right] \tag{6.54}$$

je projekční matice. Pokud podle (6.53) vyjde nenulový směr $\mathbf{s}$, pak platí

$$-\mathbf{g}_k^T\mathbf{s} = \left(-\mathbf{g}_k^T - \mathbf{s}^T + \mathbf{s}^T\right)\mathbf{s} = \mathbf{q}^T\mathbf{Q}\mathbf{s} + \mathbf{s}^T\mathbf{s} = \mathbf{s}^T\mathbf{s} > 0$$

Je to tedy přípustný směr, zajišťující klesání. Pokud podle (6.53) vyjde směr $\mathbf{s} = \mathbf{0}$, pak mohou nastat dvě možnosti:

1. Je-li vektor $\mathbf{q} \geq \mathbf{0}$, pak z (6.52) plyne

   $$-\mathbf{g}_k = \mathbf{Q}^T\mathbf{q} \tag{6.55}$$

   Pro každý jiný přípustný směr $\mathbf{z}$ musí samozřejmě platit $\mathbf{Q}\mathbf{z} \leq 0$ - viz (6.51). Vynásobíme-li tento vztah zleva vektorem $\mathbf{q}^T$, který je dle předpokladu nezáporný, pak

   $$\mathbf{q}^T\mathbf{Q}\mathbf{z} \leq 0, \qquad \Rightarrow \qquad -\mathbf{g}_k^T\mathbf{z} \leq 0.$$

   To znamená, že každý přípustný směr svírá ostrý úhel s gradientem ($\mathbf{g}_k^T\mathbf{z} \geq 0$) a proto bod $\mathbf{x}_k$, ze kterého vycházíme, je bodem minima. Iterační proces hledání tedy končí.

2. Je-li alespoň jedna složka vektoru $\mathbf{q}$ záporná, lze nalézt nový nenulový přípustný směr. Je-li tedy $q_i < 0$, pak vytvoříme novou matici $\bar{\mathbf{Q}}$ tak, že z matice $\mathbf{Q}$ vynecháme i-tý řádek a s maticí $\bar{\mathbf{Q}}$ vypočteme novou projekční matici $\bar{\mathbf{P}}$ a nový směr $\bar{\mathbf{s}} \neq 0$. Je-li více složek vektoru $\mathbf{q}$ záporných, vybereme tu nejzápornější $q_i < 0$.

   Nyní je nutno ověřit, zda nový směr $\bar{\mathbf{s}}$ neporuší omezení. Pro $\mathbf{s} = 0$ platí (6.55). Po vynásobení transpozice rovnice (6.55) zprava vektorem $\bar{\mathbf{s}}$ dostaneme (při respektování vztahu $\bar{\mathbf{Q}}\bar{\mathbf{s}} = 0$)

   $$0 \;\leq\; -\mathbf{g}_k^T\bar{\mathbf{s}} = \mathbf{q}^T\mathbf{Q}\bar{\mathbf{s}} = q_i\mathbf{a}_i\bar{\mathbf{s}}$$

   kde $\mathbf{a}_i$ je i-tý řádek matice $\mathbf{A}$, to je ten, který jsme při tvorbě matice $\bar{\mathbf{Q}}$ z matice $\mathbf{Q}$ vynechali. Protože dle předpokladu je $q_i < 0$, pak musí být $\mathbf{a}_i\bar{\mathbf{s}} < 0$ a proto nový směr $\bar{\mathbf{s}}$ neporuší omezení.

Známe-li přípustný směr $\mathbf{s} \neq \mathbf{0}$, určíme délku kroku α_k tak, že určíme $\bar{\alpha}$ podle

$$\bar{\alpha} = \max\{\alpha \;:\; \mathbf{x}_k + \alpha\mathbf{s} \quad \text{přípustné}\}$$

pak optimální α_k určíme z

$$\alpha_k = \arg\min_{\alpha}\{f(\mathbf{x}_k + \alpha\mathbf{s}) \;:\; 0 \leq \alpha \leq \bar{\alpha}\}$$

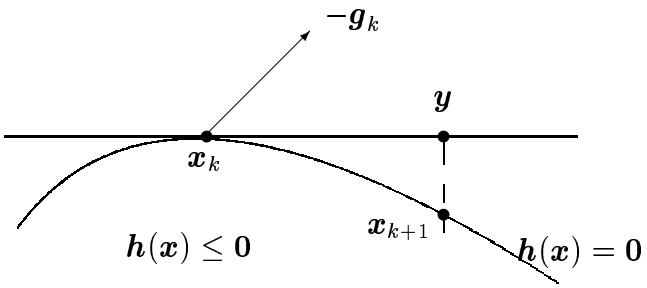
Pokud $\alpha_k = \bar{\alpha}$, pak jsme zřejmě narazili na další omezení, které se tedy stalo aktivním omezením. Proto při další iteraci toto omezení musíme přidat do množiny aktivních omezení.

Pokud jsou omezení nelineární, ve tvaru $\bar{\mathbf{h}}(\mathbf{x}) \leq \mathbf{0}$, pak je v bodě $\mathbf{x}_k$ linearizujeme, určíme aktivní omezení a vytvoříme projekci záporného gradientu funkce $f(\mathbf{x}_k)$ do tečného

podprostoru určeného těmito aktivními omezeními - viz obr. 6.10. Jsou-li aktivní omezení určena vztahem $\mathbf{h}(\mathbf{x}_k) = 0$ je projekční matice $\mathbf{P}_k$ analogicky ke vztahu (6.54) určena

$$\mathbf{P} = \left[-\mathbf{I} + \nabla\mathbf{h}(\mathbf{x}_k)^T \left(\nabla\mathbf{h}(\mathbf{x}_k) \nabla\mathbf{h}(\mathbf{x}_k)^T \right)^{-1} \nabla\mathbf{h}(\mathbf{x}_k) \right]. \tag{6.56}$$

Při tom může nastat situace znázorněná na obr. 6.10. Vlivem křivosti nelineárních omezení padne takto získaný bod, (který označíme jako bod $\mathbf{y}$), mimo omezení, nesplňuje tedy podmínku $\mathbf{h}(\mathbf{y}) = 0$.



Obrázek 6.10: Projekce bodu $\mathbf{y}$ na množinu omezení

Proto musíme postupovat následovně: Nejprve tedy získáme bod $\mathbf{y}$ pomocí projekce negativního gradientu do linearizace aktivních omezení - pomocí projekční matice $\mathbf{P}$ dle (6.56). Potom je třeba ve směru kolmém k tečné nadrovině promítnout bod $\mathbf{y}$ do množiny omezení.

Hledáme tedy bod $\mathbf{x}_{k+1} = \mathbf{y} + \nabla\mathbf{h}(\mathbf{x}_k)^T\alpha$ takový, aby $\mathbf{h}(\mathbf{x}_{k+1}) = \mathbf{0}$. To nalezneme linearizací omezení. V bodě $\mathbf{x}_k$ platí

$$\mathbf{h}(\mathbf{x}_{k+1}) = \mathbf{h}\left(\mathbf{y} + \nabla\mathbf{h}(\mathbf{x}_k)\alpha\right) \doteq \mathbf{h}(\mathbf{y}) + \nabla\mathbf{h}(\mathbf{x}_k)\nabla\mathbf{h}(\mathbf{x}_k)^T\alpha$$

Aproximace platí pro $|\alpha|$ i $|\mathbf{y} - \mathbf{x}_k|$ malé. Z podmínky $\mathbf{h}(\mathbf{x}_{k+1}) = \mathbf{0}$ určíme

$$\alpha = -\left[\nabla\mathbf{h}(\mathbf{x}_k)\nabla\mathbf{h}(\mathbf{x}_k)^T\right]^{-1}\mathbf{h}(\mathbf{y})$$

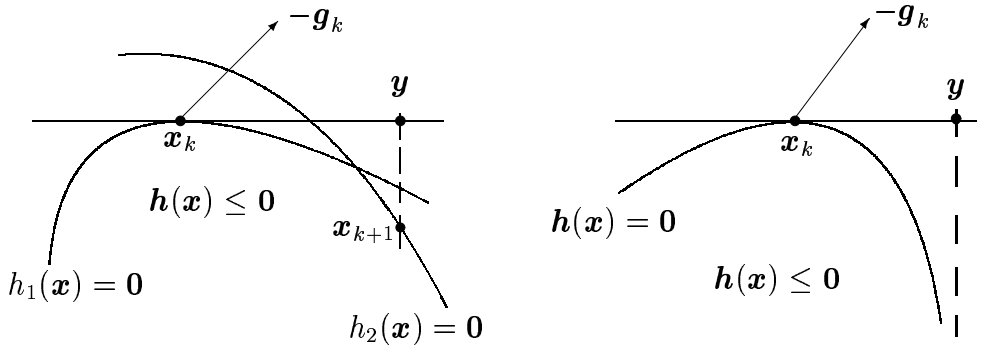
a další iteraci podle vztahu

$$\mathbf{x}_{k+1} = \mathbf{y} - \nabla\mathbf{h}(\mathbf{x}_k)^T\left[\nabla\mathbf{h}(\mathbf{x}_k)\nabla\mathbf{h}(\mathbf{x}_k)^T\right]^{-1}\mathbf{h}(\mathbf{y})$$

Je zřejmé, že se během iteračního postupu mohou uplatnit další omezení - viz obr. 6.11a, případně průmět bodu $\mathbf{y}$ do množiny omezení nemusí existivat - viz obr. 6.11b. Proto při implementaci metody je třeba provést řadu modifikací. Bylo zjištěno, že metoda má lineární konvergenci.

6.4.4 Metoda redukovaného gradientu

Metoda redukovaného gradientu byla v roce 1963 navržena Wolfem pro lineární omezení a v roce 1965 Carpentierem pro nelineární omezení. Metoda souvisí se simplexovou metodou



Obrázek 6.11: Problémy při projekci bodu $\mathbf{y}$ na množinu omezení

lineárního programování a s metodou projekce gradientu. Ve shodě se simplexovou metodou se i v této metodě proměnné dělí na bázové a nebázové.

Uvažujme nejprve lineární omezení. Problém bude potom ve tvaru

$$\min \{f(\mathbf{x}) \ : \ \mathbf{A}\mathbf{x} = \mathbf{b}, \ \mathbf{x} \ge \mathbf{0}\} \tag{6.57}$$

Lineární omezení, kterých nechť je m, předpokládáme ve standardním tvaru. Předpokládáme, že problém není degenerovaný, což znamená, že libovolná množina m řádků matice $\mathbf{A}$ je lineárně nezávislá. Potom libovolné přípustné řešení má nejvýše $n - m$ proměnných nulových.

Vektor proměnných rozdělíme na dvě části $\mathbf{x} = [\mathbf{u}^T, \mathbf{v}^T]^T$, kde subvektor $\mathbf{u}$ má dimenzi m. Obdobně rozdělíme matici $\mathbf{A}$ na dvě submatice $\mathbf{B}$ a $\mathbf{C}$ podle vztahu $\mathbf{A} = [\mathbf{B}, \mathbf{C}]$ kde submatice $\mathbf{B}$ je čtvercová regulární matice rozměru $m \times m$. Původní problém má nyní tvar

$$\min \{f(\mathbf{u}, \mathbf{v}) \ : \ \mathbf{B}\mathbf{u} + \mathbf{C}\mathbf{v} = \mathbf{b}, \ \mathbf{u} \ge \mathbf{0}, \ \mathbf{v} \ge \mathbf{0}\} \tag{6.58}$$

Z předchozího omezení můžeme proměnnou $\mathbf{u}$ vypočítat

$$\mathbf{u} = \mathbf{B}^{-1}\mathbf{b} - \mathbf{B}^{-1}\mathbf{C}\mathbf{v}$$

Potom přírůstek kritéria vyjádříme pomocí gradientu kritéria vzhledem k proměnné $\mathbf{v}$, to je pomocí tzv. redukovaného gradientu

$$\begin{aligned} df(\mathbf{u}, \mathbf{v}) &= \frac{\partial f(\mathbf{u}, \mathbf{v})}{\partial \mathbf{u}} d\mathbf{u} + \frac{\partial f(\mathbf{u}, \mathbf{v})}{\partial \mathbf{v}} d\mathbf{v} = \nabla_u f(\mathbf{u}, \mathbf{v}) d\mathbf{u} + \nabla_v f(\mathbf{u}, \mathbf{v}) d\mathbf{v} \\ &= \nabla_v f(\mathbf{u}, \mathbf{v}) d\mathbf{v} - \nabla_u f(\mathbf{u}, \mathbf{v}) \mathbf{B}^{-1}\mathbf{C} d\mathbf{v} \\ &= \left[\nabla_v f(\mathbf{u}, \mathbf{v}) - \nabla_u f(\mathbf{u}, \mathbf{v}) \mathbf{B}^{-1}\mathbf{C}\right] d\mathbf{v} \end{aligned}$$

Redukovaný gradient funkce $f(\mathbf{u}, \mathbf{v})$, který označíme $\mathbf{g}^{(r)}$ je vektor rozměru $n - m$ a je roven

$$\mathbf{g}^{(r)} = \left(\frac{df(\mathbf{u}, \mathbf{v})}{d\mathbf{v}}\right)^T = \left[\nabla_v f(\mathbf{u}, \mathbf{v}) - \nabla_u f(\mathbf{u}, \mathbf{v}) \mathbf{B}^{-1}\mathbf{C}\right]^T \tag{6.59}$$

Nutné podmínky prvního řádu pro optimalitu jsou splněny právě tehdy, když

$$\begin{aligned} g_i^{(r)} &= 0, \quad \text{pro všechny} \quad v_i > 0 \\ g_i^{(r)} &\ge 0, \quad \text{pro všechny} \quad v_i = 0. \end{aligned}$$

To je zřejmé z následující úvahy:
Přírůstek kritéria je zřejmě $df(\mathbf{u},\mathbf{v}) = \left[\mathbf{g}^{(r)}\right]^T d\mathbf{v}$. Podle nutných podmínek je přírůstek nezáporný v optimálním bodě.

Pokud je $v_i > 0$, pak přírůstek $d\mathbf{v}$ může být libovolný a proto musí být v optimu odpovídající složka gradientu nulová. Je-li naopak $v_i = 0$, pak přírůstek $d\mathbf{v}$ musí být pouze nezáporný (pro dodržení omezení) a proto musí být v optimu odpovídající složka gradientu nezáporná.

Nyní určíme tzv. promítnutý redukovaný gradient $\mathbf{s}$ jehož složky jsou

$$s_i = \begin{cases} 0 & \text{pro } \; v_i = 0, \text{ a } \; \mathbf{g}_i^{(r)} > 0 \\ -\mathbf{g}_i^{(r)} & \text{jinak} \end{cases}$$

Ve směru $\mathbf{s}$ provádíme jednorozměrovou minimalizaci a určujeme nové body

$$\begin{aligned} \mathbf{v}_{k+1} &= \mathbf{v}_k + \alpha\mathbf{s} \\ \mathbf{u}_{k+1} &= \mathbf{u}_k - \alpha\mathbf{B}^{-1}\mathbf{C}\mathbf{s} \end{aligned}$$

a optimální

$$\alpha^* = \arg\min_{\alpha} f(\mathbf{u}_{k+1}, \mathbf{v}_{k+1})$$

Pokud $\mathbf{u}_{k+1}$ a $\mathbf{v}_{k+1}$ jsou přípustné body, pak pokračujeme v nové iteraci výpočtem nového redukovaného gradientu atd.

Pokud při jednorozměrové minimalizaci narazíme pro nějaké $\bar{\alpha} < \alpha^*$ na omezení, (některá složka vektorů $\mathbf{u}_{k+1}$ nebo $\mathbf{v}_{k+1}$ se stane nulovou), pak musíme rozlišovat dva případy:

1. pokud se stane nulovou některá složka vektoru $\mathbf{v}_{k+1}$, pak položíme $\alpha^* := \bar{\alpha}$ a pokračujeme další iterací.

2. pokud se stane nulovou složka vektoru $\mathbf{u}_{k+1}$, pak položíme opět $\alpha^* := \bar{\alpha}$, ale současně musíme změnit rozdělení vektoru $\mathbf{x}$ na dva subvektory $\mathbf{u}$ a $\mathbf{v}$ tak, že příslušnou nulovou složku vektoru $\mathbf{u}_{k+1}$ přesuneme do množiny složek vektoru $\mathbf{v}$ a naopak některou nenulovou složku vektoru $\mathbf{v}_{k+1}$ zase přesuneme do množiny složek vektoru $\mathbf{u}$. Proto musíme odpovídajícím způsobem změnit rozdělení matice $\mathbf{A}$ na submatice $\mathbf{B}$ a $\mathbf{C}$.

Pokud jsou omezení nelineární, problém je ve tvaru

$$\min\{f(\mathbf{x}) \; : \; \mathbf{g}(\mathbf{x}) = \mathbf{0}, \; \mathbf{x} \geq \mathbf{0}\} \tag{6.60}$$

Vektor proměnných opět rozdělíme na dvě části $\mathbf{x} = [\mathbf{u}^T, \mathbf{v}^T]^T$, kde subvektor $\mathbf{u}$ má dimenzi m, která je rovna počtu omezení.

Nelineární omezení v bodě $\mathbf{x}_k$ linearizujeme. Z omezení $\mathbf{g}(\mathbf{x}_k) = \mathbf{0}$ plyne pro přírůstky

$$d\mathbf{g} = \frac{\partial \mathbf{g}}{\partial \mathbf{u}} d\mathbf{u} + \frac{\partial \mathbf{g}}{\partial \mathbf{v}} d\mathbf{v} = \mathbf{0}.$$

Odtud

$$d\mathbf{u} = -\left[\frac{\partial \mathbf{g}}{\partial \mathbf{u}}\right]^{-1} \frac{\partial \mathbf{g}}{\partial \mathbf{v}} d\mathbf{v}$$

Přírůstek kritéria je potom roven

$$\begin{aligned} d\ f(\mathbf{u},\mathbf{v}) &= \nabla_u f(\mathbf{u},\mathbf{v})\ d\mathbf{u} + \nabla_v f(\mathbf{u},\mathbf{v})\ d\mathbf{v} \\ &= \nabla_v f(\mathbf{u},\mathbf{v})\ d\mathbf{v} - \nabla_u f(\mathbf{u},\mathbf{v}) \left[\frac{\partial \mathbf{g}}{\partial \mathbf{u}}\right]^{-1} \frac{\partial \mathbf{g}}{\partial \mathbf{v}} d\mathbf{v} \end{aligned}$$

Redukovaný gradient funkce $f(\mathbf{u},\mathbf{v})$, který opět označíme $\mathbf{g}^{(r)}$ je vektor rozměru $n-m$ a je roven

$$\mathbf{g}^{(r)} = \left(\frac{df(\mathbf{u},\mathbf{v})}{d\mathbf{v}}\right)^T = \left[\nabla_v f(\mathbf{u},\mathbf{v}) - \nabla_u f(\mathbf{u},\mathbf{v}) \left[\frac{\partial \mathbf{g}}{\partial \mathbf{u}}\right]^{-1} \frac{\partial \mathbf{g}}{\partial \mathbf{v}}\right]^T \tag{6.61}$$

Další postup je stejný jako v lineárním případě.

Metody pokutových a bariérových funkcí

Tyto metody problém optimalizace s omezením aproximují problémem optimalizace bez omezení.

Penalizační metody aproximují původní problém s omezením problémem bez omezení, při čemž za vybočení z omezení platíme "penále", které se přidává k původnímu kritériu. Penále či pokuta je v tomto případě **vnější pokutová funkce**.

Bariérové metody aproximují původní problém s omezením problémem bez omezení přidáním členu, který nás penalizuje, pokud se blížíme k hranici omezení. V tomto případě se jedná o **vnitřní pokutovou funkci**.

6.4.5 Metody pokutových funkcí

Mějme tedy problém optimalizace s omezením

$$\min\{f(\mathbf{x}) \ : \ \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\} \tag{6.62}$$

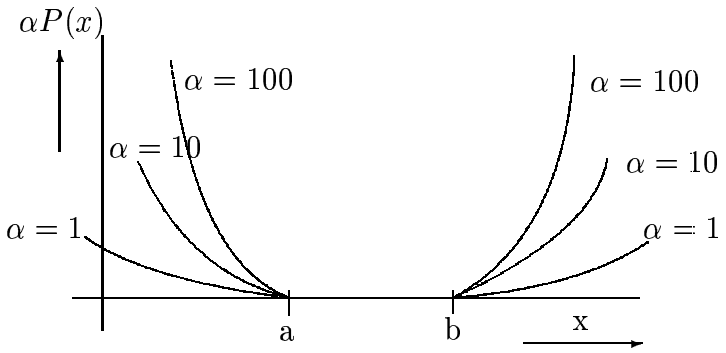
Definujeme si penalizační (pokutovou) funkci $P(t)$ s vlastností $P(t) > 0$ pro $t > 0$ a $P(t) = 0$ pro $t \leq 0$. Pomocí penalizační funkce $P(t)$ aproximuji předchozí problém optimalizace s omezením problémem bez omezení ve tvaru

$$\min\left\{f(\mathbf{x}) + \alpha \sum_{i=1}^{m} P(g_i(\mathbf{x}))\right\} \tag{6.63}$$

kde α je dostatečně velká kladná konstanta. Funkce $P(t)$ může být na příklad rovna

$$P(t) = \max[0,\ t]^2$$

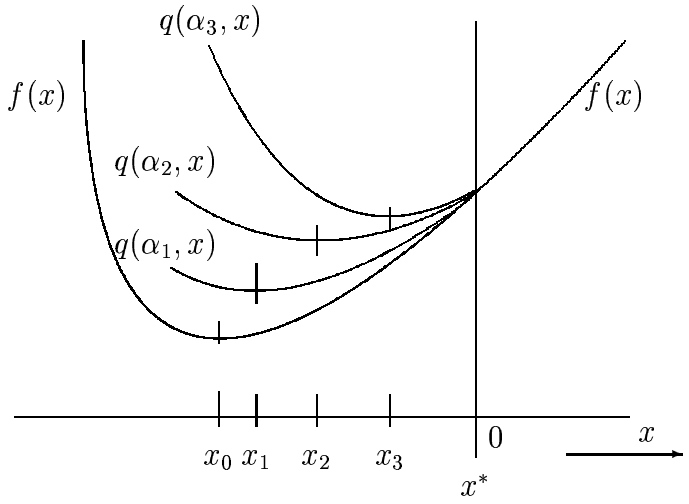
Pro omezení ve tvaru rovnosti můžeme volit pokutovou funkci $P(t) = t^2$. Pokud α bude hodně velké, tak omezení budou přibližně splněna a pro $\alpha \to \infty$ bude řešení konvergovat k řešení původního problému s omezením.



Obrázek 6.12: Pokutové funkce $\alpha P(x)$ při omezení $a \leq x \leq b$.

Postup řešení je takový, že pro rostoucí posloupnost koeficientů $\{\alpha_j\}$, kde $\alpha_{j+1} > \alpha_j > 0$ opakovaně řešíme problém bez omezení s penalizační funkcí, to je problém

$$\min_{x \in R^n} \{q(\alpha_j, \mathbf{x})\} = \min_{x \in R^n} \left\{ f(\mathbf{x}) + \alpha_j \sum_{i=1}^{m} P(g_i(\mathbf{x})) \right\} \tag{6.64}$$



Obrázek 6.13: Posloupnost řešení problému s pokutovou funkcí při omezení $x \geq 0$.

Každý subproblém má řešení, které označíme $\mathbf{x}_j$. Při tom platí řada zajímavých tvrzení

1. Pro posloupnost řešení jednotlivých problémů platí

$$\begin{aligned} q(\alpha_j, \mathbf{x}_j) &\leq q(\alpha_{j+1}, \mathbf{x}_{j+1}) \\ P(\mathbf{x}_j) &\geq P(\mathbf{x}_{j+1}) \\ f(\mathbf{x}_j) &\leq f(\mathbf{x}_{j+1}) \end{aligned}$$

2. Je-li $\mathbf{x}^*$ optimální řešení původní úlohy s omezením, pak

$$f(\mathbf{x}^*) \geq q(\alpha_j, \mathbf{x}_j) \geq f(\mathbf{x}_j)$$

3. Jestliže $\mathbf{x}_j$ je posloupnost generovaná penalizační metodou, pak limitní bod této posloupnosti je řešením původní úlohy

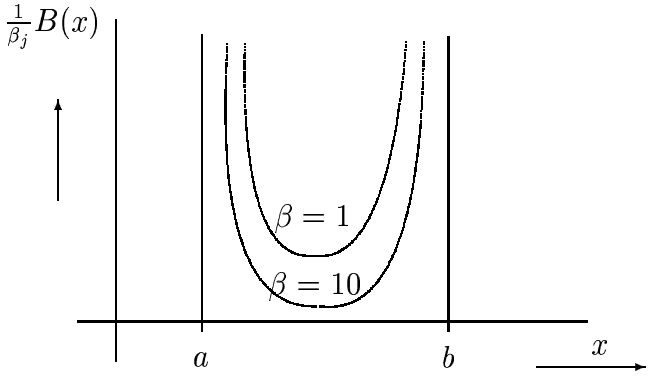
$$\lim_{j\to\infty} f(\mathbf{x}_j) = f(\mathbf{x}^*)$$

Dokažte předchozí nerovnosti a nakreslete obrázek demonstrující uvedené nerovnosti - viz obr. 6.13.

Je třeba poznamenat, že penalizační metody jsou sice velmi jednoduché a v praxi dobře použitelné, ale penalizační funkce obecně velmi zhoršují rychlost konvergence použitých algoritmů. K minimalizaci problému bez omezení je možno použít libovolnou numerickou metodu.

6.4.6 Metody bariérových funkcí

V bariérových metodách platíme penále, když se blížíme k hranici omezení - viz obr. 6.14. Přibližování se tedy děje z vnitřního bodu množiny přípustných hodnot. To znamená, že bariérová metoda je použitelná pouze tehdy, má-li množina přípustných hodnot vnitřní body. Metoda bariérových funkcí nutně musí startovat z vnitřního bodu, takže již při startu metody může nastat problém s nalezením vnitřního bodu.



Obrázek 6.14: Bariérová funkce $\frac{1}{\beta_j}B(x)$, při omezení $a \le x \le b$.

Mějme tedy optimalizační problém

$$\min\{f(\mathbf{x}) \ : \ \mathbf{g}(\mathbf{x}) \le 0\} \tag{6.65}$$

Bariérová funkce $B(\mathbf{x})$ je definovaná na množině $\mathbf{X} = \{\mathbf{x} \ : \mathbf{g}(\mathbf{x}) \le \mathbf{0}\}$. Přitom je to funkce nezáporná a spojitá a $B(\mathbf{x}) \to \infty$, pokud bod $\mathbf{x}$ se blíží ke hranici omezení.

Bariérová funkce může být na příklad rovna

$$B(t) = -\frac{1}{t}, \quad \text{nebo} \quad B(t) = -\log(-t) \tag{6.66}$$

Postup řešení je takový, že pro rostoucí posloupnost koeficientů $\{\beta_j\}$, kde $\beta_{j+1} > \beta_j > 0$ řešíme problém bez omezení s bariérovou funkcí, to je problém

$$\min_{x\in R^n}\left\{f(\mathbf{x}) + \frac{1}{\beta_j}\sum_{i=1}^{m} B(g_i(\mathbf{x}))\right\} \tag{6.67}$$

Každý subproblém má pro určité β_j řešení, které označíme jako $\mathbf{x}_j$. Při tom platí, že limita posloupnosti řešení $\mathbf{x}_j$ pro $\beta \to \infty$ generovaná bariérovou metodou je řešením původní úlohy. Musíme ale zajistit, že při iteračním výpočtu neporušíme omezení, neboť mimo omezení není bariérová funkce definována, případně bariérovou funkci musíme dodefinovat $B(t) = \infty$ pro $\mathbf{x} \notin \mathbf{X}$.

Kombinace penalizační a bariérové funkce byla použita v metodě **SUMT** (Sequential Unconstrained Minimization Technique) - techniky, která využívá sekvenční minimalizaci bez omezení k řešení problémů minimalizace s omezením. Autoři této metody byli Fiacco a McCormick v šedesátých letech. Optimalizační problém s omezením ve tvaru

$$\min \{f(\mathbf{x}) \ : \ \mathbf{g}(\mathbf{x}) \leq \mathbf{0}, \ \mathbf{h}(\mathbf{x}) = \mathbf{0}\} \tag{6.68}$$

můžeme převést na sekvenční řešení problému bez omezení ve tvaru

$$\min \left\{ f(\mathbf{x}) - \frac{1}{\beta_j} \sum_i \log(-g_i(\mathbf{x})) + \alpha_j \sum_i (h_i(\mathbf{x}))^2 \right\} \tag{6.69}$$

pro rostoucí množinu koeficientů α_j a β_j. Zde se používá kombinace penalizační i bariérové metody.

Bariérové metody se staly základem moderních metod zvaných **metody vnitřního bodu**, kterým věnujeme následující odstavec.

6.4.7 Metody vnitřního bodu

Metody vnitřního bodu (interior point methods) byly použity poprvé Karmarkarem v roce 1984 pro řešení problému lineárního programování. Tyto metody vycházejí z metody SUMT. V roce 1988 byly rozšířeny na obecný problém Něstěrovem a Němirovským.

Mějme konvexní množinu omezení $g_i(\mathbf{x}) \leq 0, \ i = 1, \ldots, m$, kde funkce g_i jsou konvexní a hladké. Uvažujme nyní neprázdnou množinu ostrých omezení

$$\mathbf{X} = \{\mathbf{x} \ : \ g_i(\mathbf{x}) < 0, \ i = 1, \ldots, m\}.$$

Na této množině si definujeme logaritmickou bariérovou funkci

$$B(\mathbf{x}) = \begin{cases} -\sum_{i=1}^{m} \log(-g_i(\mathbf{x})), & \mathbf{x} \in \mathbf{X} \\ \infty & \mathbf{x} \notin \mathbf{X} \end{cases}$$

Bariérová funkce B je konvexní a hladká na množině $\mathbf{X}$ a blíží se nekonečnu, když se bod $\mathbf{x}$ blíží ke hranici množiny $\mathbf{X}$.

Nyní určíme bod $\mathbf{x}^{(c)}$, který je bodem minima bariérové funkce B($\mathbf{x}$)

$$\mathbf{x}^{(c)} = \arg \min_{\mathbf{x} \in \mathbf{X}} B(\mathbf{x}) = \arg \max_{\mathbf{x} \in \mathbf{X}} \Pi_{i=1}^{m} (-g_i(\mathbf{x})) \tag{6.70}$$

Poznámka: Uvědomme si, že platí

$$\begin{aligned} \mathbf{x}^{(c)} &= \arg \min_x \left[-\sum_i \log(-g_i(\mathbf{x})) \right] \\ &= \arg \max_x \left[\log \Pi_i(-g_i(\mathbf{x})) \right] \\ &= \arg \max_x \left[\Pi_i(-g_i(\mathbf{x})) \right] \end{aligned}$$

□

Bod $\mathbf{x}^{(c)}$ se nazývá **analytický střed nerovností** $g_i(\mathbf{x}) < 0$. Tento bod existuje, pokud je množina $\mathbf{X}$ ohraničená. Je to bod, který je vlastně robustním řešením množiny nerovností. Je to vnitřní bod množiny $\mathbf{X}$, který je v jistém smyslu nejvíce vzdálen od hranice oblasti.

Bod $\mathbf{x}^{(c)}$ můžeme určit Newtonovou metodou za předpokladu, že známe přísně přípustný počátení bod.

Tak například pro lineární nerovnosti $\mathbf{a}_i^T\mathbf{x} \leq b_i,\ i = 1, \ldots, m$, je bariérová funkce $B(\mathbf{x}) = \sum_{i=1}^{m} \log\left(b_i - \mathbf{a}_i^T\mathbf{x}\right)^{-1}$. Konstanta $b_i - \mathbf{a}_i^T\mathbf{x} > 0$ je rovna "nesplnění" i-té rovnosti.

Mějme opět optimalizační problém

$$\min\{f(\mathbf{x}) \ :\ g_i(\mathbf{x}) \leq 0,\ i = 1, \ldots, m\} \tag{6.71}$$

kde funkce g_i jsou hladké a konvexní. Pro $\alpha > 0$ definujme funkci

$$\alpha f(\mathbf{x}) + B(\mathbf{x}) = \alpha f(\mathbf{x}) - \sum_{i=1}^{m} \log\left(-g_i(\mathbf{x})\right) \tag{6.72}$$

Tato funkce je hladká a konvexní. Je zdola omezená za předpokladu, že množina $\{\mathbf{x} \ :\ f(\mathbf{x}) \leq \gamma,\ g_i(\mathbf{x}) \leq 0\}$ je pro nějaké γ ohraničená. Koeficient α je vlastně relativní váha mezi kritériem a bariérovou funkcí.

Nyní si definujeme funkci

$$\mathbf{x}^*(\alpha) = \arg\min_{\mathbf{x}\in\mathbf{X}} \left(\alpha f(\mathbf{x}) + B(\mathbf{x})\right) \tag{6.73}$$

Funkce $\mathbf{x}^*(\alpha)$ pro $\alpha > 0$ se nazývá **centrální cesta**. Pro $\alpha = 0$ je bod $\mathbf{x}^*(0)$ analytický střed nerovností a pro $\alpha \to \infty$ vede tato cesta k řešení původního problému (6.71). Body $\mathbf{x}^*(\alpha)$ pro určité α můžeme opět určit Newtonovou metodou, ovšem opět za předpokladu, že je dán přísně přípustný počáteční bod (vnitřní bod množiny omezení).

Metoda řešení optimalizačního problému bez omezení je tedy jednoduchá:
Pro určité $\alpha > 0$ a přísně přípustný bod $\mathbf{x}$ určíme Newtonovou metodou bod $\mathbf{x}^*(\alpha)$, přičemž startujeme z bodu $\mathbf{x}$. Další iteraci provedeme z nového počátečního bodu $\mathbf{x} = \mathbf{x}^*(\alpha)$, který je roven právě vypočtenému bodu na centrální cestě. Pro nové (zvětšené) $\alpha := \alpha\beta$, kde $\beta > 1$ spočteme nový bod na centrální cestě. Tímto způsobem získáme posloupnost bodů na centrální cestě, které odpovídají rostoucí hodnotě parametru α. Posloupnost bodů na centrální cestě, které získáme řešením minimalizačního problému bez omezení, vede na řešení našeho původního problému s omezením. Uvědomme si, že složitost metody neroste s počtem omezení. To je podstatný přínos této metody (vlastně všech penalizačních a bariérových metod).

Při tom se vyskytují problémy s volbou počáteční hodnoty parametru α a s volbou koeficientu β, který zajišťuje růst koeficientu α v další iteraci. Tato volba znamená kompromis, neboť pomalý růst koeficientu α, znamená menší počet kroků Newtonovy metody v jedné iteraci, ale celková konvergence metody k řešení je pomalejší.

Nyní ukážeme na souvislost bodů na centrální cestě s dualitou. Nutné podmínky pro bod $\mathbf{x}^*(\alpha) = \arg\min(\alpha f(\mathbf{x}) + B(\mathbf{x}))$ jsou

$$\alpha \nabla f(\mathbf{x}^*(\alpha)) + \sum_{i=1}^{m} \frac{1}{-g_i(\mathbf{x}^*(\alpha))} \nabla g_i(\mathbf{x}^*(\alpha)) = \mathbf{0} \tag{6.74}$$

Předchozí rovnici přepíšeme do tvaru

$$\nabla f(\mathbf{x}^*(\alpha)) + \sum_{i=1}^{m} \lambda_i \nabla g_i(\mathbf{x}^*(\alpha)) = \mathbf{0}, \qquad \lambda_i = \frac{1}{-\alpha g_i(\mathbf{x}^*(\alpha))} > 0 \tag{6.75}$$

Z předchozí rovnice plyne, že $\mathbf{x}^*(\alpha)$ minimalizuje Lagrangeovu funkci

$$L(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}^*(\alpha)) + \sum_{i=1}^{m} \lambda_i \, g_i(\mathbf{x}^*(\alpha)) \tag{6.76}$$

Z teorie duality víme, že minimum primárního problému není menší než řešení duálního problému, čili

$$\begin{aligned} f^* \;\geq\; \psi(\boldsymbol{\lambda}) &= \min_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}) = \min_{\mathbf{x}} \left(f(\mathbf{x}) + \sum_i \lambda_i g_i(\mathbf{x}) \right) \\ &= f(\mathbf{x}^*(\alpha)) + \sum_{i=1}^{m} \lambda_i g_i(\mathbf{x}^*(\alpha)) \\ &= f(\mathbf{x}^*(\alpha)) - \frac{m}{\alpha} \end{aligned} \tag{6.77}$$

kde jsme optimum původního problému označili f^*.

To znamená, že bod na centrální cestě nám poskytuje dolní odhad

$$f(\mathbf{x}^*(\alpha)) \;\geq\; f^* \;\geq\; f(\mathbf{x}^*(\alpha)) - \frac{m}{\alpha} \tag{6.78}$$

To můžeme použít při odhadu přesnosti dosaženého řešení.

Nyní srovnáme nutné podmínky optimality původního problému (6.71) s nutnými podmínkami (6.74) pro bod na centrální cestě. Pro konvexní optimalizační problém

$$\min \{ f(\mathbf{x}) \;:\; g_i(\mathbf{x}) \leq 0, \; i = 1, \ldots, m \} \tag{6.79}$$

platí, že bod $\mathbf{x}^*$ je bod optima, právě když existuje vektor $\boldsymbol{\lambda} \geq 0$, že platí (Kuhnovy - Tuckerovy podmínky)

$$\begin{aligned} g_i(\mathbf{x}) &\leq 0 \\ \nabla f(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i \nabla g_i(\mathbf{x}^*) &= \mathbf{0} \\ -\lambda_i g_i(\mathbf{x}^*) &= 0 \end{aligned} \tag{6.80}$$

Z předchozího výkladu plyne, že bod $\mathbf{x}^*(\alpha)$ na centrální cestě vyhovuje nutným a postačujícím podmínkám v následujícím tvaru. Existuje vektor $\boldsymbol{\lambda} \geq \mathbf{0}$, že platí

$$\begin{aligned} g_i(\mathbf{x}) &\leq 0 \\ \nabla f(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i \nabla g_i(\mathbf{x}^*) &= \mathbf{0} \\ -\lambda_i g_i(\mathbf{x}^*) &= \frac{1}{\alpha} \end{aligned} \tag{6.81}$$

Odtud plyne, že centrální cesta je spojitá deformace podmínek optimality pro původní problém.

Metoda středů

Modifikací předchozí metody je metoda středů (Method of Centers). Pro konvexní optimalizační problém (6.71) si zvolíme konstantu $\gamma > f^*$, kde f^* je optimální hodnota kriteriální funkce. Definujeme body $\mathbf{x}^*(\gamma)$, které jsou analytickými středy nerovností $f(\mathbf{x}) < \gamma$, a $g_i(\mathbf{x}) < 0$. Platí

$$\mathbf{x}^*(\gamma) = \arg\min_{\mathbf{x}\in\mathbf{X}} \left(-\log(\gamma - f(\mathbf{x})) - \sum_{i=1}^{m} \log(-g_i(\mathbf{x})) \right) \tag{6.82}$$

Bod $\mathbf{x}^*(\gamma)$ splňuje nutné podmínky ve tvaru

$$\frac{1}{\gamma - f(\mathbf{x}^*(\gamma))} \nabla f(\mathbf{x}^*(\gamma)) + \sum_{i=1}^{m} \frac{1}{-g_i(\mathbf{x}^*(\alpha))} \nabla g_i(\mathbf{x}^*(\alpha)) = \mathbf{0} \tag{6.83}$$

Odtud plyne, že bod $\mathbf{x}^*(\gamma)$ je na centrální cestě

$$\mathbf{x}^*(\gamma) = \mathbf{x}^*(\alpha), \qquad \alpha = \frac{1}{(\gamma - f(\mathbf{x}^*(\gamma))} \tag{6.84}$$

Poznámka: Body $\mathbf{x}^*$ rozlišujeme pouze jejich argumenty. To znamená, že body $\mathbf{x}^*(\gamma)$ jsou definovány v (6.82) a body $\mathbf{x}^*(\alpha)$ jsou definovány v (6.73).

□

Proto body $\mathbf{x}^*(\gamma)$, $\gamma > f^*$ také parametrizují centrální cestu a tudíž poskytují také dolní odhad ve tvaru

$$f^* \geq f(\mathbf{x}^*(\gamma)) - m\,(\gamma - f(\mathbf{x}^*(\gamma))) \tag{6.85}$$

Algoritmus metody středů je následující:

- Zvolíme počáteční bod $\mathbf{x}$, parametr γ, koeficient θ a zvolenou přesnost výpočtu (toleranci) ε, které vyhovují

$$\begin{array}{lll} \mathbf{x} & \in & \mathbf{X} \\ \gamma & > & f(\mathbf{x}) \end{array} \qquad \begin{array}{lll} 0 & < \theta & < 1 \\ & \varepsilon & > 0 \end{array}$$

- Pro počáteční bod $\mathbf{x}$ vypočteme Newtonovou iterační metodou nový bod $\mathbf{x}^*(\gamma)$ podle (6.82).
  Jestliže
  $$m\,(\gamma - f(\mathbf{x}^*(\gamma)) < \varepsilon, \tag{6.86}$$
  pak je splněna zvolená přesnost výpočtu a bod $\mathbf{x}^*(\gamma)$ je s danou přesností řešením našeho problému.

- Pokud zvolená přesnost není dosažena, pak s novou počáteční podmínkou $\mathbf{x} = \mathbf{x}^*(\gamma)$ znova Newtonovou metodou řešíme problém (6.82) s novou hodnotou parametru γ
  $$\gamma := (1 - \theta) f(\mathbf{x}) + \theta\gamma \tag{6.87}$$

Koeficient θ je koeficient konvexní kombinace horního odhadu γ kritéria a skutečné hodnoty kritéria $f(\mathbf{x})$. Čím je koeficient θ menší, tím více iteračních kroků Newtonovy metody je třeba k vyřešení problému (6.82).

Výpočty bylo ověřeno, že zrychlení výpočtu je dosaženo, když použijeme $q > 1$ kopií omezení $f(\mathbf{x}) < \gamma$. Potom Newtonovou metodou řešíme problém

$$\mathbf{x}^*(\gamma) = \arg\min_{\mathbf{x}\in\mathbf{X}} \left(-q\log(\gamma - f(\mathbf{x})) - \sum_{i=1}^{m} \log(-g_i(\mathbf{x})) \right) \tag{6.88}$$

Dobrá volba je $q \doteq m$.

Metody vnitřního bodu jsou v současnosti nejefektivnější numerické metody. Spolu se sekvenčním kvadratickým programováním jsou nejoblíbenějšími numerickými metodami. Je ovšem důležité si uvědomit, že obě metody využívají při dílčích výpočtech řadu dalších numerických metod.

6.4.8 Sekvenční kvadratické programování

Uvažujme nejprve problém nelineárního programování s **omezením ve tvaru rovnosti**

$$\min\{f(\mathbf{x}) \ : \ \mathbf{h}(\mathbf{x}) = \mathbf{0}\} \tag{6.89}$$

Lagrangeova funkce pro tuto úlohu je $L(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T\mathbf{h}(\mathbf{x})$. Nutné podmínky (nulovost gradientu Lagrangeovy funkce vzhledem k oběma proměnným) vedou na soustavu nelineárních rovnic

$$\begin{array}{rcl} \nabla f(\mathbf{x}) + \boldsymbol{\lambda}^T\nabla\mathbf{h}(\mathbf{x}) & = & \mathbf{0} \\ \mathbf{h}(\mathbf{x}) & = & \mathbf{0} \end{array} \tag{6.90}$$

Pokud je funkce $f(\mathbf{x})$ kvadratická a omezení $\mathbf{h}(\mathbf{x})$ lineární, jedná se o problém kvadratického programování.

Kvadratické programování

Nejprve vyřešíme problém optimalizace kvadratického kritéria s lineárním omezením - tzv. **kvadratické programování**. Prozatím budeme uvažovat lineární **omezení typu rovnosti**

$$\min\left\{ f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{c}^T\mathbf{x} \ : \mathbf{A}\mathbf{x} = \mathbf{b} \right\} \tag{6.91}$$

Podle (6.90) jsou nutné podmínky pro optimum této úlohy lineární

$$\begin{array}{rcccl} \mathbf{H}\mathbf{x} + \mathbf{A}^T\boldsymbol{\lambda} + \mathbf{c} & = & \mathbf{0} \\ \mathbf{A}\mathbf{x} - \mathbf{b} & = & \mathbf{0} \end{array} \tag{6.92}$$

Snadno je možno ukázat, že je-li matice $\mathbf{H}$ regulární a má-li matice $\mathbf{A}$ plnou řádkovou hodnost (hod $\mathbf{A} = m$), pak je matice předchozí soustavy regulární, to znamená, že nutné podmínky (6.92) vedou na jediné řešení.

Analytické řešení získáme snadno. Z první rovnice plyne

$$\mathbf{x} = -\mathbf{H}^{-1}\mathbf{A}^T\boldsymbol{\lambda} - \mathbf{H}^{-1}\mathbf{c}$$

Předchozí rovnici vynásobíme maticí $\mathbf{A}$ a výsledek dosadíme do druhé rovnice v (6.92). Pak dostaneme

$$\boldsymbol{\lambda} = -\left(\mathbf{A}\mathbf{H}^{-1}\mathbf{A}^T\right)^{-1}\left(\mathbf{A}\mathbf{H}^{-1}\mathbf{c}+\mathbf{b}\right) \tag{6.93}$$

po dosazení do rovnice pro $\mathbf{x}$ dostaneme

$$\mathbf{x} = -\mathbf{H}^{-1}\left[\mathbf{I}-\mathbf{A}^T\left(\mathbf{A}\mathbf{H}^{-1}\mathbf{A}^T\right)^{-1}\mathbf{A}\mathbf{H}^{-1}\right]\mathbf{c}+\mathbf{H}^{-1}\mathbf{A}^T\left(\mathbf{A}\mathbf{H}^{-1}\mathbf{A}^T\right)^{-1}\mathbf{b} \tag{6.94}$$

Předchozí vztahy pro $\mathbf{x}$ a $\boldsymbol{\lambda}$ nejsou vhodné pro numerický výpočet.

Pokud máme kvadratický optimalizační problém s **omezeními ve tvaru nerovnosti**, téměř výhradně se k jeho řešení používá iterační metoda aktivních množin. Uvažujme tedy kvadratický optimalizační problém

$$\min\left\{f(\mathbf{x})=\frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x}+\mathbf{c}^T\mathbf{x}\ :\mathbf{a}_i\mathbf{x}=b_i,\ i\in\mathbf{E},\ \mathbf{a}_i\mathbf{x}\le b_i,\ i\in\mathbf{I}\right\} \tag{6.95}$$

kde $\mathbf{E}$ je množina omezení typu rovnosti a $\mathbf{I}$ je množina omezení typu nerovnosti. Z množiny omezení typu nerovnosti jsou některá aktivní a ostatní jsou neaktivní. Protože na počátku výpočtu nevíme, která z nerovnostních omezení jsou aktivní a která ne, pracujeme s tzv. pracovní množinou $\mathbf{W}$, v níž jsou všechna omezení typu rovnosti a některá omezení typu nerovnosti.

Nechť tedy v k-té iteraci jsme v bodě $\mathbf{x}_k$. Bod $\mathbf{x}_k$ vyhovuje všem omezením a splňuje omezení typu rovnosti pro současnou pracovní množinu $\mathbf{W}_k$. Množina $\mathbf{W}_k$ zahrnuje všechna omezení typu rovnosti (všechna $i\in\mathbf{E}$) a některá omezení typu nerovnosti (některá $i\in\mathbf{I}$).

Problém (6.95) můžeme zapsat v ekvivalentním tvaru

$$\min\left\{f(\mathbf{x})=f(\mathbf{x}_k)+\mathbf{g}_k^T(\mathbf{x}-\mathbf{x}_k)+\frac{1}{2}(\mathbf{x}-\mathbf{x}_k)^T\mathbf{H}(\mathbf{x}-\mathbf{x}_k):\ \begin{array}{ll}\mathbf{A}\mathbf{x}_k & =\mathbf{b},\\ \mathbf{A}\mathbf{x} & =\mathbf{b}\end{array}\right\} \tag{6.96}$$

Přitom řádky matice $\mathbf{A}$ jsou tvořeny vektory $\mathbf{a}_i^T,\ i\in\mathbf{W}_k$ a k tomu odpovídajícími prvky vektoru $\mathbf{b}$.

Aby problémy (6.95) a (6.96) byly totožné (až na konstantu, která mění pouze hodnotu kritéria, ale nemění optimální bod), pak $\mathbf{g}_k=\mathbf{c}+\mathbf{H}\mathbf{x}_k$.

Nyní si zavedeme vektor přírůstku $\mathbf{d}_k=\mathbf{x}-\mathbf{x}_k$ a pak problém (6.96) můžeme vyjádřit ve tvaru

$$\min\left\{f(\mathbf{x})=f(\mathbf{d}_k)=f(\mathbf{x}_k)+\mathbf{g}_k^T\mathbf{d}_k+\frac{1}{2}\mathbf{d}_k^T\mathbf{H}\mathbf{d}_k\ :\ \mathbf{a}_i^T\mathbf{d}_k=0,\ i\in\mathbf{W}\right\} \tag{6.97}$$

Pro tento problém vytvoříme Lagrangeovu funkci

$$L=f(\mathbf{x}_k)+\mathbf{g}_k^T\mathbf{d}_k+\frac{1}{2}\mathbf{d}_k^T\mathbf{H}\mathbf{d}_k+\sum_{i\in\mathbf{W}}\lambda_i\mathbf{a}_i^T\mathbf{d}_k \tag{6.98}$$

Odtud plynou nutné podmínky optimality

$$\begin{array}{lllll}\mathbf{H}\mathbf{d}_k & + & \mathbf{A}^T\boldsymbol{\lambda} & + & \mathbf{g}_k = 0\\ \mathbf{A}\mathbf{d}_k & & & & = 0\end{array} \tag{6.99}$$

kde řádky matice $\mathbf{A}$ jsou rovny vektorům $\mathbf{a}_i^T,\ i\in\mathbf{W}$. Z předchozí soustavy vypočteme přírůstek $\mathbf{d}_k$. Nyní může nastat několik možností:

1. Pokud $\mathbf{d}_k = 0$, pak jsme v optimu $\mathbf{x}^* = \mathbf{x}_k$, neboť to může nastat pouze pro $\mathbf{g}_k = 0$.

2. Pro $\mathbf{d}_k \neq 0$ je $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{d}_k$. Je-li $\mathbf{x}_{k+1}$ přípustné, pak $k := k + 1$ a provádíme další iteraci s nezměněnou pracovní množinou $\mathbf{W}_{k+1} = \mathbf{W}_k$.

3. Pokud $\mathbf{x}_{k+1}$ není přípustné, pak

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha \mathbf{d}_k,$$

kde koeficient $\alpha < 1$ volíme tak velký, až narazíme na nějaká nová omezení. Řekněme, že je to omezení j-té, kde $j \notin \mathbf{W}$. Pak musí platit

$$\mathbf{a}_j^T (\mathbf{x}_k + \alpha \mathbf{d}_k) = b_j$$

Protože pro všechna $j \notin \mathbf{W}$ platí $\mathbf{a}_j^T \mathbf{x}_k < b_j$, pak z předchozí rovnice plyne $\alpha \mathbf{a}_j^T \mathbf{d}_k > 0,\ j \notin \mathbf{W}$. Odtud plyne

$$\alpha = \alpha_j = \frac{b_j - \mathbf{a}_j^T \mathbf{x}_k}{\mathbf{a}_j^T \mathbf{d}_k}$$

Výpočet α_j provedeme pro všechna $j \notin \mathbf{W}$. Za skutečné $\alpha = \alpha_k$ vezmeme to nejmenší z α_j. Tímto způsobem jsme určili přípustné α_k spolu s indexem j-tého omezení, na které jsme nyní narazili a proto ho musíme zahrnout do množiny pracovních omezení, pak $\mathbf{W}_{k+1} = \mathbf{W}_k \cup j$. Proto přípustnou hodnotu α_k vypočteme podle vztahu

$$\alpha_k = \min_{\mathbf{a}_j^T \mathbf{d}_k > 0} \left[1\ ,\ \frac{b_j - \mathbf{a}_j^T \mathbf{x}_k}{\mathbf{a}_j^T \mathbf{d}_k} \right] \tag{6.100}$$

4. Tímto způsobem jsme určili bod $\mathbf{x}_{k+1}$ a novou pracovní množinu $\mathbf{W}_{k+1}$. Je-li $\lambda_i \geq 0$ pro všechna $i \in \mathbf{E}$ a $i \in \mathbf{I}$, pak bod $\mathbf{x}_{k+1} = \mathbf{x}^*$ je optimální bod.

5. Je-li některé $\lambda_i < 0$, pak z množiny všech $\lambda_i < 0$, vybereme to nejmenší. Nechť je to λ_p

$$p = \arg \min_{i,\ \lambda_i < 0} \lambda_i,$$

pak omezení p-té vypustíme z pracovní množiny $\mathbf{W}_{k+1}$.

To plyne z citlivostní věty - při minimalizaci je přírůstek kritéria $\Delta f = -\lambda_p \Delta b_p$. Protože při změně omezení z rovnosti na nerovnost je přírůstek omezení $\Delta b_p < 0$, pak pro $\Delta f < 0$ musí být příslušný Lagrangeův koeficient $\lambda_p < 0$.

To je postup řešení problému kvadratické optimalizace s lineárními omezeními ve tvaru rovnosti i nerovnosti. Nyní popíšeme numerický algoritmus pro řešení problému nelineárního programování nejprve s omezením ve tvaru rovnosti.

Přímé metody

Uvažujme tedy problém nelineárního programování (6.89). V tomto odstavci uvedeme přímé metody řešení Lagrangeových rovnic (6.90).

Zvolíme si **hodnotící funkci (merit function)**, podle které budeme oceňovat algoritmus výpočtu optima

$$m(\mathbf{x}, \boldsymbol{\lambda}) = \frac{1}{2} \left| \nabla f(\mathbf{x}) + \boldsymbol{\lambda}^T \nabla \mathbf{h}(\mathbf{x}) \right|^2 + \frac{1}{2} \left| \mathbf{h}(\mathbf{x}) \right|^2 \tag{6.101}$$

První člen v hodnotící funkci je rovný kvadrátu normy gradientu Lagrangeovy funkce podle proměnné $\mathbf{x}$ a druhý člen je vlastně penále za nesplnění omezení problému. Je zřejmé, že bod minima hodnotící funkce $m(\mathbf{x}, \boldsymbol{\lambda})$ je stejný jako bod, který splňuje Lagrangeovy rovnice (6.90) a vyhovuje omezením úlohy. Proto se hodnotící funkce také někdy nazývá absolutní penalizační funkce. Platí následující tvrzení:

Věta:
Nechť body $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ *splňují nutné podmínky prvního řádu pro lokální minimum hodnotící funkce* $m(\mathbf{x}, \boldsymbol{\lambda})$ *dle (6.101) a bod optima* $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ *je regulárním bodem (hodnost Jacobiovy matice* $\nabla \mathbf{h}(\mathbf{x}^*)$ *je rovna* m*) a Hessova matice* $\mathbf{H}(\mathbf{x}^*, \boldsymbol{\lambda}^*) = \nabla^2 f(\mathbf{x}^*) + (\boldsymbol{\lambda}^*)^T \nabla^2 \mathbf{h}(\mathbf{x}^*)$ *Lagrangeovy funkce* $L(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{h}(\mathbf{x})$ *je pozitivně definitní. Pak* $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ *je globální minimum hodnotící funkce* $m(\mathbf{x}, \boldsymbol{\lambda})$ *a* $m(\mathbf{x}^*, \boldsymbol{\lambda}^*) = 0$. □

Nyní pojednáme o dvou iteračních metodách řešení nelineárního problému. První metoda volí jako směr hledání gradient Lagrangeovy funkce, je to tedy metoda prvního řádu. Druhá metoda je Newtonova metoda druhého řádu.

Metoda prvního řádu

Nejprve k hledání optima použijeme gradientní metodu prvního řádu. Uvažujme tedy iterační proces

$$\begin{array}{rcl} \mathbf{x}_{k+1} & = & \mathbf{x}_k - \alpha_k \, \nabla_x L(\mathbf{x}_k, \boldsymbol{\lambda}_k)^T \\ \boldsymbol{\lambda}_{k+1} & = & \boldsymbol{\lambda}_k + \alpha_k \mathbf{h}(\mathbf{x}_k) \end{array} \tag{6.102}$$

Směr hledání proměnné $\mathbf{x}$ je ve směru, který je rovný zápornému gradientu Lagrangeovy funkce vzhledem k proměnné $\mathbf{x}$ (je to proto, že hledáme minimum vzhledem k $\mathbf{x}$). Směr hledání Lagrangeových koeficientů je ve směru kladného gradientu Lagrangeovy funkce vzhledem k proměnné $\boldsymbol{\lambda}$ (je to proto, že hledáme maximum vzhledem k $\boldsymbol{\lambda}$).

Nyní ukážeme, že směr hledání je směrem zajišťujícím klesání hodnotící funkce, což znamená, že je negativní skalární součin vektoru směru hledání s gradientem hodnotící funkce. Podle (6.102) je směr hledání vektor se složkami $-\nabla_x L(\mathbf{x}_k, \boldsymbol{\lambda}_k)^T$ a $\mathbf{h}(\mathbf{x}_k)$. Gradient hodnotící funkce má složky

$$\begin{array}{rcl} \nabla_x m(\mathbf{x}, \boldsymbol{\lambda}) & = & \nabla_x L(\mathbf{x}, \boldsymbol{\lambda}) \, \nabla_x^2 L(\mathbf{x}, \boldsymbol{\lambda}) + \mathbf{h}(\mathbf{x})^T \, \nabla_x \mathbf{h}(\mathbf{x}) \\ \nabla_\lambda m(\mathbf{x}, \boldsymbol{\lambda}) & = & \nabla_x L(\mathbf{x}, \boldsymbol{\lambda}) \, \nabla_x \mathbf{h}(\mathbf{x}) \end{array} \tag{6.103}$$

Skalární součin gradientu hodnotící funkce a směru hledání je roven

$$-\nabla_x L(\mathbf{x}, \boldsymbol{\lambda}) \, \nabla_x^2 L(\mathbf{x}, \boldsymbol{\lambda}) \, \nabla_x L(\mathbf{x}, \boldsymbol{\lambda})^T \leq 0.$$

Odtud ale pouze plyne, že zvolený směr hledání zaručuje klesání hodnotící funkce pokud $\nabla_x L(\mathbf{x}, \boldsymbol{\lambda}) \neq \mathbf{0}$. To znamená, že volbou koeficientu α_k pomocí nějakého algoritmu jednorozměrové optimalizace hodnotící funkce ve zvoleném směru, iterační proces konverguje

k bodu, ve kterém $\nabla_x L(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{0}$. Nemáme ale žádnou záruku, že v nalezeném bodě je $\mathbf{h}(\mathbf{x}) = 0$.

Proto je třeba provést modifikaci hodnotící funkce. Upravená hodnotící funkce je rovna

$$\bar{m}(\mathbf{x}, \boldsymbol{\lambda}) = m(\mathbf{x}, \boldsymbol{\lambda}) - \gamma \left[f(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{h}(\mathbf{x}) \right] \tag{6.104}$$

Gradient modifikované hodnotící funkce je

$$\begin{aligned} \nabla_x \bar{m}(\mathbf{x}, \boldsymbol{\lambda}) &= \nabla_x L(\mathbf{x}, \boldsymbol{\lambda})\ \nabla_x^2 L(\mathbf{x}, \boldsymbol{\lambda}) + \mathbf{h}(\mathbf{x})^T\ \nabla_x \mathbf{h}(\mathbf{x}) - \gamma \nabla_x L(\mathbf{x}, \boldsymbol{\lambda}) \\ \nabla_\lambda \bar{m}(\mathbf{x}, \boldsymbol{\lambda}) &= \nabla_x L(\mathbf{x}, \boldsymbol{\lambda})\ \nabla_x \mathbf{h}(\mathbf{x}) - \gamma \mathbf{h}(\mathbf{x})^T \end{aligned} \tag{6.105}$$

Součin gradientu modifikované hodnotící funkce se směrem hledání je roven

$$-\nabla_x L(\mathbf{x}, \boldsymbol{\lambda})\ \left[\nabla_x^2 L(\mathbf{x}, \boldsymbol{\lambda}) - \gamma \mathbf{I} \right]\ \nabla_x L(\mathbf{x}, \boldsymbol{\lambda})^T - \gamma \left| \mathbf{h}(\mathbf{x}) \right|^2$$

Protože předpokládáme, že Hessova matice Lagrangeovy funkce je pozitivně definitní, pak existuje γ (dostatečně malé), že předchozí výraz je negativní pokud současně $\nabla_x L(\mathbf{x}, \boldsymbol{\lambda}) \neq 0$ a $\mathbf{h}(\mathbf{x}) \neq 0$. Proto zvolený směr hledání zaručuje klesání modifikované hodnotící funkce. Pro určitou hodnotu koeficientu γ iterační metoda (6.102) konverguje k řešení problému (6.89).

Newtonova metoda

Pro nalezení stacionárního bodu Lagrangeovy funkce je nejpoužívanější Newtonova metoda. Budeme tedy aplikovat Newtonovu metodu na řešení soustavy

$$\begin{aligned} \nabla_x L(\mathbf{x}, \boldsymbol{\lambda}) &= \nabla f(\mathbf{x}) + \boldsymbol{\lambda}^T \nabla \mathbf{h}(\mathbf{x}) &= 0 \\ \nabla_\lambda L(\mathbf{x}, \boldsymbol{\lambda}) &= \mathbf{h}(\mathbf{x}) &= 0 \end{aligned} \tag{6.106}$$

Připomeňme si, že pro řešení rovnice $q(\mathbf{x}) = 0$ je Newtonův iterační algoritmus $\mathbf{x}_{k+1} = \mathbf{x}_k - [q'(\mathbf{x}_k]^{-1} q(\mathbf{x}_k)$.

Naši funkci z (6.106) rozvineme v řadu v okolí $\mathbf{x}_k$

$$\begin{aligned} \begin{bmatrix} \nabla_x L(\mathbf{x}_{k+1}, \boldsymbol{\lambda}_{k+1})^T \\ \mathbf{h}(\mathbf{x}_{k+1}) \end{bmatrix} &= \begin{bmatrix} \nabla_x L(\mathbf{x}_k, \boldsymbol{\lambda}_k)^T \\ \mathbf{h}(\mathbf{x}_k) \end{bmatrix} + \begin{bmatrix} \nabla_x^2 L(\mathbf{x}_k, \boldsymbol{\lambda}_k) \\ \nabla \mathbf{h}(\mathbf{x}_k) \end{bmatrix} \Delta \mathbf{x}_k \\ &\quad + \begin{bmatrix} \nabla_x \mathbf{h}(\mathbf{x}_k)^T \\ 0 \end{bmatrix} \Delta \boldsymbol{\lambda}_k \end{aligned} \tag{6.107}$$

kde $\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta \mathbf{x}_k$ a $\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \Delta \boldsymbol{\lambda}_k$. Aby

$$\begin{bmatrix} \nabla_x L(\mathbf{x}_{k+1}, \boldsymbol{\lambda}_{k+1})^T \\ \mathbf{h}(\mathbf{x}_{k+1}) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

pak platí

$$\begin{bmatrix} \nabla_x^2 L(\mathbf{x}_k, \boldsymbol{\lambda}_k) & \nabla \mathbf{h}(\mathbf{x}_k)^T \\ \nabla \mathbf{h}(\mathbf{x}_k) & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x}_k \\ \Delta \boldsymbol{\lambda}_k \end{bmatrix} = \begin{bmatrix} -\nabla_x L(\mathbf{x}_k, \boldsymbol{\lambda}_k)^T \\ -\mathbf{h}(\mathbf{x}_k) \end{bmatrix} \tag{6.108}$$

K první rovnici v předchozí soustavě přidáme člen $(\nabla \mathbf{h}(\mathbf{x}_k)^T \boldsymbol{\lambda}_k)$, pak je první rovnice

$$\nabla_x^2 L(\mathbf{x}_k, \boldsymbol{\lambda}_k) \Delta \mathbf{x}_k + \nabla \mathbf{h}(\mathbf{x}_k)^T (\Delta \boldsymbol{\lambda}_k + \boldsymbol{\lambda}_k) = -\nabla_x L(\mathbf{x}_k, \boldsymbol{\lambda}_k)^T + \nabla \mathbf{h}(\mathbf{x}_k)^T \boldsymbol{\lambda}_k$$

Výraz na pravé straně předchozí rovnice je roven $-\nabla f(\mathbf{x}_k)^T$. Potom lze rovnici (6.108) zapsat ve tvaru

$$\begin{bmatrix} \nabla_x^2 L(\mathbf{x}_k, \boldsymbol{\lambda}_k) & \nabla \mathbf{h}(\mathbf{x}_k)^T \\ \nabla \mathbf{h}(\mathbf{x}_k) & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x}_k \\ \boldsymbol{\lambda}_{k+1} \end{bmatrix} = \begin{bmatrix} -\nabla f(\mathbf{x}_k)^T \\ -\mathbf{h}(\mathbf{x}_k) \end{bmatrix} \tag{6.109}$$

Předchozí soustava rovnic je podobná soustavě (6.92). Bude mít tedy jednoznačné řešení, pokud matice $\nabla \mathbf{h}(\mathbf{x}^*)$ má plnou řádkovou hodnost a matice $\nabla_x^2 L(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ je pozitivně definitní. Postačující podmínky druhého řádu pro náš původní problém (6.89) jsou kromě toho, že matice $\nabla \mathbf{h}(\mathbf{x}^*)$ má plnou řádkovou hodnost, ještě navíc matice $\nabla_x^2 L(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ musí být pozitivně definitní na tečném podprostoru M k množině omezení, kde $M = \{\mathbf{x} : \nabla \mathbf{h}(\mathbf{x}^*)\mathbf{x} = \mathbf{0}\}$.

Postačující podmínky jsou tedy trochu odlišné od předchozích podmínek pro jednoznačné řešení předchozí soustavy rovnic. Abychom zajistili pozitivní definitnost Hessovy matice Lagrangeovy funkce na celém prostoru, pak je třeba opět provést modifikaci původního problému (6.89). Místo problému $\min\{f(\mathbf{x}) : \mathbf{h}(\mathbf{x}) = \mathbf{0}\}$ uvažujeme ekvivalentní problém $\min\left\{f(\mathbf{x}) + \frac{1}{2}c\left|\mathbf{h}(\mathbf{x})\right|^2 : \mathbf{h}(\mathbf{x}) = 0\right\}$ přidáním penále $\frac{1}{2}c\left|\mathbf{h}(\mathbf{x})\right|^2$. Hessova matice tohoto problému je

$$\mathbf{H}(\mathbf{x}) = \nabla_x^2 L(\mathbf{x}, \boldsymbol{\lambda}) + c\nabla \mathbf{h}(\mathbf{x})^T \nabla \mathbf{h}(\mathbf{x})$$

kde $L(\mathbf{x}, \boldsymbol{\lambda})$ je Lagrangeova funkce původního problému. Pro dostatečně velkou penalizační konstantu c je tato nová Hessova matice pozitivně definitní na celém prostoru.

Newtonova metoda zaručuje řád konvergence alespoň dvě, pokud jsme dostatečně blízko řešení. Abychom zaručili konvergenci ze vzdálených bodů - globální konvergenci - je třeba zaručit, aby proces hledání zajišťoval klesání hodnotící funkce. To zaručíme, když v Newtonově směru budeme provádět jednorozměrovou minimalizaci volbou optimálního koeficientu α. Ono totiž platí, že směr generovaný Newtonovou metodou je směr, který zaručuje klesání hodnotící funkce (6.101). Součin gradientu (6.103) hodnotící funkce s Newtonovým směrem podle (6.108) je roven

$$-\left|\nabla_x L(\mathbf{x}_k, \boldsymbol{\lambda}_k)\right|^2 - \left|\mathbf{h}(\mathbf{x}_k)\right|^2$$

Předchozí výraz je záporný pokud $\nabla_x L(\mathbf{x}_k, \boldsymbol{\lambda}_k) \neq \mathbf{0}$ nebo $\mathbf{h}(\mathbf{x}_k) \neq \mathbf{0}$.

Odtud plyne, že Newtonova metoda zaručuje globální konvergenci, pokud je aplikována s proměnným krokem.

Modifikované Newtonovy metody a kvazi-Newtonovy metody

Modifikace Newtonovy metody se provádí proto, abychom se vyhnuli přímému výpočtu Hessovy matice Lagrangeovy funkce $\mathbf{H}_k = \nabla_x^2 L(\mathbf{x}_k, \boldsymbol{\lambda}_k)$. To je možno provést různým způsobem. Hessovu matici můžeme aproximovat nějakou maticí $\mathbf{Q}_k$, která může být buď konstantní během iteračního procesu, nebo může být aktualizovaná například metodou BFGS, pak

$$\mathbf{Q}_{k+1} = \mathbf{Q}_k + \frac{\mathbf{q}_k \mathbf{q}_k^T}{\mathbf{q}_k^T \mathbf{d}_k} - \frac{\mathbf{Q}_k^T \mathbf{Q}_k}{\mathbf{d}_k^T \mathbf{Q}_k \mathbf{d}_k} \tag{6.110}$$

kde

$$\begin{aligned} \mathbf{d}_k &= \mathbf{x}_{k+1} - \mathbf{x}_k \\ \mathbf{q}_k^T &= \nabla_x L(\mathbf{x}_{k+1}, \boldsymbol{\lambda}_{k+1}) - \nabla_x L(\mathbf{x}_k, \boldsymbol{\lambda}_k) \end{aligned}$$

Při tom $\boldsymbol{\lambda}_{k+1}$ získáme řešením (6.109). Abychom soustavu rovnic (6.109) nemuseli řešit celou, provádí se aproximace vektoru $\boldsymbol{\lambda}_{k+1}$ a řešíme pouze horní část soustavy (6.109), to je rovnici

$$\nabla_x^2 L(\mathbf{x}_k, \boldsymbol{\lambda}_k)\ \Delta\mathbf{x}_k + \nabla\mathbf{h}(\mathbf{x}_k)^T\ \boldsymbol{\lambda}_{k+1} = -\nabla f(\mathbf{x}_k)^T$$

jejíž řešení je po úpravě

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \left[\nabla_x^2 L(\mathbf{x}_k, \boldsymbol{\lambda}_k)\right]^{-1} \nabla_x L(\mathbf{x}_k, \widehat{\boldsymbol{\lambda}}_k)^T \tag{6.111}$$

kde $\widehat{\boldsymbol{\lambda}}_k$ je nějaký odhad aktualizace vektoru Lagrangeových koeficientů. Existuje celá řada možností, z nichž zde uvedeme pouze jedinou

$$\widehat{\boldsymbol{\lambda}}_k = \boldsymbol{\lambda}_k + c\mathbf{h}(\mathbf{x}_k).$$

Poslední člen vznikl z penalizačního členu $\frac{1}{2}c\,|\mathbf{h}(\mathbf{x})|^2$, který se přidá k minimalizované funkci $f(\mathbf{x})$ a $c > 0$ je nějaká konstanta.

Rozšíření na nerovnostní omezení

Newtonova metoda řeší soustavu rovnic, která je podobná soustavě rovnic při řešení problému kvadratického programování při lineárním omezení ve tvaru rovnosti. Kvadratické programování jsme posléze rozšířili na řešení problémů se smíšenými omezeními ve tvaru rovnosti i nerovnosti. Na řešení tohoto problému jsme použili metodu aktivních množin.

Proto je možno navrhnout obdobným způsobem rozšíření Newtonovy metody na řešení problémů s nerovnostními omezeními. Uvažujme tedy následující optimalizační problém

$$\min\{f(\mathbf{x})\ :\ \mathbf{h}(\mathbf{x}) = \mathbf{0},\ \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\} \tag{6.112}$$

Tento problém řešíme iteračně. Definujeme si Lagrangeovu funkci

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T\mathbf{h}(\mathbf{x}) + \boldsymbol{\mu}^T\mathbf{g}(\mathbf{x}) \tag{6.113}$$

Mějme tedy přípustný bod $\mathbf{x}_k$ a vektory Lagrangeových koeficientů $\boldsymbol{\lambda}_k$ a $\boldsymbol{\mu}_k$. Iteračně řešíme kvadratický problém

$$\min\left\{\nabla f(\mathbf{x}_k)\mathbf{s}_k + \frac{1}{2}\mathbf{s}_k^T\left[\nabla^2 f(\mathbf{x}_k) + \boldsymbol{\lambda}_k^T\ \nabla^2\mathbf{h}(\mathbf{x}_k) + \boldsymbol{\mu}_k^T\ \nabla^2\mathbf{g}(\mathbf{x}_k)\right]\mathbf{s}_k\right\} \tag{6.114}$$

s omezením

$$\begin{aligned} \nabla\mathbf{h}(\mathbf{x}_k)\mathbf{s}_k + \mathbf{h}(\mathbf{x}_k) &= \mathbf{0} \\ \nabla\mathbf{g}(\mathbf{x}_k)\mathbf{s}_k + \mathbf{g}(\mathbf{x}_k) &\leq \mathbf{0} \end{aligned} \tag{6.115}$$

kde $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$ a Lagrangeovy multiplikátory jsou Lagrangeovými multiplikátory problému kvadratického programování (6.114).

Sekvenční kvadratické programování se skládá ze tří základních kroků:

1. Aktualizace Hessovy matice Lagrangeovy funkce
2. Použití kvadratického programování na řešení kvadratického problému s použitím metody aktivních množin
3. Jednorozměrové hledání a výpočet hodnotící funkce

Metoda vyžaduje start algoritmu z přípustného bodu. Pokud ho neznáme (nelze zvolit z reálné podstaty problému), pak přípustný bod získáme řešením následujícího problému

$$\min \{\gamma \ : \ \mathbf{h}(\mathbf{x}) = \mathbf{0}, \ \mathbf{g}(\mathbf{x}) - \gamma \mathbf{e} \leq \mathbf{0}\} \tag{6.116}$$

kde $\mathbf{e} = [1, 1, \ldots, 1]^T$ je vektor s jednotkovými prvky. Za počáteční přípustný bod $\mathbf{x}_0$ zvolíme nějaký bod, splňující pouze omezení typu rovnosti.

Algoritmus sekvenčního kvadratického programování je syntézou řady metod. Na závěr uvedeme v souhrnu algoritmus metody.

1. Nalezení přípustného počátečního bodu $\mathbf{x}_0$ a počáteční pozitivně definitní matice $\mathbf{Q}_0$.
2. Řešení kvadratického problému (6.114) s omezními (6.115). Je-li $\mathbf{s}_k = \mathbf{0}$, pak jsme získali řešení problému v bodě $\mathbf{x}_k$.
3. Ve směru $\mathbf{s}_k$ provedeme jednorozměrovou minimalizaci, pak

   $$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{s}_k$$

   Parametr α_k je volen tak, aby zajistil dostatečný pokles hodnotící funkce. Hodnotící funkci volíme ve tvaru

   $$m(\mathbf{x}) = f(\mathbf{x}) + \sum r_i h_i(\mathbf{x}) + \sum r_j \max [0, \ g_i(\mathbf{x})]$$

   Váhové koeficienty r_i, r_j se volí úměrné Lagrangeovým koeficientům λ_i, μ_j.
4. Provedeme aktualizaci matice $\mathbf{Q}_k$ podle metody BFGS. V literatuře jsou popsány i jiné možnosti aktualizace aproximace Hessovy matice.

Metoda sekvenčního kvadratického programování je spolehlivá a velmi efektivní numerická metoda.

Kapitola 7

Řešení regulačních problémů variačními metodami

7.1 Problém optimálního řízení dynamických systémů

Dosud jsme se věnovali řešení problémů optimalizace statických systémů, které byly popsány soustavou rovnic a nerovnic. Čas se v těchto problémech nevyskytoval.

Nyní se budeme věnovat problémům optimalizace dynamických systémů, to je systémů v nichž jsou proměnné závislé na čase. Nejprve se budeme věnovat optimalizaci spojitých systémů. Pro řešení takových problémů je třeba znát

1. Popis dynamického systému, modelujícího reálný objekt, který chceme optimálně řídit. Toto omezení je vyjádřeno obvykle diferenciálními rovnicemi.

2. Z podstaty problému vyplývají omezení některých proměnných. Proto součástí formulace problému jsou často soustavy rovnic a nerovnic.

3. Nezbytnou součástí problému je výběr cíle, který chceme dosáhnout. Tento cíl se obvykle formuluje ve tvaru kritéria optimality a naším cílem je optimalizovat (minimalizovat či maximalizovat) toto kritérium.

Nyní tyto obecné úvahy budeme konkretizovat.
Dynamický systém je obvykle popsán stavovými rovnicemi

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \\ \mathbf{y}(t) &= \bar{\mathbf{f}}(\mathbf{x}, \mathbf{u}, t) \end{aligned} \tag{7.1}$$

kde $\mathbf{x}(t)$ je stavový vektor systému,
$\mathbf{u}(t)$ je řídicí vektor - nezávisle proměnná,
$\mathbf{y}(t)$ je výstupní vektor.
Veličiny v systému nemohou obvykle nabývat libovolných hodnot, jejich omezení je

$$\mathbf{u}(t) \in \mathbf{U} \subset R^r, \qquad \mathbf{x}(t) \in \mathbf{X} \subset R^n. \tag{7.2}$$

kde **U**,**X** jsou množiny přípustných hodnot řízení a stavu systému, které jsou obvykle kompaktní, nebo jestě lépe konvexní množiny.

Obvykle známe počáteční stav systému $\mathbf{x}(t_0) = \mathbf{x}_0$. Naším problémem je řídit systém tak, aby na konci intervalu řízení $[t_0,\ t_1]$ byl systém ve stavu $\mathbf{x}(t_1) = \mathbf{x}_1$.

Pokud je koncový stav dosažitelný, pak přechod z počátečního stavu $\mathbf{x}_0$ do koncového stavu $\mathbf{x}_1$ může být uskutečněn různým způsobem. Abychom mohli vybrat optimální řešení, je třeba zvolit kritérium kvality řízení, které nějakým způsobem ohodnotí řešení úlohy - každému řešení přiřadí reálné číslo (často pouze nezáporné) a podle jeho velikosti můžeme porovnávat jednotlivá řešení a vybírat z nich to nejlepší. Kritérium kvality řízení (které budeme značit J) v problémech dynamické optimalizace je obvykle ve tvaru

$$J\left(t_0, \mathbf{x}(t_0), t_1, \mathbf{x}(t_1), \mathbf{u}(t)\right) = h(\mathbf{x}(t_1)) + \int_{t_0}^{t_1} g(\mathbf{x}(t), \mathbf{u}(t), t)\ dt \tag{7.3}$$

kde h a g jsou skalární funkce svých argumentů.
Kritérium kvality řízení má dva členy. První člen $h(\mathbf{x}(t_1)$, je závislý na stavu systému v koncovém čase t_1. Tento člen hodnotí tedy cíl trajektorie. Druhý, integrální člen, hodnotí průběh trajektorie systému - způsob jakým je dosaženo cíle.

Definice problému optimálního řízení:
Problém optimálního řízení spočívá v určení takového řízení $\mathbf{u}(t)$ *systému (7.1) na intervalu* $t_0 \leq t \leq t_1$*, aby byla splněna omezení (7.2) a kritérium kvality řízení (7.3) bylo minimální. Takové řízení je optimální řízení a budeme ho značit* $\mathbf{u}^*(t)$*.* □

Tento obecný problém má různé modifikace:
Koncový bod trajektorie, to je koncový čas t_1 a koncový stav $\mathbf{x}(t_1)$ může být pevně zadán. Pak mluvíme o **problému s pevným koncem trajektorie**. V tomto případě není třeba uvažovat první člen v kritériu (7.3), protože je konstantní a nemá vliv na optimalizaci. Není-li určen pouze koncový čas t_1, pak hovoříme o **problému s volným koncovým časem**. Není-li určen pouze koncový stav $\mathbf{x}(t_1)$, pak hovoříme o **problému s volným koncovým stavem**. Případně koncový bod musí být prvkem nějaké dané množiny, zvané **cílová množina**.

Nejprve budeme řešit takové problémy, ve kterých nejsou omezeny množiny přípustných stavů **X** a přípustných řízení **U**.

Povšimněme si, že kritérium J podle (7.3) přiřadí každé funkci $\mathbf{u}(t)$ a $\mathbf{x}(t)$ splňující omezení (7.1) a (7.2) nějaké reálné číslo (hodnotu kritéria). Kritérium (7.3) je tedy **funkcionál**. Problém optimálního řízení spojitého systému je tedy problémem nalezení extrému funkcionálu při respektování omezení danými stavovou rovnicí systému (7.1) a omezujícími podmínkami (7.2).

Určením optimálního řízení $\mathbf{u}^*(t)$ vyřešíme problém **optimálního ovládání**, to je řízení systému v otevřené smyčce. Pro určení optimálního zpětnovazebního řízení je třeba určit optimální řízení jako funkci stavu, pak $\mathbf{u}^* = \mathbf{u}^*(\mathbf{x}, t)$. Tím provedeme syntézu **optimálního regulátoru**.

Nutnou podmínkou existence optimálního řízení je to, že existuje alespoň jedno přípustné řízení, kterým stav $\mathbf{x}(t_1)$ v čase t_1 dosáhneme. Není-li řízení omezeno, předchozí požadavek splníme, leží-li koncový stav $\mathbf{x}(t_1)$ v podprostoru dosažitelných stavů. Pro omezené řízení dle (7.2) ani dosažitelnost koncového stavu obecně nestačí pro řešitelnost

problému.

Budeme-li v kritériu (7.3) hodnotit místo stavu $\mathbf{x}(t)$ pouze výstup systému $\mathbf{y}(t)$, pak takové kritérium snadno do tvaru (7.3) převedeme, dosazením z (7.1b) za $\mathbf{y}(t)$.

Často máme jiné požadavky na řízení systému. Máme dánu funkci $\mathbf{w}(t)$ - nazývanou referenční signál, reference nebo požadovaná funkce - a naším požadavkem je, aby výstup systému co nejvěrněji sledoval tuto referenci. Chceme tedy, aby odchylka trajektorie $\mathbf{e}(t) = \mathbf{w}(t) - \mathbf{y}(t)$ byla v nějakém ohledu minimální. Přitom často nemáme žádné požadavky na koncový bod trajektorie - bod $\mathbf{x}(t_1)$ je volný. Tento problém se nazývá **problém sledování** (Tracking Problem). Také pro tento problém můžeme zavést kritérium jehož obecný tvar je totožný s tvarem (7.3).

Je-li reference $\mathbf{w}(t)$ konstantní (nejčastěji nulová), koncový bod $\mathbf{x}(t_1)$ je volný a koncový čas t_1 je pevný, pak se jedná o specielní problém sledování, který se nazývá **problém regulátoru** (Regulator Problem).

7.2 Variační metody

V této kapitole velmi stručně pojednáme o variačních metodách, které určují nutné a postačující podmínky, které musí splňovat extrém funkcionálu - kritéria kvality řízení.

7.2.1 Základní variační úloha

Základní variační úloha spočívá v nalezení extrému funkcionálu ve tvaru

$$J\left(x(t)\right) = \int_{t_0}^{t_1} g\left(x(t), \dot{x}(t), t\right)\ dt \tag{7.4}$$

Funkce $g(.)$ je tzv. **jádro funkcionálu**. Pro danou reálnou funkci $x(t)$ má funkcionál určitou hodnotu. Přitom funkce $x(t)$ musí splňovat určité požadavky - obvykle to musí být funkce spojitá s derivacemi definovanými jednoznačně s výjimkou konečného počtu bodů. Takové funkce nazýváme přípustné funkce a body, ve kterých nejsou derivace funkce jednoznačně definovány se nazývají úhlové body. Křivka $x^*(t)$, která zajišťuje extrém funkcionálu, se nazývá **extremála**.

Nyní odvodíme nutné podmínky, které splňuje extremála. Nutné podmínky budou obdobné nutným podmínkám pro extrém funkce - nulovost první derivace v bodě extrému. Postup odvození nutných podmínek extrému funkcionálu (7.4) je následující. Funkci $x(t)$ vnoříme do třídy funkcí

$$x(t) = x^*(t) + \varepsilon\delta x(t)$$

kde $x^*(t)$ je extremála, $\delta x(t)$ je variace funkce $x(t)$ (odchylka funkce od extremály) a ε je nějaké reálné číslo. Spočteme hodnotu funkcionálu $J(x^*(t))$ a $J(x^*(t) + \varepsilon\delta x(t))$ a určíme přírůstek hodnoty funkcionálu

$$\Delta J(\varepsilon, x^*(t), \delta x(t)) = J(x^*(t) + \varepsilon\delta x(t)) - J(x^*(t)) \tag{7.5}$$

Je-li extremála $x^*(t)$ taková funkce, při níž nabývá funkcionál (7.4) minimální hodnoty, je zřejmé, že přírůstek funkcionálu ΔJ podle (7.5) je nezáporný pro libovolné ε a $\delta x(t)$

Za předpokladu, že přírůstek funkcionálu má konečné derivace podle ε v okolí $\varepsilon = 0$, můžeme přírůstek funkcionálu rozvést do řady, pak

$$\Delta J = \left.\frac{\partial J(x^* + \varepsilon\delta x)}{\partial \varepsilon}\right|_{\varepsilon=0} \varepsilon + \frac{1}{2} \left.\frac{\partial^2 J(x^* + \varepsilon\delta x)}{\partial \varepsilon^2}\right|_{\varepsilon=0} \varepsilon^2 + \cdots \tag{7.6}$$

První člen rozvoje se nazývá první variace funkcionálu, kterou značíme δJ. První variace funkcionálu je obdobná první derivaci funkce. Obdobně je druhý člen rozvoje nazýván druhou variací funkcionálu, pak přírůstek funkcionálu je

$$\Delta J = \varepsilon\ \delta J + \frac{1}{2}\varepsilon^2\ \delta^2 J + \cdots$$

Aby v $x^*(t)$ nastal extrém funkcionálu, je nutná podmínka nulovost první variace funkcionálu pro $x(t) = x^*(t)$, čili

$$\left.\frac{\partial J(x^* + \varepsilon\delta x)}{\partial \varepsilon}\right|_{\varepsilon=0} = 0 \quad \text{pro libovolné} \quad \delta x \tag{7.7}$$

Znaménko druhé variace funkcionálu

$$\left.\frac{\partial^2 J(x^* + \varepsilon\delta x)}{\partial \varepsilon^2}\right|_{\varepsilon=0} \tag{7.8}$$

určuje druh extrému, to je zda se jedná o minimum či maximum funkcionálu, pro $x(t) = x^*(t)$.

Nyní odvodíme nutné podmínky extrému funkcionálu ve tvaru (7.4). Pak platí

$$J(x^* + \varepsilon\delta x) = \int_{t_0}^{t_1} g\left(x^*(t) + \varepsilon\delta x(t), \dot{x}^*(t) + \varepsilon\delta\dot{x}(t), t\right)\ dt \tag{7.9}$$

kde $\delta\dot{x}(t) = \frac{d}{dt}\delta x(t)$. Nutná podmínka extrému je nulovost první variace funkcionálu, čili

$$\left.\frac{\partial J(x^* + \varepsilon\delta x)}{\partial \varepsilon}\right|_{\varepsilon=0} = \frac{\partial}{\partial \varepsilon}\int_{t_0}^{t_1} g\left(x^*(t) + \varepsilon\delta x(t), \dot{x}^*(t) + \varepsilon\delta\dot{x}(t), t\right)\ dt = 0 \tag{7.10}$$

Jádro funkcionálu rozvineme do řady, pak

$$\begin{aligned} &\left.\frac{\partial J(x^* + \varepsilon\delta x)}{\partial \varepsilon}\right|_{\varepsilon=0} = \\ &= \frac{\partial}{\partial \varepsilon}\int_{t_0}^{t_1} \left[g\left(x^*(t), \dot{x}^*(t), t\right) + \varepsilon\delta x(t)\frac{\partial g}{\partial x} + \varepsilon\delta\dot{x}(t)\frac{\partial g}{\partial \dot{x}} + \delta t\frac{\partial g}{\partial t} + 0(\varepsilon)\right]\ dt \end{aligned} \tag{7.11}$$

kde derivace funkce g podle svých proměnných bereme v bodě $x(t) = x^*(t)$ (to je pro $\varepsilon = 0$, $\delta x = 0$). Po provedení derivace dostaneme

$$\begin{aligned} \delta J = \left.\frac{\partial J(x^* + \varepsilon\delta x)}{\partial \varepsilon}\right|_{\varepsilon=0} &= \int_{t_0}^{t_1} \left[\frac{\partial g}{\partial x}\delta x(t) + \frac{\partial g}{\partial \dot{x}}\delta\dot{x}(t)\right]\ dt \\ &= \int_{t_0}^{t_1} \left[g_x\delta x(t) + g_{\dot{x}}\delta\dot{x}(t)\right]\ dt \end{aligned} \tag{7.12}$$

kde pro jednoduchost jsme zavedli značení $g_x = \frac{\partial g}{\partial x}$, $g_{\dot{x}} = \frac{\partial g}{\partial \dot{x}}$. Nyní budeme integrovat po částech předchozí výraz.

Poznámka: Integraci per partes provádíme podle vztahu

$$\int_a^b u\ dv = [u\ v]_a^b - \int_a^b v\ du \tag{7.13}$$

kde $u = u(t)$ a $v = v(t)$. Analogicky platí

$$\int_a^b u\dot{v}\ dt = [u\ v]_a^b - \int_a^b \dot{u}v\ dt \tag{7.14}$$

□

Provedeme integraci per partes druhého členu v (7.12), pak

$$\int_{t_0}^{t_1} \frac{\partial g}{\partial \dot{x}} \delta\dot{x}(t)\ dt = [g_{\dot{x}}\delta x(t)]_{t_0}^{t_1} - \int_{t_0}^{t_1} \frac{d}{dt} g_{\dot{x}}\delta x(t)\ dt$$

Odtud variace funkcionálu je

$$\delta J = [g_{\dot{x}}\delta x(t)]_{t_0}^{t_1} + \int_{t_0}^{t_1} \left(g_x - \frac{d}{dt}g_{\dot{x}}\right) \delta x(t)\ dt = \int_{t_0}^{t_1} \left(g_x - \frac{d}{dt}g_{\dot{x}}\right) \delta x(t)\ dt \tag{7.15}$$

Poslední úprava plyne z toho, že uvažujeme nejprve pevné krajní body a proto $\delta x(t_0) = \delta x(t_1) = 0$. Nutná podmínka pro extremálu je nulovost první variace funkcionálu ($\delta J = 0$).

Dále použijeme pomocnou větu:

Věta: *Je-li nějaká funkce $G(t)$ spojitá na intervalu $(t_0,\ t_1)$ a je-li*

$$\int_{t_0}^{t_1} G(t)\delta x\ dt\ =\ 0,$$

pak pro libovolnou funkci $\delta x(t)$, splňující okrajové podmínky $\delta x(t_0) = \delta x(t_1) = 0$, musí být funkce $G(t) = 0$ na intervalu $t_0 \le t \le t_1$.

□

Odtud plyne nutná podmínka, aby funkce $x(t)$ byla extremálou. Podle (7.15) musí extremála vyhovovat **Eulerově - Lagrangeově** rovnici

$$g_x - \frac{d}{dt}g_{\dot{x}} = 0 \tag{7.16}$$

Eulerova - Lagrangeova rovnice je obyčejná diferenciální rovnice druhého řádu. Po provedení derivace podle času je tvaru

$$\frac{\partial g}{\partial x} - \frac{\partial^2 g}{\partial t \partial \dot{x}} - \frac{\partial^2 g}{\partial x \partial \dot{x}}\frac{dx}{dt} - \frac{\partial^2 g}{\partial \dot{x}^2}\frac{d^2x}{dt^2} = 0 \tag{7.17}$$

neboli při úsporném zápisu

$$g_x - g_{t\dot{x}} - g_{x\dot{x}}\frac{dx}{dt} - g_{\dot{x}\dot{x}}\frac{d^2x}{dt^2} = 0 \tag{7.18}$$

Jádro funkcionálu - funkce $g(x, \dot{x}, t)$ - musí být funkce spojitá spolu se svými parciálními derivacemi do druhého řádu pro $t_0 \leq t \leq t_1$. Extremála $x^*(t)$ je funkce spojitá, má spojité první a druhé derivace až na konečný počet bodů - tyto body se nazývají **úhlové body**.

Eulerova rovnice (7.16), (7.17) nebo (7.18) je nelineární diferenciální rovnice druhého řádu. Jejím řešením dostaneme dvouparametrickou soustavu funkcí $x(t, \alpha, \beta)$ v níž parametry α a β určíme z okrajových podmínek, neboť podle zadání problému, extremála $x(t, \alpha, \beta)$ musí procházet body $x(t_0) = x_0$, $x(t_1) = x_1$.

Příklad 1: Zobecněná kvadratická regulační plocha
Nalezneme extremály, které minimalizují funkcionál ve tvaru zobecněné kvadratické regulační plochy

$$J = \int_0^\infty \left(x^2(t) + T^2 \dot{x}^2(t) \right) \, dt \tag{7.19}$$

Okrajové podmínky jsou $x(0) = x_0$, $x(\infty) = 0$.

Z Eulerových - Lagrangeových rovnic plyne, že extremály získáme řešením diferenciální rovnice

$$g_x - \frac{d}{dt} g_{\dot{x}} = 2x - 2T^2 \ddot{x} = 0$$

Řešení předchozí rovnice je zřejmě rovno

$$x(t) = \alpha e^{-\frac{t}{T}} + \beta e^{+\frac{t}{T}}$$

Z okrajových podmínkek dostaneme řešení - extremála je funkce

$$x^*(t) = x_0 e^{-\frac{t}{T}} \tag{7.20}$$

Extremála je exponenciela s časovou konstantou T. Pro $T = 0$ neexistuje spojitá funkce splňující okrajové podmínky.

Při zpětnovazebním optimálním řízení systému někdy navrhujeme konstanty regulátoru s pevnou strukturou tak, aby minimalizovaly kritérium (7.19). Hledáme tedy takové konstanty regulátoru, aby se regulační odchylka při odezvě na skok řídicí veličiny či poruchy co nejvíce blížila extremále (7.20). Volbou časové konstanty v kritériu (7.19) volíme vlastně vhodnou dobu přechodového děje a tím také můžeme zmenšit či odstranit překývnutí reálného regulačního obvodu.

□

První problém s omezením je tzv. **izoperimetrická úloha**. Izoperimetrická úloha je úlohou na minimalizaci funkcionálu (7.4) za omezující podmínky

$$\int_{t_0}^{t_1} f(x, \dot{x}, t) \, dt \; = \; K. \tag{7.21}$$

Omezující podmínka v izoperimetrické úloze je integrálním omezením. Tuto úlohu řešíme tak, že vytvoříme rozšířený funkcionál

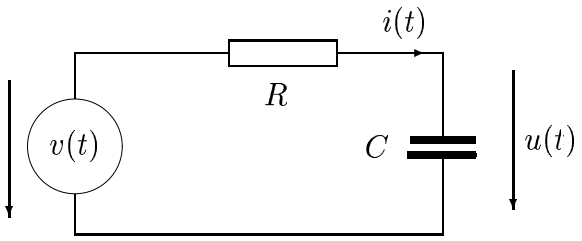
$$\bar{J} = \int_{t_0}^{t_1} g(x, \dot{x}, t) + \lambda f(x, \dot{x}, t) \, dt \; = \; \int_{t_0}^{t_1} \bar{g}(x, \dot{x}, t) \, dt \tag{7.22}$$

kde $\lambda = \text{konst.}$ je Lagrangeův koeficient. Je to obdoba řešení úlohy na vázaný extrém.

Pro tento funkcionál vyřešíme Eulerovy - Lagrangeovy rovnice a získáme extremály $x^*(t, \alpha, \beta, \lambda)$. Integrační konstanty opět určíme z okrajových podmínek a Lagrangeův koeficient λ určíme z izoperimetrické podmínky (7.21).

Z předchozího je zřejmý **princip reciprocity**: Soustava extremál je táž, budeme-li hledat extrém funkcionálu J dle (7.4) za podmínky, že integrál v (7.21) je konstantní, nebo budeme hledat extrém funkcionálu v (7.21) za podmínky, že funkcionál v (7.4) má konstantní hodnotu.

Příklad 2: Nabíjení kondenzátoru
Mějme sériový elektrický obvod tvořený zdrojem napětí $v(t)$, rezistorem R a kondenzátorem C - viz obr. 7.1.



Obrázek 7.1: Elektrický RC obvod - nabíjení kondenzátoru.

Hledáme průběh takového impulsu napětí $u(t)$ na kondenzátoru C, aby průměrná hodnota napětí na kondenzátoru C byla maximální na intervalu $0 \leq t \leq T$. Kritérium je tedy

$$J = \frac{1}{T} \int_0^T u(t)\ dt \tag{7.23}$$

Hodnoty napětí $u(t)$ na konci a počátku nabíjení jsou dané $u(0) = 0$, $u(T) = U$. Při nabíjení kondenzátoru C přes rezistor R ze zdroje $v(t)$ jsme omezeni energií, kterou můžeme přeměnit na odporu R na teplo. Odtud plyne omezení

$$\int_0^T Ri^2(t)\ dt = K$$

kde $i(t)$ je proud tekoucí seriovým obvodem. Mezi proudem $i(t)$ a napětím $u(t)$ na kondenzátoru C platí vztah $i(t) = C\dfrac{du}{dt}$. Podle (7.22) zavedeme rozšířený funkcionál

$$J = \int_0^T \left(\frac{1}{T}u(t) + \lambda Ri^2(t)\right)\ dt = \int_0^T \left(\frac{1}{T}u(t) + \lambda RC^2\dot{u}^2(t)\right)\ dt \tag{7.24}$$

Eulerova - Lagrangeova rovnice má potom tvar

$$\frac{1}{T} - 2\lambda RC^2\ddot{u}^2(t) \ = \ 0.$$

Jejím řešením je zřejmě kvadratická funkce $u(t) = \alpha + \beta t + \gamma t^2$. Integrační konstanty α, β, γ určíme z okrajových podmínek a izoperimetrické podmínky. Po dosazení a úpravách dostaneme

$$\alpha = 0, \quad \beta = \frac{1}{T}U + \frac{1}{T}\sqrt{\frac{3KT}{RC^2} - 3U^2}, \quad \gamma = -\frac{1}{T^2}\sqrt{\frac{3KT}{RC^2} - 3U^2}$$

Úloha má reálné řešení pouze tehdy, je-li výraz pod odmocninou kladný, to je pro

$$K \geq \frac{U^2RC^2}{T}$$

Pokud není předchozí nerovnost splněna, nelze současně splnit izoperimetrickou podmínku a okrajové podmínky. Předchozí omezení plyne z následující úvahy: Pokud máme kondenzátoru dodat nějaký náboj $Q = UC$, kde $Q = \int_0^T i(t)\ dt$, pak výraz $\int_0^T Ri^2(t)\ dt$ je nejmenší, bude-li proud $i(t) = \text{konst.}$.

To prokážeme snadno řešením následující pomocné úlohy

$$\min \int_0^T Ri^2(t)\ dt, \quad \text{za podmínky} \quad \int_0^T i(t)\ dt = Q = UC = konst.$$

Rozšířený funkcionál pro tento problém

$$\bar{J} = \int_0^T \left(Ri^2(t) + \lambda i(t) \right)\ dt$$

vede na Eulerovu - Lagrangeovu rovnici $2Ri(t) + \lambda = 0$. Jejím řešením je zřejmě $i(t) = \frac{UC}{T} = konst.$. Energie přeměněná na odporu na teplo je nejmenší, bude-li proud $i(t)$ konstantní (chceme-li dodat kondenzátoru C náboj Q).

Proto izoperimetrická konstanta K v naší původní úloze musí být větší než

$$K \geq \int_0^T Ri^2(t)\ dt = \int_0^T R\left(\frac{UC}{T}\right)^2\ dt = \frac{RU^2C^2}{T^2}\int_0^T dt = \frac{U^2RC^2}{T}.$$

7.2.2 Volné koncové body

Často nejsou dány počáteční (t_0, x_0), nebo koncové body (t_1, x_1). Abychom mohli v těchto případech variační úlohy řešit, potřebujeme nějaké podmínky, které ve volném koncovém bodu musí platit. Tyto podmínky se nazývají **podmínky transverzality**.

Podmínky transverzality odvodíme z přírůstku funkcionálu, podobně jako jsme odvodili Eulerovu - Lagrangeovu rovnici. Úvahu provedeme pro volný koncový bod. Pro volný počáteční bod je postup úplně obdobný. Aby bod $(t_1,\ x_1)$ byl optimální koncový bod, musí být nulová lineární část přírůstku funkcionálu (7.4) při změně koncového času t_1 o $\varepsilon\ \delta t_1$ a změně koncového stavu x_1 o $\varepsilon\ \delta x_1$.

Přírůstek funkcionálu je roven

$$\Delta J = \int_{t_0}^{t_1} \left[g(x + \delta x,\ \dot{x} + \delta\dot{x}, t) - g(x,\ \dot{x},\ t) \right]\ dt + \int_{t_1}^{t_1+\delta t_1} g(x + \delta x,\ \dot{x} + \delta\dot{x}, t)\ dt \quad (7.25)$$

První integrál upravíme rozvojem jádra v řadu, pak platí

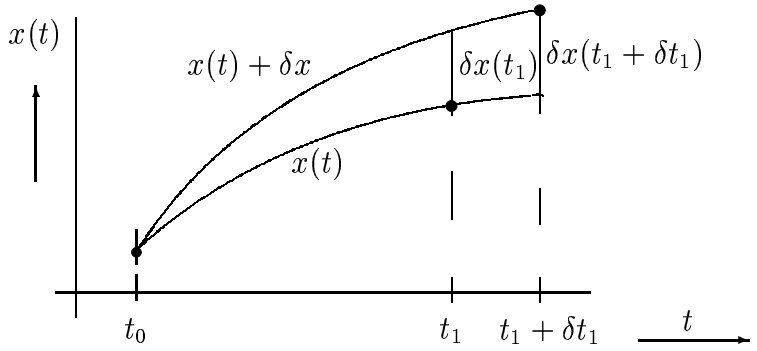
$$\int_{t_0}^{t_1} [\ .\ ]\ dt = \int_{t_0}^{t_1} (g_x\delta x + g_{\dot{x}}\delta\dot{x})\ dt \quad (7.26)$$

Integrací druhého členu per partes dostaneme

$$[g_{\dot{x}}\delta x]_{t_0}^{t_1} + \int_{t_0}^{t_1} \left(g_x - \frac{d}{dt}g_{\dot{x}} \right) \delta x\ dt$$

Jelikož extremála splňuje Eulerovu - Lagrangeovu rovnici je integrál v předchozím vztahu roven nule. Druhý integrál v (7.25) při zanedbání členů řádu $(\delta t_1)^2$ a vyšších je roven

$$\int_{t_1}^{t_1+\delta t_1} g(x+\delta x,\ \dot{x}+\delta\dot{x}, t)\ dt = g(x,\ \dot{x}, t)|_{t=t_1}\ \delta t_1.$$



Obrázek 7.2: Volný koncový bod - variace v koncovém bodě.

Přírůstek funkcionálu je po těchto úpravách roven první variaci funkcionálu, neboť uvažujeme pouze členy prvního řádu. Pak tedy

$$\Delta J \doteq \delta J = [g_{\dot{x}}\delta x]_{t=t_1} + g(x,\ \dot{x},\ t)|_{t=t_1}\ \delta t. \tag{7.27}$$

Přitom variaci v koncovém bodě můžeme podle obr. 7.2 vyjádřit pomocí totální variace ve tvaru

$$\delta x(t_1) \doteq \delta x(t_1+\delta t_1) - \dot{x}(t_1)\delta t_1$$

Totální variaci v koncovém bodě označíme δx, pak $\delta x = \delta x(t_1+\delta t_1)$. Potom po úpravě dostaneme obecnou podmínku transverzality ve volném koncovém bodu

$$\delta J = 0 = [(g - \dot{x}g_{\dot{x}})\ \delta t]_{t=t_1} + [g_{\dot{x}}\delta x]_{t=t_1} \tag{7.28}$$

Úplně obdobné podmínky platí pro volný počáteční bod trajektorie.

Z obecných podmínek transverzality plynou některé zvláštní případy:

1. **Volný konec trajektorie**
   Pokud je koncový čas t_1 volný, je variace δt v koncovém čase libovolná. Je-li také koncový stav $x(t_1)$ volný, je i variace δx v koncovém stavu libovolná. Z obecných podmínek transverzality (7.28) dostaneme v tomto případě

$$\begin{aligned} g - \dot{x}g_{\dot{x}} &= 0 \quad \text{pro} \quad t = t_1 \\ g_{\dot{x}} &= 0 \quad \text{pro} \quad t = t_1. \end{aligned} \tag{7.29}$$

   Pro volný počáteční bod platí obdobné podmínky.

2. **Pevný čas, volný konec**
   Pokud je pevný koncový čas t_1, pak je variace v koncovém čase nulová. Podmínky transverzality (7.28) se v tomto případě redukují na

$$g_{\dot{x}} = 0 \quad \text{pro} \quad t = t_1 \tag{7.30}$$

3. **Pevný konec, volný čas**
   Pokud je naopak $x(t_1) = x_1$ pevné, ale koncový čas t_1 je volný, pak z (7.28) plyne v tomto případě

$$g - \dot{x} g_{\dot{x}} = 0 \quad \text{pro} \quad t = t_1 \tag{7.31}$$

4. **Koncový bod leží na křivce** $x = \varphi(t)$
   V tomto případě nejsou variace δx a δt v koncovém bodě nezávislé. Zřejmě platí

$$\delta x(t_1) = \frac{d\ \varphi(t)}{dt}\ \delta t_1 = \dot{\varphi}(t)\delta t_1$$

   Po dosazení předchozího vztahu do (7.28) dostaneme podmínky transverzality pro koncový bod ležící na křivce $x = \varphi(t)$ ve tvaru

$$g + (\dot{\varphi}(t) - \dot{x})\, g_{\dot{x}} = 0 \quad \text{pro} \quad t = t_1 \tag{7.32}$$

5. **Pevné krajní body, pevná doba řešení** $t_1 - t_0 = \alpha$
   V tomto případě je volný počáteční čas t_0 i koncový čas t_1 s podmínkou, že jejich rozdíl je konstantní $t_1 - t_0 = \alpha$. Variace δt_0 a δt_1 jsou vázány podmínkou

$$(t_1 + \delta t_1) - (t_0 + \delta t_0) = \alpha$$

   Odtud $\delta t_0 = \delta t_1 = \delta t$. Podmínky transverzality jsou v tomto případě

$$[g - \dot{x} g_{\dot{x}}]_{t=t_1} - [g - \dot{x} g_{\dot{x}}]_{t=t_0} = 0 \tag{7.33}$$

Příklad 3: V minulém odstavci jsme vyřešili úlohu o optimálním nabíjení kondenzátoru C tak, aby střední hodnota napětí na kondenzátoru byla maximální při omezení tepelných ztrát na nabíjecím rezistoru R. Přitom byly pevně dány počáteční napětí $u(0)$ a koncové napětí $u(T) = U$ a také koncový a počáteční čas - viz příklad 2. Nyní vyřešíme stejnou úlohu, ale velikost napětí na konci nabíjení $u(T)$ není pevně dána. Jedná se tedy o úlohu s pevným koncovým časem a volným koncem trajektorie. Podle (7.24) je jádro funkcionálu $g(u, \dot{u}, t, \lambda) = \left(\frac{1}{T}u(t) + \lambda RC^2\dot{u}^2(t)\right)$. Podle (7.30) jsou podmínky transverzality

$$2\ \lambda\ R\ C^2\ \dot{u}(T)\ =\ 0$$

V předchozím příkladu jsme odvodili, že extremály jsou paraboly. Z podmínek transverzality plyne, že derivace napětí v koncovém čase je nulová $\dot{u}(T) = 0$. Snadno odvodíme, že extremála splňující počáteční podmínku $u(0) = 0$, izoperimetrickou podmínku i podmínku transverzality je $u^*(t) = \alpha + \beta t + \gamma t^2$, kde

$$\alpha = 0, \qquad \beta = -\frac{1}{C}\sqrt{\frac{3K}{RT}}, \qquad \gamma = \frac{1}{2CT}\sqrt{\frac{3K}{RT}}.$$

V tomto případě má úloha vždy řešení.

7.2.3 Další nutné a postačující podmínky

Legendrova podmínka

Eulerova - Lagrangeova rovnice byla odvozena z požadavku rovnosti nule první variace funkcionálu. Pro zjištění o jaký typ extrému se jedná je nutno připojit další nutné podmínky.

Jsou-li funkce $x^*(t)$ extremály (vyhovují Eulerově - Lagrangeově rovnici), pak, pokud mezi spojitými křivkami existuje maximum či minimum funkcionálu, můžeme jejich výběr omezit na funkce $x^*(t)$.

Je-li první variace funkcionálu rovna nule, pak podle (7.6) znaménko druhé variace funkcionálu určuje znaménko přírůstku funkcionálu. Aby extremála $x^*(t)$ byla minimem funkcionálu, pak přírůstek funkcionálu musí být nezáporný. Odtud plyne nutná podmínka pro relativní minimum

$$g_{\dot{x}\dot{x}} \geq 0, \tag{7.34}$$

případně pro relativní maximum

$$g_{\dot{x}\dot{x}} \leq 0. \tag{7.35}$$

Tyto nutné podmínky druhého řádu se nazývají **Legendrovy podmínky**. Jejich zesílená verze, to je

$$\begin{aligned} g_{\dot{x}\dot{x}} &> 0 \qquad \text{pro minimum} \\ g_{\dot{x}\dot{x}} &< 0 \qquad \text{pro maximum} \end{aligned} \tag{7.36}$$

jsou postačující podmínky pro slabé relativní maximum či minimum.

Příklad 4: Ověřte, že Legendrova podmínka je splněna v příkladu 1. Jaké jsou Legendrovy podmínky v příkladu 2?

Jacobiho podmínka

Extremály tvoří svazek křivek vycházejících z daného počátečního bodu $x(t_0)$. Tato soustava křivek tvoří pole. Požadujeme, aby v intervalu $t_0 \leq t \leq t_1$ se extremály v nějakém jiném bodě neprotínaly, čili aby tvořily tzv. **centrální pole**.

Protínají-li se extremály v nějakém jiném bodě než je střed svazku (to je bod $x(t_0)$), takový bod nazýváme **konjugovaný bod**, pak pole extremál není centrální.

Poznámka: Konjugované body soustavy křivek $x(t, c)$, kde c je parametr soustavy, tvoří obálku soustavy křivek, jejíž rovnice je

$$\frac{\partial x(t,c)}{\partial c} = y(t,c) = 0$$

Tak například pro soustavu parabol $x(t, c) = (t-c)^2$ je rovnice svazku $\frac{\partial x(t,c)}{\partial c} = 2(t-c) = 0$. Odtud $c = t$ a po dosazení do rovnice svazku dostaneme množinu konjugovaných bodů, která je v tomto případě rovna $x = 0$. Ukažte, že svazek přímek $x = ct$ má jediný konjugovaný bod a tím bodem je střed svazku $t = 0$, $x = 0$.

□

Nyní odvodíme podmínku, aby extremály tvořily centrální pole. Extremály vyhovují Eulerově - Lagrangeově rovnici $g_x - \frac{d}{dt}g_{\dot{x}} = 0$. Tuto rovnici budeme derivovat podle nějakého parametru c

$$\frac{d}{dc}g_x - \frac{d}{dc}\frac{d}{dt}g_{\dot{x}} = 0$$

Odtud plyne

$$\frac{\partial g_x}{\partial x}\frac{\partial x}{\partial c} + \frac{\partial g_x}{\partial \dot{x}}\frac{\partial \dot{x}}{\partial c} - \frac{d}{dt}\left(\frac{\partial g_{\dot{x}}}{\partial x}\frac{\partial x}{\partial c} + \frac{\partial g_{\dot{x}}}{\partial \dot{x}}\frac{\partial \dot{x}}{\partial c}\right) = 0$$

Po dosazení $\frac{\partial x}{\partial c} = u$ a úpravě dostaneme diferenciální rovnici

$$\left(g_{xx} - \frac{d}{dt}g_{x\dot{x}}\right)u - \frac{d}{dt}g_{\dot{x}\dot{x}}\dot{u} + g_{x\dot{x}}\dot{u} = 0$$

Odtud plyne konečný tvar Jacobiho diferenciální rovnice

$$g_{\dot{x}\dot{x}}\ \ddot{u} + \left[\frac{d}{dt}g_{\dot{x}\dot{x}} - g_{x\dot{x}}\right]\ \dot{u} + \left[\frac{d}{dt}g_{x\dot{x}} - g_{xx}\right]\ u = 0 \tag{7.37}$$

Předchozí rovnice je diferenciální rovnice druhého řádu pro funkci $u(t) = \frac{\partial x}{\partial c}$.

Jacobiho podmínka je splněna, pokud $u(t) \neq 0$ kromě počátečního bodu v čase t_0. Jacobiho podmínka je nutnou podmínkou extrému funkcionálu. Vyhovuje-li pole extremál Jacobiho podmínce, potom jsou-li dány počáteční a koncové body trajektorie, existuje pouze jediná extremála, která splňuje obě okrajové podmínky.

Příklad 5: Ověřte, zda je splněna Jacobiho podmínka v úloze na minimum kvadratické regulační plochy - viz příklad 1. Funkcionál, jehož minimum hledáme je

$$J = \int_0^\infty \left(x^2 + (T\dot{x})^2\right)\ dt, \qquad x(0) = x_0,\ x(\infty) = 0$$

Jakobiho rovnice je dle (7.37)

$$2T^2\ddot{u}(t) + 0\dot{u}(t) - 2u(t) = 0$$

Její řešení je

$$u(t) = c_1 e^{\frac{t}{T}} + c_2 e^{-\frac{t}{T}}$$

Z podmínky $u(0) = 0$ plyne $c_1 = -c_2 = c$. Kromě středu svazku v bodě $u(0)$ je Jakobiho podmínka $u(t) \neq 0$ splněna. Pole extremál tvoří centrální pole se středem svazku v bodě $x(0) = x_0$.

Weierstrassova podmínka

Hledáme znaménko přírůstku funkcionálu při přechodu od extremály (křivky, kterou označíme C) k nějaké jiné, blízké křivce (označíme ji C_1). Pak

$$\Delta J\ =\ \int_{C_1} g(x, \dot{x}, t)\ dt - \int_C g(x, \dot{x}, t)\ dt$$

Tento přírůstek funkcionálu můžeme po úpravě vyjádřit ve tvaru

$$\Delta J \;=\; \int_C E(x,z,s,t)\; dt \tag{7.38}$$

kde **Weierstrassova funkce** E je rovna

$$E \;=\; g(x,z,t) - g(x,s,t) - (z-s)\frac{\partial g(x,s,t)}{\partial s} \tag{7.39}$$

kde s je derivace $\dot{x}(t)$ na extremále a
z je libovolná jiná derivace funkce $x(t)$.

Postačující podmínkou, aby funkcionál J nabýval na extremále minima, je podle (7.38) nezápornost Weierstrassovy funkce E

$$E \geq 0 \tag{7.40}$$

Pro slabé minimum stačí, aby nerovnost $E \geq 0$ byla splněna pro x, $z = \dot{x}$ blízké k extremále C. V tomto případě můžeme funkci $g(x,\dot{x},t)$ rozvinout do řady

$$g(x,z,t) = g(x,s,t) + \frac{\partial g(x,s,t)}{\partial s}\,(z-s) + \frac{\partial^2 g(x,s,t)}{\partial s^2}\;\frac{(z-s)^2}{2},$$

kde q leží mezi z a s. Weierstrassova funkce má potom tvar

$$E(x,z,s,t) = g_{\dot{x}\dot{x}}(x,q,t)\;\frac{(z-s)^2}{2},$$

Weierstrassovu podmínku (7.40) můžeme potom podle předchozího vztahu nahradit zesílenou Legendrovou podmínkou (7.36). Pro silné minimum musí být nerovnost $E \geq 0$ splněna pro x, t blízké k bodům extremály C, ale pro libovolné hodnoty $z = \dot{x}(t)$.

Příklad 6: Prověřte, zda je splněna Legendrova i Weierstrassova podmínka při minimalizaci obecné kvadratické plochy podle příkladu 1.

Zesílená Legendrova podmínka je

$$\frac{\partial^2 g(x,\dot{x},t)}{\partial x^2} = 2/T^2 > 0$$

pro $T \neq 0$, což vyhovuje předpokladu. Weierstrassova funkce je pro daný problém

$$E(x,z,s,t) = x^2 + T^2z^2 - x^2 - T^2s^2 - (z-s)2T^2s = T^2(z-s)^2$$

Z předchozího plyne, že Legendrova i Weierstrassova podmínka jsou splněny pro libovolné $\dot{x}(t) = z$. Extremála, která je řešením Eulerovy - Lagrangeovy rovnice tvoří silné relativní minimum.

Úhlové body

Extremály, které získáme řešením Eulerovy - Lagrangeovy rovnice jsou spojité křivky, které mají spojité derivace. Přitom extremála se může skládat z úseků, z nichž každý je řešením Eulerovy rovnice. Přitom v místě styku dvou extremál může dojít k tomu, že derivace není v bodě styku jednoznačně určena (derivace zprava a zleva jsou navzájem různé). Takovému bodu říkáme **úhlový bod**. V úhlovém bodě má derivace extremály nespojitost prvního druhu.

Budeme hledat podmínky, které musí být splněny, aby extremála měla úhlový bod. Předpokládejme, že v bodě $t = t_a$ má extremála úhlový bod. Funkcionál J můžeme vyjádřit ve tvaru

$$J = \int_{t_0}^{t_a} g(x, \dot{x}, t)\ dt + \int_{t_a}^{t_1} g(x, \dot{x}, t)\ dt$$

Je-li úhlových bodů více, rozdělíme funkcionál na více dílčích částí. Mezi úhlovými body extremály vyhovují Eulerově - Lagrangeově rovnici. Podobně jako v úloze s volnými konci můžeme předpokládat, že t_a a $x(t_a)$ jsou volné a můžeme je uvažovat jako proměnné koncové a počáteční podmínky integrálů v předchozím vztahu. Potom musí platit podmínky transverzality v bodě $t = t_a$, z nichž plyne

$$(g - \dot{x} g_{\dot{x}})\ \delta t\big|_{t=t_a+0}^{t=t_a-0} + g_{\dot{x}}\ \delta x\big|_{t=t_a+0}^{t=t_a-0} = 0$$

Jelikož variace δt a $\delta x(t_a)$ jsou nezávislé, plynou z předchozí rovnice dvě podmínky

$$\begin{aligned} (g - \dot{x} g_{\dot{x}})_{t=t_a-0} &= (g - \dot{x} g_{\dot{x}})_{t=t_a+0} \\ g_{\dot{x}}\big|_{t=t_a-0} &= g_{\dot{x}}\big|_{t=t_a+0} \end{aligned} \tag{7.41}$$

Předchozím podmínkám říkáme **Weierstrassovy - Erdmannovy podmínky**.

Příklad 7: Ověřte, zda existují úhlové body extremál, které minimalizují funkcionál

$$\int_0^{t_1} \left(\dot{x}^4 - 6\dot{x}^2\right)\ dt, \quad x(0) = 0, \quad x(t_1) = x_1$$

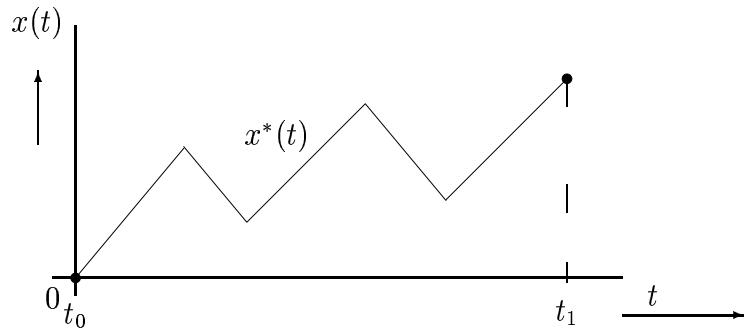
Snadno se přesvědčíme, že extremály jsou přímky. Weierstrassovy - Erdmannovy podmínky jsou

$$\begin{aligned} \dot{x}^4 - 6\dot{x}^2 - \dot{x}\left(4\dot{x}^3 - 12\dot{x}\right)\Big|_{t=t_a-0} &= \dot{x}^4 - 6\dot{x}^2 - \dot{x}\left(4\dot{x}^3 - 12\dot{x}\right)\Big|_{t=t_a+0} \\ 4\dot{x}^3 - 12\dot{x}\Big|_{t=t_a-0} &= 4\dot{x}^3 - 12\dot{x}\Big|_{t=t_a+0} \end{aligned}$$

Označíme-li $\dot{x}|_{t=t_a-0} = u$, $\dot{x}|_{t=t_a+0} = v$ dostaneme předchozí podmínky ve tvaru

$$\begin{aligned} 3u^2(2 - u^2) &= 3v^2(2 - v^2) \\ 4u(u^2 - 3) &= 4v(v^2 - 3) \end{aligned}$$

První rovnice bude splněna, pokud $|u| = |v|$, z druhé rovnice kromě toho plyne $u = \pm\sqrt{3}$, $v = \pm\sqrt{3}$. Úhlové body extremály mohou být libovolné. Extremály jsou tedy lomené křivky se směrnicemi $\pm\sqrt{3}$ - viz obr. 7.3.



Obrázek 7.3: Extremály z příkladu 7 jsou lomené křivky.

7.3 Rozšíření základní úlohy

7.3.1 Extrémy funkcionálu v n-rozměrném prostoru

Hledejme extrém funkcionálu

$$J = \int_{t_0}^{t_1} g\left(\mathbf{x}, \dot{\mathbf{x}}, t\right)\ dt \tag{7.42}$$

kde nyní $\mathbf{x}(t) = [x_1(t), \ldots, x_n(t)]^T$ je vektorová funkce.

Protože jednotlivé složky vektoru $\mathbf{x}$ jsou nezávislé, platí nutná podmínka extrému funkcionálu ve tvaru Eulerovy - Lagrangeovy rovnice pro každou složku, tedy

$$g_{x_i} - \frac{d}{dt} g_{\dot{x}_i} = 0, \qquad i = 1, 2, \ldots, n, \tag{7.43}$$

kterou můžeme zapsat také ve tvaru $g_{\mathbf{X}} - \frac{d}{dt} g_{\dot{\mathbf{X}}} = 0$, kde $\mathbf{x}$ je vektor rozměru n.

Soustava (7.43) je soustava n obyčejných diferenciálních rovnic druhého řádu. Integrační konstanty, kterých je $2n$ určíme z n počátečních a n koncových podmínek $\mathbf{x}(t_0) = \mathbf{x}_0$, $\mathbf{x}(t_1) = \mathbf{x}_1$.

Legendrova podmínka vyžaduje pozitivní semidefinitnost matice druhých parciálních derivací

$$\frac{\partial^2 g}{\partial \dot{\mathbf{x}}^2} = \begin{bmatrix} \frac{\partial^2 g}{\partial \dot{x}_1 \dot{x}_1} & \cdots & \frac{\partial^2 g}{\partial \dot{x}_1 \dot{x}_n} \\ \vdots & & \vdots \\ \frac{\partial^2 g}{\partial \dot{x}_n \dot{x}_1} & \cdots & \frac{\partial^2 g}{\partial \dot{x}_n^2} \end{bmatrix} \tag{7.44}$$

Pro volný koncový bod platí podmínky transverzality ve tvaru

$$\left(g - \sum_{i=1}^{n} g_{\dot{x}_i} \dot{x}_i\right) \delta t + \sum_{i=1}^{n} g_{\dot{x}_i} \delta x_i = 0, \qquad \text{pro} \quad t = t_1. \tag{7.45}$$

Extremála může mít úhlové body. Podmínky, které musí být splněny v úhlovém bodě jsou

$$\begin{aligned} \delta g_{\dot{x}_i}\big|_{t=t_a-0} &= \delta g_{\dot{x}_i}\big|_{t=t_a+0}, \qquad i = 1, 2, \ldots, n \\ g - \sum_{i=1}^{n} g_{\dot{x}_i} \dot{x}_i \Big|_{t=t_a-0} &= g - \sum_{i=1}^{n} g_{\dot{x}_i} \dot{x}_i \Big|_{t=t_a+0} \end{aligned} \tag{7.46}$$

Je-li funkcionál závislý na vyšších derivacích

$$J = \int_{t_0}^{t_1} g\left(\mathbf{x}, \dot{\mathbf{x}}, \ldots, \mathbf{x}^m, t\right)\ dt \tag{7.47}$$

je nutnou podmínkou extrému funkcionálu tzv. **Euler - Poissonova** rovnice

$$g_{\mathbf{x}} - \frac{d}{dt} g_{\dot{\mathbf{x}}} + \ldots + (-1)^m \frac{d^m}{dt^m} g_{\mathbf{x}^{(m)}} = \mathbf{0}. \tag{7.48}$$

Předchozí rovnice je vektorová diferenciální rovnice řádu $2m$. Integrační konstanty určíme z m počátečních a m koncových vektorových podmínek.

7.3.2 Variační problémy s omezením

Jednu variační metodu s omezením jsme již řešili. Jednalo se o izoperimetrickou úlohu s omezením ve tvaru integrálu, kterou jsme řešili zavedením konstantního Lagrangeova koeficientu a rozšířeného funkcionálu.

Často je omezující podmínka určena algebraickou či diferenciální rovnicí. Hledejme tedy extrém funkcionálu (7.42) při omezení ve tvaru diferenciální rovnice

$$f_j\left(\mathbf{x}, \dot{\mathbf{x}}, t\right) = 0, \qquad j = 1, 2, \ldots, m \tag{7.49}$$

Tomuto problému říkáme **Lagrangeova úloha**. Tuto úlohu řešíme zavedením Lagrangeova vektoru $\boldsymbol{\lambda}(t) = [\lambda_1(t), \ldots, \lambda_m(t)]$. Extremály Lagrangeovy úlohy jsou extremálami funkcionálu

$$\bar{J} = \int_{t_0}^{t_1} \left[g(\mathbf{x}, \dot{\mathbf{x}}, t) + \boldsymbol{\lambda}^T(t) \mathbf{f}(\mathbf{x}, \dot{\mathbf{x}}, t) \right]\ dt \tag{7.50}$$

kde $\mathbf{f} = [f_1, \ldots, f_n]^T$ je vektor omezení (7.49).

Lagrangeovu úlohu řešíme tak, že pro funkcionál (7.50) napíšeme Eulerovy - Lagrangeovy rovnice a jejich řešením dostaneme extremály původní úlohy. Podrobnější rozbor této úlohy bude proveden v následujících odstavcích.

Poznámka: Omezení nerovnicí

$$f(\mathbf{x}, \dot{\mathbf{x}}, t) \leq 0 \tag{7.51}$$

můžeme převést na omezení typu rovnosti zavedením další složky $x_{n+1}(t)$ vektoru $\mathbf{x}$. Předchozí omezení je ekvivalentní omezení

$$f(\mathbf{x}, \dot{\mathbf{x}}, t) + x_{n+1}^2 = 0. \tag{7.52}$$

Podobně omezení ve tvaru oboustranné nerovnosti

$$\alpha \leq f(\mathbf{x}, \dot{\mathbf{x}}, t) \leq \beta, \qquad \alpha < \beta \tag{7.53}$$

převedeme na omezení rovnosti ve tvaru

$$(f - \alpha)(\beta - f) - x_{n+1}^2 = 0, \tag{7.54}$$

opět zavedením další složky vektoru $\mathbf{x}$.

Příklad 8: Mějme opět problém nabíjení kondenzátoru - viz příklad 2. Nyní hledáme maximum střední hodnoty napětí na kondenzátoru, ale při omezení energie E_N nabíjecího zdroje $v(t)$

$$E_N = \int_0^T i(t)v(t)\ dt = \int_0^T C\dot{u}(t)v(t)\ dt \tag{7.55}$$

Napětí $u(t)$ na kondenzátoru C a napětí zdroje $v(t)$ jsou vázány diferenciální rovnicí

$$v(t) = Ri(t) + u(t), \qquad i(t) = C\dot{u}(t) \tag{7.56}$$

Počáteční podmínka je $u(0) = 0$ a hodnota napětí $u(t)$ v koncovém čase $t_1 = T$ není určena.

Problémem je tedy maximalizovat funkcionál (7.23) při izoperimetrické podmínce (7.55) a vazební podmínce (7.56). Sestavíme tedy rozšířený funkcionál

$$\bar{J} = \int_0^T \left[\frac{1}{T}u(t) + \lambda_1 C\dot{u}(t)v(t) + \lambda_2(t)\left(v(t) - u(t) - RC\dot{u}(t)\right)\right]\ dt \tag{7.57}$$

kde $\lambda_1 =$ konst. a $\lambda_2(t)$ je Lagrangeův koeficient resp. Lagrangeova funkce. Předchozí funkcionál je závislý na $(u(t), v(t), \lambda_2(t))$ a proto napíšeme Eulerovu - Lagrangeovu rovnici pro každou proměnnou zvlášť:

$$\begin{array}{rcrl}
\text{E-L rce pro } u(t) & : & \frac{1}{T} - \lambda_2(t) + RC\dot{\lambda}_2(t) - \lambda_1 C\dot{v}(t) & = 0 \\
\text{E-L rce pro } v(t) & : & \lambda_1 C\dot{u}(t) + \lambda_2(t) & = 0 \\
\text{E-L rce pro } \lambda_2(t) & : & v(t) - u(t) - RC\dot{u}(t) & = 0
\end{array}$$

Řešením této soustavy tří diferenciálních rovnic prvního řádu získáme extremály $u^*(t), v^*(t), \lambda_2^*(t)$ (ověřte, že $u^*(t) = \alpha + \beta t + \gamma t^2$). Integrační konstanty určíme z počáteční podmínky $u(0) = 0$, izoperimetrické podmínky (7.55) a podmínky transverzality, která je zde tvaru

$$\bar{g}_{\dot{u}} = \lambda_1 C v(T) - \lambda_2(T)RC = 0$$

kde $\bar{g}$ je jádro rozšířeného funkcionálu (7.57).

7.3.3 Lagrangeova, Mayerova a Bolzova úloha

Známe tři základní typy variačních úloh.

Lagrangeova úloha spočívá v minimalizaci funkcionálu

$$J_1\left(\mathbf{x}(t)\right) = \int_{t_0}^{t_1} g\left(\mathbf{x}, \dot{\mathbf{x}}, t\right)\ dt \tag{7.58}$$

s omezením ve tvaru

$$\mathbf{f}\left(\mathbf{x}, \dot{\mathbf{x}}, t\right) = 0 \tag{7.59}$$

Lagrangeovu úlohu řešíme zavedením rozšířeného funkcionálu podle (7.50).

Mayerova úloha spočívá v minimalizaci funkcionálu

$$J_2\left(\mathbf{x}(t)\right) = \left[h\left(\mathbf{x},t\right)\right]_{t_0}^{t_1} = h\left(\mathbf{x}(t_1),t_1\right) - h\left(\mathbf{x}(t_0),t_0\right) \tag{7.60}$$

s omezením (7.59).

Bolzova úloha je kombinací obou předchozích typů úloh a spočívá v minimalizaci funkcionálu

$$J_3\left(\mathbf{x}(t)\right) = \left[h\left(\mathbf{x},t\right)\right]_{t_0}^{t_1} + \int_{t_0}^{t_1} g\left(\mathbf{x},\dot{\mathbf{x}},t\right)\ dt \tag{7.61}$$

s omezením (7.59).

Pro Lagrangeovu úlohu mohou být oba koncové body pevné. Jsou-li volné, potom pro jejich určení použijeme podmínky transverzality. V Mayerově úloze je vždy alespoň jeden koncový bod volný. Jeho určením minimalizujeme funkcionál (7.60). Bolzova úloha vyžaduje opět volné koncové body. Jsou-li koncové body pevné, je konstantní neintegrální člen v kritériu (7.61) a proto jej při minimalizaci nemusíme uvažovat. Bolzova úloha přechází potom v úlohu Lagrangeovu.

Mezi těmito typy úloh existuje těsná souvislost. Mayerovu úlohu můžeme převést na úlohu Lagrangeovu. Je-li funkce $h(\mathbf{x},t)$ v (7.60) diferencovatelná, pak funkcionál (7.60) můžeme vyjádřit ve tvaru

$$J_2\left(\mathbf{x}(t)\right) = \int_{t_0}^{t_1} \frac{dh\left(\mathbf{x},t\right)}{dt}\ dt = \int_{t_0}^{t_1} \left(\frac{\partial h\left(\mathbf{x},t\right)}{\partial t} + \frac{\partial h}{\partial \mathbf{x}}\dot{\mathbf{x}}\right)\ dt \tag{7.62}$$

Mayerova úloha přešla na úlohu Lagrangeovu ve tvaru (7.58).

Obráceně můžeme Lagrangeovu úlohu převést na úlohu Mayerovu. Zavedeme si novou souřadnici $x_{n+1}(t)$, určenou diferenciální rovnicí

$$\dot{x}_{n+1}(t) = g\left(\mathbf{x},t\right), \qquad x_{n+1}(t_0) = 0 \tag{7.63}$$

Lagrangeova úloha minimalizace funkcionálu (7.58) se pomocí (7.63) změní na Mayerovu úlohu minimalizace souřadnice $x_{n+1}(t_1)$, neboť

$$J_1 = \int_{t_0}^{t_1} g(.)\ dt = \int_{t_0}^{t_1} \dot{x}_{n+1}(t)\ dt = x_{n+1}(t_1) - x_{n+1}(t_0) = x_{n+1}(t_1) \tag{7.64}$$

Stejným postupem můžeme převést Bolzovu úlohu na úlohu Mayerovu. Obdobným postupem bychom modifikovali podmínky transverzality.

7.4 Řešení problému optimálního řízení dynamických systémů

Problém optimálního řízení spojitých dynamických systémů byl formulován na počátku této kapitoly. Jedná se zřejmě o variační problém Bolzova typu. Pokud jsou nějaká omezení na stavy či řízení, je tento poblém obtížně řešitelný klasickými variačními metodami.

7.4.1 Optimální řízení bez omezení

Odvodíme si vztahy pro řešení problému optimálního řízení spojitých dynamických systémů, kde dovolená množina řízení a stavů není omezená, pak $\mathbf{U} \equiv R^r$, $\mathbf{X} \equiv R^n$. Mějme tedy kritérium kvality řízení ve tvaru

$$J\left(\mathbf{x},\mathbf{u}\right) = \int_{t_0}^{t_1} g(\mathbf{x},\mathbf{u},t)\ dt \tag{7.65}$$

s omezením daným pouze stavovou rovnicí spojitého systému

$$\dot{\mathbf{x}}(t) = \mathbf{f}\left(\mathbf{x}(t),\mathbf{u}(t),t\right), \qquad \mathbf{x}(t_0) = \mathbf{x}_0. \tag{7.66}$$

Problém určení optimálního řízení $\mathbf{u}(t)$ minimalizující (7.65) a respektující omezení (7.66) je variační problém Lagrangeova typu. Člen $h(\mathbf{x}(t_1))$ v kritériu zatím neuvažujeme. Podle (7.50) zavedeme rozšířený funkcionál

$$\bar{J} = \int_{t_0}^{t_1} \left[g(\mathbf{x},\mathbf{u},t) + \boldsymbol{\lambda}^T\left(\dot{\mathbf{x}} - \mathbf{f}(\mathbf{x},\mathbf{u},t)\right)\right]\ dt \tag{7.67}$$

Jádro funkcionálu označíme ϕ, pak

$$\phi(\mathbf{x},\mathbf{u},\boldsymbol{\lambda},t) = g(\mathbf{x},\mathbf{u},t) + \boldsymbol{\lambda}^T\left(\dot{\mathbf{x}} - \mathbf{f}(\mathbf{x},\mathbf{u},t)\right). \tag{7.68}$$

Extremály $\mathbf{u}^*(t)$, $\mathbf{x}^*(t)$, $\boldsymbol{\lambda}^*(t)$ splňují Eulerovy - Lagrangeovy rovnice, které mají tvar

$$\begin{aligned}
\frac{\partial\phi}{\partial\mathbf{x}} - \frac{d}{dt}\frac{\partial\phi}{\partial\dot{\mathbf{x}}} &= 0\\
\frac{\partial\phi}{\partial\mathbf{u}} - \frac{d}{dt}\frac{\partial\phi}{\partial\dot{\mathbf{u}}} &= 0\\
\frac{\partial\phi}{\partial\boldsymbol{\lambda}} - \frac{d}{dt}\frac{\partial\phi}{\partial\dot{\boldsymbol{\lambda}}} &= 0
\end{aligned} \tag{7.69}$$

Rovnice (7.69a) v předchozí soustavě je soustava n diferenciálních rovnic prvního řádu. Druhý člen v (7.69b) je nulový, neboť jádro ϕ neobsahuje explicitně $\dot{\mathbf{u}}(t)$ a proto rovnice (7.69b) není diferenciální rovnicí. Proto (7.69b) je soustava r algebraických rovnic, ze kterých určíme optimální řízení $\mathbf{u}^*(\mathbf{x}^*,\boldsymbol{\lambda}^*,t)$. Také v (7.69c) je druhý člen nulový, neboť jádro ϕ neobsahuje explicitně $\dot{\boldsymbol{\lambda}}(t)$. Rovnice (7.69c) je soustava n diferenciálních rovnic, totožných se stavovými rovnicemi systému (7.66).

Dosadíme-li za ϕ ze (7.68) do (7.69) dostaneme po rozepsání do složek soustavu diferenciálních a algebraických rovnic

$$\begin{aligned}
\frac{\partial g}{\partial x_j} - \sum_{i=1}^{n}\lambda_i\frac{\partial f_i}{\partial x_j} - \dot{\lambda}_j &= 0, \quad j = 1,2,\ldots,n\\
\frac{\partial g}{\partial u_k} - \sum_{i=1}^{n}\lambda_i\frac{\partial f_i}{\partial u_k} &= 0 \quad k = 1,2,\ldots,r\\
\dot{x}_i - f_i(\mathbf{x},\mathbf{u},t) &= 0 \quad i = 1,2,\ldots,n
\end{aligned} \tag{7.70}$$

Soustava diferenciálních rovnic (7.70a) je soustava diferenciálních rovnic pro složky vektoru $\lambda(t)$ a nazývá se rovnice **konjugovaného systému**. Integrováním soustavy diferenciálních rovnic konjugovaného systému (7.70a) a diferenciálních rovnic systému (7.70c),

spolu se soustavou r algebraických rovnic (7.70b) získáme optimální funkce $\mathbf{x}^*(t)$, $\mathbf{u}^*(t)$ $\boldsymbol{\lambda}^*(t)$. Integrační konstanty, kterých je $2n$ vypočteme z n počátečních podmínek $\mathbf{x}(t_0)$ a n koncových podmínek $\mathbf{x}(t_1)$.

Jsou-li volné koncové body, použijeme pro určení integračních konstant podmínky transverzality. Ty jsou podle (7.45) pro volný konec $(\mathbf{x}(t_1), t_1)$ rovny

$$\left(\phi - \sum_{i=1}^{n} \phi_{\dot{x}_i} \dot{x}_i\right) \delta t + \sum_{i=1}^{n} \phi_{\dot{x}_i} \delta x_i = 0, \qquad \text{pro} \quad t = t_1. \tag{7.71}$$

Po dosazení za ϕ z (7.68) dostaneme

$$\left(g - \sum_{i=1}^{n} \lambda_i f_i\right) \delta t + \sum_{i=1}^{n} \lambda_i \delta x_i = 0, \qquad \text{pro} \quad t = t_1. \tag{7.72}$$

Je-li t_0, t_1, $\mathbf{x}(t_0)$ pevné a pouze $\mathbf{x}(t_1)$ je volné, pak z (7.72) plyne

$$\sum_{i=1}^{n} \lambda_i \delta x_i = 0, \qquad \text{pro} \quad t = t_1. \tag{7.73}$$

Jelikož variace δx_i je ve volném koncovém bodě libovolná, pak z předchozího plyne

$$\lambda_i(t_1) = 0. \tag{7.74}$$

Je-li tedy koncový bod $\mathbf{x}(t_1)$ volný, pak je nulová koncová podmínka konjugovaného systému.

Je-li konec trajektorie určen křivkou $\varphi(\mathbf{x}) = 0$, pak přípustná variace $\delta\mathbf{x}$ je možná jen ve směru kolmém ke gradientu funkce $\varphi(\mathbf{x})$. Platí tedy

$$\frac{\partial \varphi}{\partial \mathbf{x}} \, \delta\mathbf{x} \;=\; 0 \tag{7.75}$$

Podmínky transverzality (7.73) můžeme zapsat ve tvaru $\boldsymbol{\lambda}^T \delta\mathbf{x}\Big|_{t=t_1} = 0$. Odtud porovnáním s (7.75) plyne

$$\boldsymbol{\lambda}(t_1) = \alpha \left(\frac{\partial \varphi}{\partial \mathbf{x}}\right)^T, \tag{7.76}$$

což znamená, že vektor $\boldsymbol{\lambda}(t_1)$ v koncovém čase má stejný směr jako gradient omezení.

Uvědomme si, že diferenciální rovnice (7.70a) a (7.70c) nám spolu s algebraickou rovnicí (7.70b) neumožňují řešit problém optimálního řízení v reálném čase (on line). Pro stavovou rovnici systému (7.70c) známe počáteční podmínku, ale pro konjugovaný systém (7.70a) počáteční podmínku $\boldsymbol{\lambda}(t_0)$ neznáme.

Pro jednoznačnost trajektorie známe až koncovou podmínku systému $\mathbf{x}(t_1)$, nebo, je-li konec volný, známe podle (7.74) či (7.76) koncovou podmínku $\boldsymbol{\lambda}(t_1)$ konjugovaného systému.

Vztahy (7.70) převedou problém optimálního řízení na řešení okrajového problému soustavy diferenciálních rovnic. Analytické řešení je totiž proveditelné pouze pro specielní problémy (jako na příklad problém optimálního řízení lineárního systému s kvadratickým kritériem optimality, který je známý pod zkratkou **LQ řízení**).

Chceme-li řešit problém optimálního řízení v reálném čase, nezbývá nám než počáteční podmínku $\boldsymbol{\lambda}(t_0)$ konjugovaného systému zvolit a řešit diferenciální rovnice (7.70a) a (7.70c) spolu s (7.70b) až do koncového času $t = t_1$. Pokud $\mathbf{x}(t_1)$ není rovno koncové podmínce $\mathbf{x}_1$, pak musíme celé řešení znovu opakovat s vhodně upravenou počáteční podmínkou $\boldsymbol{\lambda}(t_0)$ konjugovaného systému. Tento postup musíme opakovat tak dlouho, až $\mathbf{x}(t_1) = \mathbf{x}_1$. Konvergence tohoto postupu není zaručena a téměř vždy je velmi pomalá.

Mějme nyní obecný problém optimálního řízení Bolzova typu s kritériem

$$J\left(\mathbf{x}(t), \mathbf{u}(t)\right) = h\left(\mathbf{x}(t_1)\right) + \int_{t_0}^{t_1} g\left(\mathbf{x}, \mathbf{u}, t\right)\ dt \tag{7.77}$$

Tento problém převedeme podle (7.62) na problém Lagrangeova typu. Při respektování omezení daném stavovou rovnicí systému dostaneme rozšířený funkcionál ve tvaru

$$\bar{J}\left(\mathbf{x}(t), \mathbf{u}(t)\right) = \int_{t_0}^{t_1} \left[g\left(\mathbf{x}, \mathbf{u}, t\right) + \boldsymbol{\lambda}^T \left(\dot{\mathbf{x}} - \mathbf{f}(\mathbf{x}, \mathbf{u}, t)\right) + \frac{\partial h}{\partial t} + \frac{\partial h}{\partial \mathbf{x}}\dot{\mathbf{x}} \right]\ dt \tag{7.78}$$

Eulerovy - Lagrangeovy rovnice zůstanou stejné jako (7.70). Ověřte!

Pro volný konec trajektorie $\mathbf{x}(t_1)$ platí podmínky transverzality $\phi_{\dot{\mathbf{x}}} = 0$ pro $t = t_1$, kde ϕ je jádro funkcionálu (7.78). Odtud plyne

$$\boldsymbol{\lambda}(t_1) = -\left(\frac{\partial h}{\partial \mathbf{x}}\right)^T \Bigg|_{t=t_1} . \tag{7.79}$$

Příklad 9: Optimální doběh stejnosměrného motoru.
Stejnosměrný motor s konstantním buzením, řízený napětím na kotvě, je při zanedbání nelinearit a reakce kotvy popsán soustavou rovnic

$$\begin{aligned} U &= RI + k_1 n \\ k_2 I &= J\frac{dn}{d\tau} \end{aligned}$$

kde $U(\tau)$, $I(\tau)$, R je napětí, proud a odpor vinutí kotvy, J je moment setrvačnosti a n jsou otáčky hřídele motoru a konečně k_1, k_2 jsou konstanty motoru.

Zavedeme poměrné hodnoty

$$u(t) = \frac{U}{U_0}, \quad i(t) = \frac{I}{I_0}, \quad t = \frac{\tau}{T}$$

kde veličiny označené indexem nula jsou jmenovité hodnoty a T je elektromechanická časová konstanta. Potom jsou rovnice motoru v bezrozměrném tvaru

$$\begin{aligned} u(t) &= i(t) + \omega(t) \\ \dot{\omega}(t) &= i(t) = u(t) - \omega(t) \end{aligned}$$

Budeme hledat optimální řízení $u^*(t)$, které zabrzdí motor, to je převede předchozí systém ze stavu $\omega(0) = \omega_0$ do stavu $\omega(\infty) = 0$. Kritérium optimality zvolíme ve tvaru

$$J = \nu \int_0^\infty u(t)i(t)\ dt + \mu \int_0^\infty \omega^2(t)\ dt$$

První člen kritéria je úměrný energii dodané ze zdroje a druhý člen je úměrný kvadratické ploše odchylky. Snadno se můžeme přesvědčit, že pro $\nu = 0$ neexistují spojité extremály. Pokud tedy $\nu \neq 0$, kritérium můžeme upravit do tvaru

$$J = \int_0^\infty \left(u(t)i(t) + \alpha\omega^2(t)\right)\ dt = \int_0^\infty \left(u^2(t) - u(t)\omega(t) + \alpha\omega^2(t)\right)\ dt$$

Jedná se zřejmě o variační problém Lagrangeova typu. Sestavíme rozšířený funkcionál

$$\bar{J} = \int_0^\infty \left(u^2(t) - u(t)\omega(t) + \alpha\omega^2(t) + \lambda(t)\left(\dot{\omega}(t) - u(t) + \omega(t)\right)\right)\ dt$$

Eulerovy - Lagrangeovy rovnice jsou

$$\begin{aligned} -u(t) + 2\alpha\omega(t) + \lambda(t) - \dot{\lambda}(t) &= 0 \\ 2u(t) - \omega(t) - \lambda &= 0 \\ \dot{\omega}(t) - u(t) + \omega(t) &= 0 \end{aligned}$$

Z druhé, algebraické rovnice, plyne řízení

$$u^*(t) = \frac{1}{2}\left(\omega + \lambda\right)$$

Po dosazení optimálního řízení do zbylých rovnic dostaneme

$$\begin{aligned} \dot{\omega}(t) &= \frac{1}{2}\left(\lambda(t) - \omega(t)\right) \\ \dot{\lambda}(t) &= \omega(t)\left(2\alpha - \frac{1}{2}\right) + \frac{1}{2}\lambda(t) \end{aligned}$$

Vlastní čísla matice této soustavy jsou $\mu_{1,2} = \pm\sqrt{\alpha}$. Řešení potom můžeme psát ve tvaru

$$\begin{aligned} \omega(t) &= c_1\Delta_1(\mu_1)e^{-\sqrt{\alpha}t} + c_2\Delta_1(\mu_2)e^{\sqrt{\alpha}t} \\ \lambda(t) &= c_1\Delta_2(\mu_1)e^{-\sqrt{\alpha}t} + c_2\Delta_2(\mu_2)e^{\sqrt{\alpha}t} \end{aligned}$$

kde c_1, c_2 jsou integrační konstanty a $\Delta_i(\mu_j)$ je subdeterminant i-tého sloupce libovolného řádku matice $(\mathbf{A} - \mu_j\mathbf{I})$ (jsou to vlastně složky vlastního vektoru odpovídajícímu příslušnému vlastnímu čislu). Pro druhou řádku vypočteme subdeterminanty a dostaneme

$$\begin{aligned} \Delta_1(\mu_1) = \Delta_1(\sqrt{\alpha}) &= -\frac{1}{2} \\ \Delta_2(\mu_1) = \Delta_2(\sqrt{\alpha}) &= -\frac{1}{2} + \sqrt{\alpha} \end{aligned}$$

S ohledem na koncovou podmínku je $c_2 = 0$, potom

$$\begin{aligned} \omega(t) &= \omega_0 e^{-\sqrt{\alpha}t} \\ \lambda(t) &= \lambda_0 e^{-\sqrt{\alpha}t} \end{aligned}$$

Protože $c_1\Delta_1(\mu_1) = \omega_0$, $c_1\Delta_2(\mu_1) = \lambda_0$, pak platí $\lambda_0 = \omega_0\left(1 - 2\sqrt{\alpha}\right)$. Pak pro optimální řešení platí

$$\begin{aligned} \omega^*(t) &= \omega_0 e^{-\sqrt{\alpha}t} \\ \lambda^*(t) &= \omega_0\left(1 - 2\sqrt{\alpha}\right)e^{-\sqrt{\alpha}t} \\ u*(t) &= \left(1 - \sqrt{\alpha}\right)\omega^*(t) \end{aligned}$$

Optimální řízení je úměrné úhlové rychlosti. Doběh se děje s časovou konstantou $\tau = \frac{1}{\sqrt{\alpha}}$.

Pro $\alpha = 1$ je $u^*(t) = 0$, časová konstanta doběhu je rovna jedné. Brzdíme učinně tak, že obvod kotvy zkratujeme.

Pro $\alpha > 1$ je časová konstanta menší než jedna. Napětí zdroje $u(t)$ je záporné, proud je také záporný. Energie se ze zdroje napětí spotřebovává.

Pro $\alpha < 1$ je časová konstanta optimálního doběhu větší než jedna. Napětí zdroje $u(t)$ je kladné, ale proud je záporný. Energii v tomto případě rekuperujeme.

Pro $\alpha = 0$ je proud $i^*(t) = 0$, potom nebrzdíme a energeticky optimální doběh neexistuje.

Dosadíme-li optimální hodnoty veličin do kritéria kvality řízení, dostaneme

$$J = \int_0^\infty u(t)i(t)\ dt + \alpha \int_0^\infty \omega^2(t)\ dt = \frac{\omega_0^2}{2}\left(\sqrt{\alpha} - 1\right) + \alpha\left(\frac{\omega_0^2}{a\sqrt{\alpha}}\right), \qquad \alpha \neq 0$$

Z předchozího vztahu je názorně patrno, jak se změnou α mění jednotlivé členy v kombinovaném kritériu kvality řízení.

7.4.2 Řešení optimalizačního problému s omezením

Často máme omezenu oblast změn řídicích nebo stavových veličin. Potom na hranici oblasti omezení existují pouze jednostranné variace směřující dovnitř dovolené oblasti. Variační problém je v tomto případě tzv. **neklasického typu**.

Variační problémy s omezením se snáze řeší principem maxima. Zde si pouze ukážeme, jak lze omezení ve tvaru jednoduchých nerovnic převést vhodnou transformací na problémy bez omezení.

Máme-li řídicí veličiny omezeny podmínkami

$$g_i(\mathbf{x}(t), \mathbf{u}(t)) \leq 1, \qquad i = 1, \ldots, p \tag{7.80}$$

pak tento systém nerovností můžeme nahradit systémem rovnic

$$s_i(\mathbf{x}(t), \mathbf{u}(t), \mathbf{v}(t)) - g_i(\mathbf{x}(t), \mathbf{u}(t)) = 0, \qquad i = 1, \ldots, p, \tag{7.81}$$

kde funkce $s_i(\mathbf{x}(t), \mathbf{u}(t), \mathbf{v}(t))$ mají modul menší než jedna a jsou dvakráte diferencovatelné v otevřené oblasti argumentů. Je výhodné zvolit si za funkce s_i funkce trigonometrické.

$$\mathbf{s}(t) = [\sin v_1, \sin v_2, \ldots, \sin v_p] \tag{7.82}$$

Tímto způsobem můžeme omezení ve tvaru jednostranné i oboustranné nerovnosti nahradit rovností. Uvedeme si dva nejčastěji se vyskytující případy omezení kladená na řízení. Máme-li například řízení omezeno nerovností

$$|u_i(t)| \leq 1, \qquad i = 1, \ldots, r \tag{7.83}$$

pak toto omezení nahradíme rovností

$$\sin v_i(t) - u_i(t) = 0, \qquad i = 1, \ldots, r \tag{7.84}$$

kde $v_i(t)$ je nová proměnná, na kterou se nekladou žádná omezení. Jiné omezení ve tvaru oboustranné nerovnosti

$$\alpha(\mathbf{x}, \mathbf{u}) \leq g(\mathbf{x}, \mathbf{u}) \leq \beta(\mathbf{x}, \mathbf{u}) \tag{7.85}$$

nahradíme rovností ve tvaru

$$(\beta - \alpha) \sin v + (\beta - \alpha) - 2g(\mathbf{x}, \mathbf{u}) = 0 \tag{7.86}$$

Nelineární transformace (7.81), (7.84) a (7.86) zobrazují ohraničenou oblast variací řídicí funkce $u(t)$ na mnohalistou neohraničenou oblast změn jiné řídicí funkce $v(t)$.

Příklad 10: Časově optimální doběh stejnosměrného motoru.
Diferenciální rovnice motoru je stejná jako v předchozím příkladě 9. Kritérium optimality je doba regulačního pochodu

$$J = \int_0^T dt = T \tag{7.87}$$

Bez omezení řízení roste řídicí veličina nade všechny meze. V naší úloze je řídicí veličina omezena, nechť platí $|u(t)| \leq 1$. Toto omezení nahradíme rovností $\sin v(t) - u(t) = 0$.

Potom se jedná o Lagrangeův problém. Upravený funkcionál je

$$\bar{J} = \int_0^T \left(1 + \lambda_1(t)(\sin v(t) - u(t)) + \lambda_2(t)\left(\dot{\omega}(t) - u(t) + \omega(t)\right)\right) dt$$

Eulerovy - Lagrangeovy rovnice pro proměnné ω, u, v λ_1, λ_2 jsou

$$\begin{aligned}
\omega: &\quad \lambda_2(t) - \dot{\lambda}_2(t) &&= 0 \\
u: &\quad \lambda_1(t) - \lambda_2(t) &&= 0 \\
v: &\quad \lambda_1(t) \cos v(t) &&= 0 \\
\lambda_1: &\quad \sin v(t) - u(t) &&= 0 \\
\lambda_2: &\quad \dot{\omega}(t) - u(t) + \omega(t) &&= 0
\end{aligned}$$

Řešením prvních tří rovnic dostaneme $\lambda_1(t) = -\lambda_2(t) = \lambda_0 e^{-t}$,
$\lambda_1\sqrt{1 - \sin^2 v(t)} = \lambda_1\sqrt{1 - u^2(t)} = 0$. Protože $\lambda_1(t)$ je nenulová a nemění znaménko, je optimální řízení $u^*(t) = +1$ nebo -1 po celou dobu přechodového děje. Znaménko optimálního řízení je určeno znaménkem počáteční podmínky. Pro $\omega_0 > 0$ je $u(t) = -1$.

Pokud je systém, jádro funkcionálu i omezující podmínka lineární vzhledem k řízení $u(t)$, pak optimální řízení leží na hranici oblasti omezení.

7.5 Kanonický tvar Eulerovy - Lagrangeovy rovnice

Nyní se vrátíme k základnímu variačnímu problému minimalizace funkcionálu

$$J = \int_{t_0}^{t_1} g(\mathbf{x}, \dot{\mathbf{x}}, t)\ dt \tag{7.88}$$

Zavedeme si skalární funkci

$$H(\mathbf{x}, \dot{\mathbf{x}}, t) = -g(\mathbf{x}, \dot{\mathbf{x}}, t) + \left(\frac{\partial}{\partial \dot{\mathbf{x}}} g(\mathbf{x}, \dot{\mathbf{x}}, t) \right) \dot{\mathbf{x}} \tag{7.89}$$

a vektorovou funkci

$$p(\mathbf{x}, \dot{\mathbf{x}}, t) = \left(\frac{\partial g(\mathbf{x}, \dot{\mathbf{x}}, t)}{\partial \dot{\mathbf{x}}} \right)^T \tag{7.90}$$

Funkci H nazýváme **Hamiltonovou funkcí** nebo krátce **hamiltonián**. Vektor $\mathbf{p}$ se nazývá **konjugovaný vektor** k vektoru $\mathbf{x}$.

Z rovnice (7.90) lze vyjádřit $\dot{\mathbf{x}}$ jako funkci $\mathbf{p}$ a proto $H = H(\mathbf{x}, \mathbf{p}, t)$. Totální diferenciál hamiltoniánu H je

$$d\ H = \frac{\partial H}{\partial \mathbf{x}} d\mathbf{x} + \frac{\partial H}{\partial t} dt + \frac{\partial H}{\partial \mathbf{p}} d\mathbf{p} \tag{7.91}$$

Podle (7.89) však současně také platí

$$d\ H = -d\ g + \dot{\mathbf{x}}^T\ d\ \mathbf{p} + \mathbf{p}^T\ d\ \dot{\mathbf{x}} = -\frac{\partial g}{\partial \mathbf{x}} d\mathbf{x} - \frac{\partial g}{\partial t} dt + \dot{\mathbf{x}}^T d\mathbf{p} \tag{7.92}$$

Odtud porovnáním plyne

$$\frac{\partial H}{\partial \mathbf{x}} = -g_{\mathbf{x}}, \qquad \frac{\partial H}{\partial t} = -g_t, \qquad \frac{\partial H}{\partial \mathbf{p}} = \dot{\mathbf{x}}^T \tag{7.93}$$

Z (7.93) plynou Eulerovy - Lagrangeovy rovnice $g_{\mathbf{x}} - \frac{d}{dt} g_{\dot{\mathbf{x}}} = 0$ ve tvaru

$$\begin{aligned} \frac{d\mathbf{x}}{dt} &= \left(\frac{\partial H}{\partial \mathbf{p}} \right)^T \\ \frac{d\mathbf{p}}{dt} &= -\left(\frac{\partial H}{\partial \mathbf{x}} \right)^T \end{aligned} \tag{7.94}$$

Rovnice (7.94) se nazývají **Hamiltonovou** nebo **kanonickou formou** Eulerovy - Lagrangeovy rovnice. Soustavu Eulerových - Lagrangeových diferenciálních rovnic druhého řádu nahrazujeme soustavou dvou vektorových rovnic prvního řádu.

Poznámka: Předchozí rovnice se obvykle v literatuře udávají bez transpozice. My zde budeme důsledně uvažovat, že derivace skalární funkce podle vektoru je řádkový vektor, zatímco derivace vektoru (sloupcového) podle skaláru je opět sloupcový vektor. Proto, aby dimeze vektorů v předchozí soustavě souhlasily, je třeba je psát tak, jak je uvedeno.

□

Podél extremály platí

$$\frac{dH}{dt} = \frac{\partial H}{\partial t} \tag{7.95}$$

a jestliže jádro funkcionálu g a tím ani H nezávisí explicitně na čase t, pak $\frac{\partial H}{\partial t} = 0$ a hamiltonián je podél extremály konstantní.

Přitom platí

$$d\ J = H\ dt + (-\mathbf{p})^T d\mathbf{x} \tag{7.96}$$

Odtud plyne

$$H = \frac{\partial J}{\partial t}, \qquad \mathbf{p}^T = -\frac{\partial J}{\partial \mathbf{x}} \tag{7.97}$$

Obecné podmínky transverzality (7.28) ve volném koncovém bodě mají v tomto případě tvar

$$-H(\mathbf{x}, \mathbf{p}, t)\delta t + \mathbf{p}^T \delta \mathbf{x} = 0, \quad \text{pro} \;\; t = t_1 \tag{7.98}$$

Je-li t_0, t_1, $\mathbf{x}(t_0)$ pevné a pouze $\mathbf{x}(t_1)$ je volné, je podmínka transverzality zřejmě

$$\mathbf{p}(t_1) = 0. \tag{7.99}$$

Uvědomme si znovu, že kanonický tvar Eulerových - Lagrangeových rovnic opět převádí variační problém na okrajový problém řešení diferenciálních rovnic (7.94). Pro pevný koncový bod známe počáteční a koncové podmínky první diferenciální rovnice v (7.94). Pro druhou diferenciální rovnici v (7.94) nemáme žádné okrajové podmínky. Pokud je koncový bod volný, pak podle (7.99) známe koncovou podmínku pro druhou diferenciální rovnici v (7.94) (pro konjugovaný vektor $\mathbf{p}(t)$). V žádném případě nemůžeme řešit soustavu (7.94) v reálném čase.

Weierstrassova podmínka (7.40) je v kanonickém tvaru rovna

$$E(\mathbf{x}, \mathbf{z}, \mathbf{s}, t) \;\; = \;\; H(\mathbf{x}, \mathbf{s}, t) - H(\mathbf{x}, \mathbf{z}, t) \geq 0 \tag{7.100}$$

kde $\mathbf{s}(t)$ je derivace $\dot{\mathbf{x}}(t)$ na extremále a $\mathbf{z}(t)$ je libovolná jiná derivace funkce $\mathbf{x}(t)$. Z předchozí rovnice plyne, že na extremále nabývá Hamiltonova funkce svého maxima - viz kapitola o principu maxima.

Ověřte, že Weierstrassovy - Erdmannovy podmínky vyžadují spojitost funkce $\mathbf{p}$ a H v úhlovém bodě.

Uvažujme nyní znovu problém optimálního řízení

$$\min \left\{ J = \int_{t_0}^{t_1} g(\mathbf{x}, \mathbf{u}, t) \; dt \;\; : \;\; \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t); \;\; \mathbf{x}(t_0) = \mathbf{x}_0 \right\}. \tag{7.101}$$

Hamiltonova funkce H je opět podle (7.89), kde za jádro g dosadíme jádro ϕ rozšířeného funkcionálu. Připomeňme, že toto jádro bylo rovno - viz (7.68)

$$\phi(\mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}, t) = g(\mathbf{x}, \mathbf{u}, t) + \boldsymbol{\lambda}^T \left(\dot{\mathbf{x}} - \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \right). \tag{7.102}$$

Hamiltonova funkce je tedy

$$H = -\left(g + \boldsymbol{\lambda}^T (\dot{\mathbf{x}} - \mathbf{f}) \right) + \left(\frac{\partial \left(g + \boldsymbol{\lambda}^T (\dot{\mathbf{x}} - \mathbf{f}) \right)}{\partial \dot{\mathbf{x}}} \right) \dot{\mathbf{x}}.$$

Odtud po úpravách dostaneme

$$H\left(\mathbf{x}, \dot{\mathbf{x}}, \mathbf{u}, \boldsymbol{\lambda}, t\right) = -g(\mathbf{x}, \mathbf{u}, t) + \boldsymbol{\lambda}^T(t)\mathbf{f}(\mathbf{x}, \mathbf{u}, t) \tag{7.103}$$

kde $\boldsymbol{\lambda}(t)$ je konjugovaný vektor totožný s Lagrangeovým vektorem.

Hamiltonovy rovnice pro problém optimálního řízení (7.101) mají tvar

$$\begin{aligned} \frac{d\boldsymbol{\lambda}}{dt} &= -\left(\frac{\partial H}{\partial \mathbf{x}}\right)^T \\ 0 &= \frac{\partial H}{\partial \mathbf{u}} \\ \frac{d\mathbf{x}}{dt} &= \left(\frac{\partial H}{\partial \boldsymbol{\lambda}}\right)^T \end{aligned} \tag{7.104}$$

Předchozí soustava je totožná se soustavou variačních rovnic (7.70). O transpozicích v soustavě (7.104) viz předchozí poznámka.

Pro optimalizační problém Bolzova typu s hodnocením dosaženého cílového bodu pomocí funkce $h(\mathbf{x}(t_1))$ v kritériu, mají Hamiltonovy rovnice stejný tvar jako předchozí soustava (7.104).

7.6 Příklady

1. Rozeberte tvar a řešení Eulerovy - Lagrangeovy rovnice, je-li jádro funkcionálu
   a) nezávislé na $\dot{x}(t)$,
   b) lineárně závislé na $\dot{x}(t)$
   c) závislé pouze na $\dot{x}(t)$.

2. Řešte klasickou úlohu o brachystochroně:
   V rovině kolmé k zemskému povrchu máme dva body o souřadnicích (x_1, y_1), (x_2, y_2) (vodorovná vzdálenost, výška). Určete v této rovině křivku spojující oba body takovou, aby pohyb hmotného bodu po této křivce trval nejkratší dobu. Pohyb hmotného bodu začíná z počátečního bodu s větší výškovou souřadnicí a končí v druhém bodě, přičemž je hmotný bod vystaven pouze působení gravitačních sil. Všechny druhy tření zanedbáváme.
   Uvažujte rozšíření základní úlohy:
   a) Pohyb je brzděn, přičemž brzdicí účinek je úměrný kvadrátu okamžité rychlosti (odpor prostředí).
   b) Brzdicí účinek je nepřímo úměrný kvadrátu výšky (tím respektujeme vliv hustoty prostředí).
   c) Koncový bod není pevný, je určena pouze jeho
      1) vodorovná souřadnice
      2) výšková souřadnice.
   Ověřte, že řešením základní úlohy o brachystochroně jsou cykloidy určené parametricky

$$\begin{aligned} x &= \frac{c}{2}(t - \sin t) \\ y &= -\frac{c}{2}(1 - \cos t). \end{aligned}$$

   Řešením je funkce $y = y(x)$ a počáteční bod má souřadnice $(0, 0)$.

3. Určete křivku procházející danými body, která otáčením kolem osy souřadnic vytvoří plochu o nejmenším povrchu.
   Ověřte, že tato křivka (jmenuje se katenoida) je určena
   $$y = c_1 \cosh \frac{x - c_2}{c_1}$$

4. Určete minimální i maximální hodnotu $x(T)$ při omezení
   $$\begin{aligned} \frac{dx(t)}{dt} &= ax(t) + u(t); \quad x(0) = \alpha \\ \int_0^T u^2(t)\ dt &\leq K \end{aligned}$$

5. Určete minimální hodnotu kritéria s omezením
   $$\min \left\{ J = \int_0^1 (3x(t) + 2u(t))\ dt \ :\ \dot{x} = 3x + u, x(0) = 5, \frac{1}{2} \leq u(t) \leq 2 \right\}$$

6. Dynamický systém je popsán stavovými rovnicemi
   $$\dot{x}_1(t) = x_2(t), \quad \dot{x}_2(t) = u(t); \quad x_1(0) = \alpha, \quad x_2(0) = \beta$$
   Tento systém chceme převést z daného počátečního stavu $\mathbf{x}(0)$ do koncového stavu, který je co nejblíže počátku. Kritérium volíme
   $$J = a\ x_1^2(t_1) + \int_0^{t_1}\ dt$$
   Uvažujte nejprve úlohu bez omezení řízení a potom uvažujte omezení $|u(t)| \leq 1$.

7. Syntéza optimálního servomechanismu.
   Uvažujme servomotor s přenosem
   $$G(s) = \frac{Y(s)}{U(s)} = \frac{K}{s(Js + B)}$$
   kde K je direktivní konstanta motoru, J je moment setrvačnosti a B je konstanta tlumení. Napětí na kotvě servomotoru je $u(t)$ a $y(t)$ je poloha hřídele servomotoru. Počáteční poloha je $y(0) = 0$. Požadovaná koncová poloha je $w = \text{konst}$. Kritériem optimality je kvadratická plocha odchylky od požadované polohy
   $$J = \int_0^\infty (y(t) - w)^2\ dt.$$
   Aby problém měl reálné řešení, uvažujme omezení takové, že je omezena viskozní energie na jednotku výchylky. Platí tedy
   $$\int_0^\infty \frac{B\,(\dot{y}(0))^2}{w^2}\ dt = K$$
   Určete optimální $y^*(t)$ pro minimální kritérium při respektování předchozího izoperimetrického omezení. Určete Laplaceův obraz $y^*(t)$ a vypočtěte Laplaceův obraz $u^*(t)$. Odtud vypočtěte přenos optimálního zpětnovazebního regulátoru, který v regulačním obvodu zajistí optimální průběh všech veličin. Ukažte, že je to regulátor PD. Diskutujte izoperimetrickou podmínku.

8. Uvažujte servomotor se stejným přenosem jako v předchozím příkladě. Kritérium optimality je také stejné. Řízení je omezeno

$$\int_0^\infty u^2(t)\ dt = K$$

Počáteční podmínky jsou $y(0) = \dot{y}(0) = 0$ a koncové podmínky jsou zřejmě $y(\infty) = w$, $\dot{y}(\infty) = 0$.

9. Jaké jsou podmínky transverzality pro Mayerovu úlohu s volným koncem trajektorie?

Kapitola 8

Dynamické programování

Dynamické programování je velmi účinný nástroj k numerickému řešení problémů optimalizace. Používá se při řešení nejrůznějších problémů od problémů optimalizace k problémům umělé inteligence.

Mezi metodami optimalizace má metoda dynamického programování zvlášťní místo. Tato metoda je velmi přitažlivá díky jednoduchosti jejího základního principu - **principu optimality**. Princip optimality i celá metoda dynamického programování jsou spojené s pracemi amerického matematika R. Bellmana. Princip optimality představuje vlastně princip postupné analýzy problému. Vedle principu optimality je v metodě dynamického programování velmi důležitá myšlenka vnoření konkrétního optimalizačního problému do třídy analogických problémů. Tento princip se nazývá **princip invariantního vnoření**.

Další zvláštností této metody je tvar konečného výsledku. Výsledkem jsou rekurentní vztahy, které původní problém rozdělí na posloupnost řešení jednodušších problémů. Tyto rekurentní vztahy lze snadno řešit na počítači. Jediná potíž při aplikaci dynamického programování na řešení optimalizačních problémů je to, že s růstem počtu stavů procesu prudce rostou požadavky na operační paměť počítače. Tento jev označil R. Bellman jako **"prokletí rozměrnosti"** (curse of dimensionality). V aplikacích metody dynamického programování je tomuto jevu věnována značná pozornost.

Použitím metody dynamického programování nalezneme globální optimum a všechna optimální řešení.

8.1 Princip metody dynamického programování

8.1.1 Princip optimality a princip invariantního vnoření

Dynamické programování (DP), jak již název metody napovídá, využívá při řešení problému optimalizace jakousi "dynamičnost" problému. Problém optimalizace jako problém rozhodovací převedeme na mnohastupňový rozhodovací problém. Jediné rozhodnutí převedeme na posloupnost rozhodování a jediné řešení převedeme na posloupnost řešení jednodušších úloh.

Mnohastupňový rozhodovací proces je například proces diskrétního řízení systému. Jeho jednotlivé rozhodovací stupně jsou určeny volbou řízení v diskrétních časech. V

jiných případech musíme "mnohastupňovost" zavést do optimalizačního problému uměle.

Princip optimality tvrdí, že optimální posloupnost rozhodování v mnohastupňovém rozhodovacím procesu má tu vlastnost, že ať jsou jakékoliv vnitřní stavy procesu a předchozí rozhodování, zbylá rozhodování musí tvořit optimální posloupnost vycházející ze stavu, který je výsledkem předchozích rozhodování.

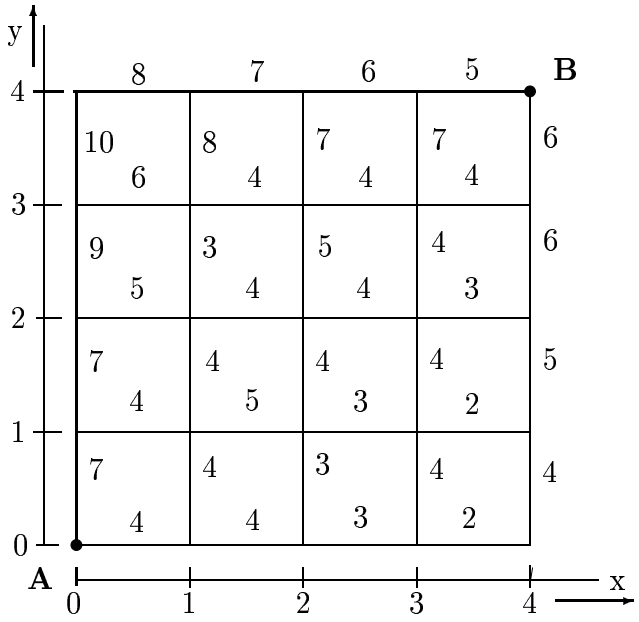
Jinými slovy, v každém rozhodovacím stupni musíme volit optimální rozhodnutí, přičemž vycházíme ze stavu, ve kterém se právě nacházíme. Princip optimality je v podstatě jinou formulací známého přísloví "Neplač nad rozlitým mlékem". Stav, ve kterém právě jsme, je výsledkem předchozích rozhodování a tento stav již nemůžeme ovlivnit, ale naše další rozhodování musí být optimální.

Princip invariantního vnoření znamená to, že náš optimalizační problém vnoříme do celé třídy analogických problémů. Tuto celou třídu problémů vyřešíme a tím také jaksi mimoděk vyřešíme náš jediný problém. Je zajímavé, že tento zdánlivě složitý postup je mnohdy velice efektivní.

8.1.2 Řešení jednoduché úlohy metodou DP

Základní principy metody dynamického programování si nejpřístupněji vysvětlíme na jednoduché úloze určení optimální cesty ve městě z jednoho místa na druhé. Jednoduchým modelem této úlohy je úloha na optimální průchod ve čtvercové síti z bodu A do bodu B - viz obr. 8.1.

Cesta z bodu A do B je možná pouze po úsečkách (ulice ve městě, hrany grafu na obr. 8.1). Překonání každé úsečky ohodnotíme kritériem či penálem. Může to být čas nebo spotřeba paliva potřebného k překonání části trajektorie, či stupeň znečistění příslušné ulice a podobně. Naším úkolem je nalézt takovou trajektorii z bodu A do B, aby součet

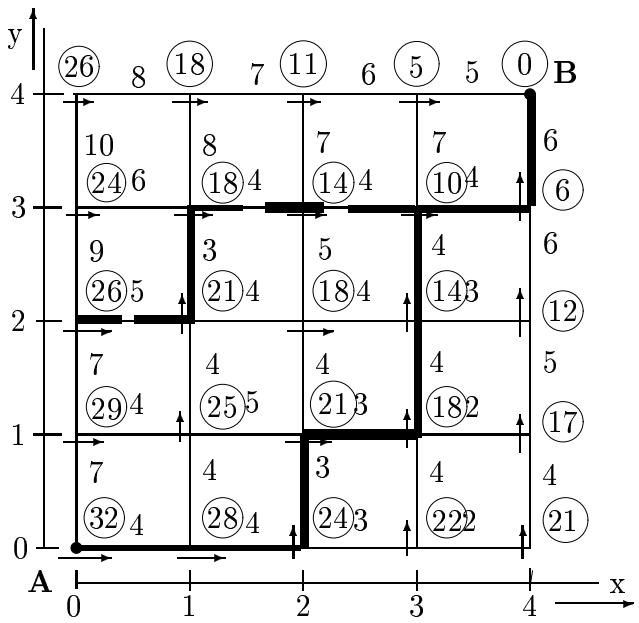


Obrázek 8.1: Optimální průchod sítí

dílčích kritérií po projitých úsečkách byl minimální. Uzlům sítě přiřadíme souřadnice (x, y) podle obr. 8.1, pak počáteční bod A má souřadnici $(0, 0)$ a koncový bod B má souřadnici $(4, 4)$. V úloze jsou souřadnice omezeny $0 \leq x \leq 4$ a $0 \leq y \leq 4$. Budeme dále předpokládat, že při pohybu sítí souřadnice uzlů neklesají. Pak v každém uzlu sítě se rozhodujeme, zda půjdeme nahoru či vpravo (podle obr. 8.1) - úsečky jsou tedy orientovány - graf na obr. 8.1 je tedy orientovaný graf.

Takovou jednoduchou úlohu můžeme řešit tak, že prozkoumáme všechny možné cesty z A do B a vybereme tu nejlepší. Počet všech cest ve čtvercové síti o maximální souřadnici n je při zvolených omezeních roven $\frac{(2n)!}{(n!)^2}$. V našem případě pro $n = 4$ máme 70 cest, ale pro $n = 20$ je celkový počet cest již 2.10^{12}. Při rozsáhlejší síti tuto metodu přímého výběru (optimalizace hrubou silou) již nelze použít.

Nyní použijeme metodu dynamického programování.
Náš problém nalezení optimální trajektorie z bodu A do B vnoříme do třídy úloh hledání optimální trajektorie z libovolného bodu o souřadnicích (x, y) do cíle v bodě B. Tím jsme si zdánlivě náš původní problém podstatně zkomplikovali. Dále uvidíme, že tato komplikace je opravdu pouze zdánlivá.



Obrázek 8.2: Optimální funkce a optimální rozhodování

Zavedeme si tak zvanou **optimální funkci** $V(x, y)$, která je rovna optimální hodnotě kritéria při přechodu po optimální trajektorii z bodu (x, y) do cíle B. Optimální funkce $V(x, y)$ se také nazývá **Bellmanova funkce**.

V každém bodu sítě se rozhodujeme, zda půjdeme nahoru či vpravo. Hodnotu kritéria při překonání úsečky mezi body (x, y) a $(x + 1, y)$, to je ve směru růstu souřadnice x si označíme jako $g_x(x, y)$. Podobně $g_y(x, y)$ je hodnota kritéria při překonání úsečky mezi body (x, y) a $(x, y + 1)$, to je ve směru růstu souřadnice y. Podle principu optimality

můžeme pro optimální funkci $V(x, y)$ napsat rekurentní vztah

$$V;(x,y) = \min \left[\begin{array}{l} g_x(x,y) + V(x+1,y) \\ g_y(x,y) + V(x,y+1) \end{array} \right], \quad 0 \le x \le n, \quad 0 \le y \le n. \tag{8.1}$$

Předchozí vztah plyne z toho, že z bodu (x, y) můžeme jít buď vpravo a potom zbytek cesty po optimální trajektorii do cíle (kritérium po zbytku cesty je $V(x + 1, y)$. Nebo můžeme jít z bodu (x, y) nahoru a opět zbytek cesty po optimální trajektorii. Nejlepší z těchto dvou možností je optimální trajektorie z bodu (x, y) do cíle.

Vztah (8.1) je rekurentní vztah pro výpočet optimální funkce $V(x, y)$. V koncovém bodě B jsme již v cíli a proto zřejmě platí

$$V(n, n) = V(4, 4) = 0 \tag{8.2}$$

Rekurentní vztah (8.1) řešíme "od zadu", vycházejíce z koncové podmínky (8.2). Pro síť z obr. 8.1 je na obr. 8.2 uvedena v každém uzlu sítě hodnota optimální funkce $V(x, y)$ (je zakreslena v kroužku). Ke každému uzlu sítě je šipkou zaznamenáno optimální rozhodnutí o další cestě. Z obr. 8.2 je zřejmé, že optimální hodnota kritéria z bodu A do B je rovna $V(0, 0) = 32$. Optimální trajektorie je na obr. 8.2 nakreslena silnou čarou.

Je zřejmé, že jsme tímto způsobem vyřešili i problém citlivosti optimální trajektorie na změnu rozhodnutí. Odchýlíme-li se z nějakého důvodu od optimální trajektorie, víme jak dále pokračovat optimálním způsobem. Optimální trajektorie z jiného počátečního bodu - bodu C na obr. 8.2 - je v tomto obrázku zakreslena silnou přerušovanou čarou.

Nyní budeme analyzovat **složitost předchozího algoritmu**.
Při výpočtu optimální funkce $V(x, y)$ podle (8.1) provádíme rozhodování (minimalizaci) pouze ve vnitřních bodech sítě. Těch je n^2, jsou to body o souřadnicích $0 \le x \le n - 1$, $0 \le y \le n - 1$. Podle (8.1) provádíme v každém rozhodovacím bodě pouze dvě sčítání. V krajních bodech sítě - to je v bodech o souřadnicích $x = n$, nebo $y = n$ - provádíme pouze jedno sčítání, neboť v těchto bodech je trajektorie určena jednoznačně. Celkový počet sčítání, který označíme S_{DP}, při použití metody dynamického programování je tedy roven

$$S_{DP}(n) = 2n^2 + 2n \tag{8.3}$$

Při přímém výběru je celkový počet cest $(2n)!/(n!)^2$ a na každé cestě je nutno provést $2n$ sčítání. Proto celkový počet sčítání, který označíme S_{PV}, je při přímém výběru

$$S_{PV}(n) = 2n\frac{(2n)!}{(n!)^2} \tag{8.4}$$

Pro náš případ $n = 4$ je $S_{DP}(4) = 40$ a $S_{PV}(4) = 560$, ale pro $n = 10$ je $S_{DP}(10) = 220$, ale $S_{PV}(10) = 272000$. Výpočetní složitost při použití metody dynamického programování roste kvadraticky, zatímco při přímém výběru roste exponenciálně.

Výpočet můžeme provádět tak, že si pamatujeme pouze optimální funkci $V(x, y)$ pro všechny body sítě a z ní určíme optimální rozhodování. Optimální rozhodování zjistíme v každém bodě sítě následující úvahou. Z bodu (x, y) se pohybujeme vodorovně, platí-li

$$\begin{array}{rcl} V(x,y) & = & a_x(x,y) + V(x+1,y) \\ V(x,y) & \le & a_y(x,y) + V(x,y+1) \end{array} \tag{8.5}$$

Obdobné vztahy platí pro pohyb vzhůru (znaménka v (8.5a) a (8.5b) se prohodí). Platí-li rovnost v (8.5a) a (8.5b), můžeme se z bodu (x, y) pohybovat vpravo nebo vzhůru. Vidíme, že výhoda metody dynamického programování je kromě jiného také v tom, že nalezneme absolutní minimum a všechna optimální řešení.

Výpočet můžeme provádět také tak, že si pamatujeme pouze hodnoty optimální funkce v bodech (x, y) a v bodech $(\bar{x}, \bar{y})$, kde $\bar{x} > x$ a $\bar{y} > y$ hodnotu optimální funkce můžeme zapomenout. V těchto bodech si ale pamatujeme optimální rozhodnutí. Po dosažení bodu A známe optimální rozhodnutí ve všech uzlech sítě a nemusíme je počítat podle (8.5).

Rovnice (8.1) se nazývá **Bellmanova rovnice**. Protože optimální funkci počítáme zpětně od cíle B, nazývá se (8.1) také zpětná Bellmanova rovnice.

Přímou Bellmanovu rovnici dostaneme tak, že naši úlohu o výběru optimální trajektorie z bodu A do bodu B vnoříme do třídy úloh optimalizace trajektorie z bodu A do libovolného bodu o souřadnicích (x, y). Optimální funkci v této úloze označíme $U(x, y)$. Pro ni platí rekurentní vztah

$$U(x,y) = \min \left[\begin{array}{l} g_x(x-1,y) + U(x-1,y) \\ g_y(x,y-1) + U(x,y-1) \end{array} \right], \quad 0 \le x \le n, \quad 0 \le y \le n. \tag{8.6}$$

s okrajovou podmínkou $U(0,0) = 0$. Vztah (8.6) se nazývá přímá Bellmanova rovnice. Zřejmě platí $U(n,n) = V(0,0)$. Tím je určena optimální hodnota kritéria z bodu A do bodu B.

8.2 Optimální řízení diskrétních systémů

8.2.1 Diskrétní úloha optimalizace

Při řešení problému optimálního řízení dynamických systémů je nejjednodušší použít metodu dynamického programování při diskrétním modelu situace. Mějme tedy stavové rovnice diskrétního dynamického systému

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k), k) \\ \mathbf{y}(k) &= \bar{\mathbf{f}}(\mathbf{x}(k), \mathbf{u}(k), k) \end{aligned} \tag{8.7}$$

kde $\mathbf{x}$, $\mathbf{u}$, $\mathbf{y}$ je stav, vstup a výstup systému a $k = \{\ldots, 0, 1, \ldots\}$ je diskrétní čas. Řízení $\mathbf{u}(k)$ i stav $\mathbf{x}(k)$ jsou vždy nějakým způsobem omezeny, pak

$$\mathbf{u}(k) \in \mathbf{U} \subset R^r, \qquad \mathbf{x}(k) \in \mathbf{X} \subset R^n \tag{8.8}$$

kde $\mathbf{U}$ a $\mathbf{X}$ jsou množiny přípustných řízení a stavů (obecně mohou záviset na diskrétním čase k). Volbou řízení $\mathbf{u}(k_0)$ až $\mathbf{u}(k_1-1)$ dostaneme ze stavových rovnic (8.7) a počátečního stavu $\mathbf{x}(k_0)$ posloupnost stavů $\mathbf{x}(k)$, kde $k \in [k_0, k_1]$ a k_0 je počáteční a k_1 je koncový stav.

Abychom mohli vybrat optimální řešení úlohy řízení je třeba opět zvolit kritérium kvality řízení, které umožní porovnávat různé varianty řešení. Obecný tvar kritéria kvality řízení je

$$J\left(k_0, k_1, \mathbf{x}(k_0), \mathbf{x}(k_1), \mathbf{u}(k_0), \ldots, \mathbf{u}(k_1-1)\right) = h\left(\mathbf{x}(k_1)\right) + \sum_{k=k_0}^{k_1-1} g\left(\mathbf{x}(k), \mathbf{u}(k), k\right) \tag{8.9}$$

První člen hodnotí dosažený cíl trajektorie a druhý - sumační - člen hodnotí průběh trajektorie - způsob dosažení cíle.

Problém optimálního řízení spočívá v nalezení takového řízení $\mathbf{u}^*(k)$ systému (8.7) na intervalu $k \in [k_0, k_1 - 1]$, aby byla splněna omezení (8.8) a kritérium (8.9) bylo minimální.

Uvědomme si nejprve, že problém diskrétního optimálního řízení na konečném intervalu je konečněrozměrný problém. Tím se zásadně liší od spojitých problémů optimalizace. Formálně je problém diskrétního optimálního řízení podobný spojitému problému řízení - je to úloha Bolzova typu.

Optimální řízení závisí na počátečním čase k_0 a počátečním stavu $\mathbf{x}(k_0) = \mathbf{x}_0$. Koncový bod trajektorie může být libovolný (penalizační faktor $h(\mathbf{x}(k_1))$ zajistí pouze přibližné dosažení cíle). V problému s volným koncem může být koncový stav určen cílovou množinou C, $\mathbf{x}(t_1) \in C$ - pak se jedná o problém s pohyblivým koncem. Je-li cíl C bod, jedná se o problém s pevným koncem a první člen v kritériu není třeba uvažovat, neboť je konstantní.

Obecný tvar kritéria (8.9) můžeme upravit a dostat úlohu Mayerova či Lagrangeova typu. Zavedením další stavové souřadnice

$$x_{n+1}(k) = \sum_{i=k_0}^{k_1-1} g\left(\mathbf{x}(i), \mathbf{u}(i), i\right)$$

která zřejmě vyhovuje diferenční rovnici

$$x_{n+1}(k+1) = x_{n+1}(k) + g\left(\mathbf{x}(k), \mathbf{u}(k), k\right), \qquad x_{n+1}(k_0) = 0, \tag{8.10}$$

můžeme krérium (8.9) zapsat ve tvaru

$$J = h\left(\mathbf{x}(k_1)\right) + x_{n+1}(k_1) = \bar{h}\left(\bar{\mathbf{x}}(k_1)\right), \tag{8.11}$$

kde $\bar{\mathbf{x}} = [\mathbf{x}^T, x_{n+1}]^T$ je rozšířený stav. Rozšířením stavu o jednu složku, pro kterou platí stavová rovnice (8.10), dostaneme problém Mayerova typu s kritériem (8.11).

Obráceně zavedeme funkci

$$\begin{aligned} g_1\left(\mathbf{x}(k), \mathbf{u}(k), k\right) &= h\left(\mathbf{x}(k+1)\right) - h\left(\mathbf{x}(k)\right) \\ &= h\left(\mathbf{f}\left(\mathbf{x}(k), \mathbf{u}(k), k\right)\right) - h\left(\mathbf{x}(k)\right) \end{aligned}$$

Potom zřejmě

$$\sum_{k=k_0}^{k_1-1} g_1\left(\mathbf{x}(k), \mathbf{u}(k), k\right) = h\left(\mathbf{x}(k_1)\right) - h\left(\mathbf{x}(k_0)\right)$$

Protože $h\left(\mathbf{x}(k_0)\right)$ nezávisí na řízení, nemusíme tento člen v kritériu uvažovat a potom kritérium (8.9) lze zapsat ve tvaru

$$\sum_{k=k_0}^{k_1-1} \left(g\left(\mathbf{x}(k), \mathbf{u}(k), k\right) + g_1\left(\mathbf{x}(k), \mathbf{u}(k), k\right)\right) \tag{8.12}$$

Popsaným postupem jsme optimalizační problém Bolzova typu převedli na problém Lagrangeova typu s kritériem (8.12).

8.2.2 Převod spojitého optimalizačního problému na diskrétní

Při numerických výpočtech na číslicovém počítači je nezbytné převést spojitý optimalizační problém na diskrétní.

Můžeme samozřejmě diferenciální stavové rovnice spojitého systému řešit některou spolehlivou numerickou metodou. Také spojité kritérium můžeme počítat numericky. Problém je pouze v diskretizaci řídicí veličiny $\mathbf{u}(t)$.

Diskretizaci provedeme volbou periody vzorkování T a potom spojitý čas t převedeme na diskrétní čas k podle vztahu

$$t = kT \tag{8.13}$$

Volba periody vzorkování T závisí především na dynamických vlastnostech systému, celkové době řízení $(t_1 - t_0)$ a výpočetních možnostech.

Řízení spojitého systému můžeme předpokládat konstantní během periody vzorkování, pak

$$u(t) = u(k), \quad \text{pro} \quad kT \leq t < (k+1)T, \tag{8.14}$$

Případně můžeme řízení aproximovat složitější funkcí (například lineární nebo kvadratickou funkcí či aproximovat spliny).

Použijeme-li nejjednodušší Eulerovu metodu integrace diferenciálních rovnic systému i kritéria, pak spojitý optimalizační problém

$$\min_{\mathbf{u}(t)} \left\{ J = h(\mathbf{x}(t_1)) + \int_{t_0}^{t_1} g\left(\mathbf{x}, \mathbf{u}, t\right)\ dt, \quad \dot{\mathbf{x}}(t) = \mathbf{f}\left(\mathbf{x}, \mathbf{u}, t\right) \right\} \tag{8.15}$$

aproximujeme diskrétním optimalizačním problémem

$$\min_{\mathbf{u}(k)} \left\{ J = h(\mathbf{x}(k_1)) + T \sum_{k=k_0}^{k_1-1} g\left(\mathbf{x}, \mathbf{u}, k\right), \quad \mathbf{x}(k+1) = \mathbf{x}(k) + T\mathbf{f}\left(\mathbf{x}, \mathbf{u}, k\right) \right\} \tag{8.16}$$

kde spojitý čas t a diskrétní čas k jsou dle (8.14).

8.2.3 Převod diskrétního optimalizačního problému na úlohu matematického programování

Protože jsou vypracovány spolehlivé numerické algortmy řešení úloh matematického programování, je velmi výhodné převést i problém optimálního řízení diskrétního dynamického systému na tuto úlohu.

Uvědomme si, že v tomto případě se nejedná o žádnou aproximaci. Pro konečnou dobu diskrétního řízení $(k_1 - k_0)$ hledáme konečný počet optimálních hodnot řízení $\mathbf{u}^*(k_0)$ až $\mathbf{u}^*(k_1 - 1)$.

Také spojitý optimalizační problém převádíme často na úlohu matematického programování. V tomto případě se vždy jedná o aproximaci, neboť nejprve je třeba aproximovat spojitý problém problémem diskrétním a ten převést bez aproximace na úlohu matematického programování.

Nejjednodušší převod diskrétního optimalizačního problému je zavedení složeného vektoru $\mathbf{z}$ tvořeného řídicími veličinami

$$\mathbf{z} = \left[\mathbf{u}^T(k_0), \mathbf{u}^T(k_0+1), \ldots, \mathbf{u}^T(k_1-1)\right]^T \tag{8.17}$$

Řešením stavových rovnic diskrétního systému vypočteme stavy diskrétního systému a pak hodnotu kritéria $J(\mathbf{z})$, které závisí na řízení a tím na vektoru $\mathbf{z}$. Diskrétní problém optimálního řízení je úlohou matematického programování

$$\min_{\mathbf{Z}} \{J(\mathbf{z}) \ : \ \mathbf{z} \in \mathbf{Z}\} \tag{8.18}$$

kde množina $\mathbf{Z}$ je dovolená množina řízení. V této formulaci nemůžeme respektovat omezení kladená na stavové veličiny.

Abychom mohli respektovat stavová omezení, je třeba zahrnout stavové veličiny do rozšířeného vektoru $\mathbf{z}$, pak

$$\mathbf{z} = \left[\mathbf{x}^T(k_0), \mathbf{u}^T(k_0), \mathbf{x}^T(k_0+1), \ldots, \mathbf{x}^T(k_1-1), \mathbf{u}^T(k_1-1), \mathbf{x}^T(k_1)\right]^T \tag{8.19}$$

Omezení daná stavovými rovnicemi respektujeme vektorovou funkcí

$$\mathbf{r}(\mathbf{z}) = \begin{bmatrix} \mathbf{x}(k_0+1) - \mathbf{f}\left(\mathbf{x}(k_0), \mathbf{u}(k_0), k_0\right) \\ \vdots \\ \mathbf{x}(k_1) - \mathbf{f}\left(\mathbf{x}(k_1-1), \mathbf{u}(k_1-1), k_1-1\right) \end{bmatrix} \tag{8.20}$$

Kritérium (8.9) je opět závislé na vektoru $\mathbf{z}$. Problém optimálního řízení diskrétního systému je ekvivalentní úloze matematického programování ve tvaru

$$\min_{\mathbf{Z}} \{J(\mathbf{z}) \ : \ \mathbf{r}(\mathbf{z}) = 0, \ \mathbf{z} \in \mathbf{Z}\} \tag{8.21}$$

kde množina $\mathbf{Z}$ respektuje omezení kladená na stavy i řízení. Problémy s pevným koncovým stavem můžeme pomocí penalizačních metod převést na problémy bez omezení na koncový stav.

Problémy optimalizace dynamických systémů se v současnosti nejčastěji řeší tímto způsobem.

8.2.4 Řešení problému diskrétního optimálního řízení pomocí DP

Pro diskrétní dynamický systém (8.7) s počátečním stavem $\mathbf{x}(k_0) = \mathbf{x}_0$, hledáme takové řízení $\mathbf{u}(k)$ na intervalu $k \in [k_0, k_1 - 1]$, aby bylo minimální kritérium (8.9), při splnění všech omezení na stavy a řízení.

Tuto jedinou úlohu vnoříme do třídy úloh určení optimálního řízení stejného dynamického systému (8.7), kde ale uvolníme počáteční čas (který označíme i) i počáteční stav (který označíme $\mathbf{s}$). Kritérium optimality je potom

$$J\left(i, \mathbf{s}, \mathbf{u}(k_0), \ldots, \mathbf{u}(k_1-1)\right) = h\left(\mathbf{x}(k_1)\right) + \sum_{k=i}^{k_1-1} g\left(\mathbf{x}(k), \mathbf{u}(k), k\right) \tag{8.22}$$

Koncový čas t_1 je pevný. Vyřešením této třídy úloh vyřešíme pro $i = k_0$ a $\mathbf{s} = \mathbf{x}_0$ také naši původní úlohu.

Pro řešení uvedené třídy optimalizačních úloh si zavedeme optimální funkci $V(\mathbf{s}, i)$

$$V(\mathbf{s}, i) = \min_{\mathbf{u}(i),\ldots\mathbf{u}(k_1-1)} J\left(i, \mathbf{s}, \mathbf{u}(k_0), \ldots, \mathbf{u}(k_1-1)\right) \tag{8.23}$$

Jednoduchou úpravou předchozího vztahu dostaneme

$$V(\mathbf{s}, i) = \min_{\mathbf{u}(i)} \left\{ g\left(\mathbf{s}, \mathbf{u}(i), i\right) + \min_{\mathbf{u}(i+1),\ldots} \left[h\left(\mathbf{x}(k_1)\right) + \sum_{k=i+1}^{k_1-1} g\left(\mathbf{x}(k), \mathbf{u}(k), k\right) \right] \right\} \tag{8.24}$$

Druhý člen na pravé straně předchozí rovnice je roven
$V(\mathbf{s}(i+1), i+1) = V\left(\mathbf{f}\left(\mathbf{s}, \mathbf{u}(i), i\right), i+1\right)$. Proto pro optimální funkci platí funkcionální rekurentní předpis

$$V(\mathbf{s}, i) = \min_{\mathbf{u}(i)} \left\{ g\left(\mathbf{s}, \mathbf{u}(i), i\right) + V\left(\mathbf{f}\left(\mathbf{s}, \mathbf{u}(i), i\right), i+1\right) \right\} \tag{8.25}$$

Okrajová podmínka pro funkcionálně rekurentní předpis (8.25) je

$$V(\mathbf{s}, k_1) = h\left(\mathbf{s}(k_1)\right) \tag{8.26}$$

kde h je hodnocení cíle v kritériu (8.9). Vztah (8.25) se nazývá Bellmanova rovnice.

Výpočet optimální funkce je principiálně jednoduchý. Podle (8.26) určíme pro všechna $\mathbf{s} = \mathbf{x}(k_1) \in \mathbf{X}$ optimální funkci $V(\mathbf{s}, k_1)$. Potom v (8.25) položíme $i = k_1 - 1$ a pro všechna $\mathbf{s} = \mathbf{x}(k_1 - 1) \in \mathbf{X}$ určíme z (8.25) optimální funkci $V(\mathbf{s}, k_1 - 1)$. Tak pokračujeme pro $k = k_1 - 2$ až do $k = k_0$. Předpis (8.25) je úloha jednokrokové optimalizace a je obvykle mnohem jednodušší než původní úloha.

Vztah (8.24) jsme dostali využitím aditivních vlastností kritéria. Optimální trajektorii od diskrétního času $k = i$ do konce $k = k_1$ jsme hledali mezi trajektoriemi, které jsou libovolné v prvním kroku pro $k = i$ a v dalších krocích jsou již optimální. Při tom vycházíme ze stavu, do kterého jsme se dostali vlivem předchozího řízení. V (8.24) jsme využili princip optimality. Mezi všemi těmito trajektoriemi je i optimální trajektorie.

Princip optimality jsme zde využili v poněkud jiném tvaru, než byl dříve vysloven. Původní formulace principu optimality byla vyslovena ve tvaru nutné podmínky optimality. Zde jsme použili princip optimality ve tvaru postačující podmínky.
V této úpravě **princip optimality** zní:
Interval řízení rozdělíme na dva subintervaly. Jestliže je řízení na druhém intervalu optimální vzhledem ke stavu vzniklém jako výsledek v prvním intervalu řízení a řízení v prvním intervalu je optimální, pak je řízení optimální na celém intervalu.

Vypočteme-li optimální funkci $V(\mathbf{s}, i)$ pro všechna $i \in [k_0, k_1]$ a $\mathbf{s} \in \mathbf{X}$, snadno určíme optimální řízení $\mathbf{u}^*(k)$, $k \in [k_0, k_1 - 1]$. Pro optimální řízení $\mathbf{u}^*(k)$ platí

$$g\left(\mathbf{x}(k), \mathbf{u}^*(k), k\right) = V\left(\mathbf{x}(k), k\right) - V\left(\mathbf{f}\left(\mathbf{x}(k), \mathbf{u}^*(k), k\right), (k+1)\right) \tag{8.27}$$

Pro neoptimální řízení platí v (8.27) místo rovnosti nerovnost $\geq$. Vztah (8.27) přímo plyne z (8.24).

Dovolenou množinu stavů je třeba diskretizovat a optimální funkci $V(\mathbf{s}, i)$ počítáme pouze v těchto diskrétních bodech sítě stavů. Již z této jednoduché úvahy je zřejmé, že nároky na paměť při numerickém výpočtu jsou značné. Bellmanem označené "prokletí rozměrnosti" nám často znemožní řešení složitějšího problému.

Pro osvětlení našich obecných úvah si vyřešíme následující jednoduchý příklad, ve kterém vhodnou diskretizací nejsou obtíže vznikající při reálných problémech.

Příklad 2: Mějme jednoduchý diskrétní systém prvního řádu se stavovou rovnicí

$$x(k+1) = x(k) + u(k), \qquad x(0) = 5$$

Kritérium jakosti řízení je

$$J(u(k)) = 2,5\,(x(10) - 2)^2 + \sum_{0}^{9} x^2(k) + u^2(k)$$

Stavy a řízení jsou omezeny

$$0 \leq x(k) \leq 8, \qquad -2 \leq u(k) \leq 2$$

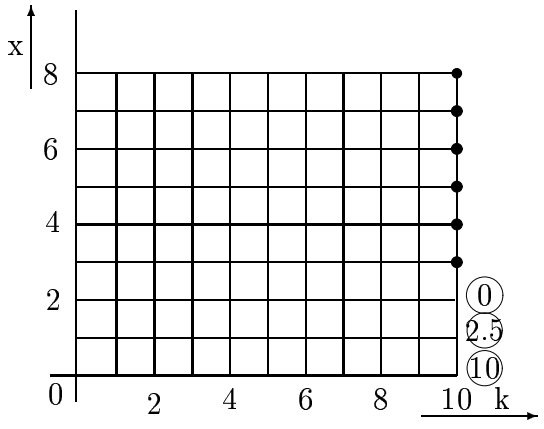
Cílová množina je

$$0 \leq x(10) \leq 2$$

Hledáme optimální řízení $u^*(k)$, které minimalizuje kritérium, respektuje omezení a které převádí počáteční stav $x(0) = 5$ do určeného cíle.

Abychom tuto úlohu mohli numericky řešit, budeme diskretizovat množinu stavů i řízení

$$x = \{0, 1, 2, \ldots, 8\}, \qquad u = \{-2, -1, 0, 1, 2\}$$



Obrázek 8.3: Optimální funkce a optimální rozhodování

Vhodná diskretizace nás zbaví řady problémů, které budou diskutovány později. Síť bodů, ve kterých počítáme optimální funkci $V(x, i)$, je na obr.8.3. Optimální funkci $V(x, i)$ počítáme "od zadu" z koncového času $k = 10$. Podle kritéria platí

$$V(x, 10) = 2,5\,(x - 2)^2$$

Tabulka 8.1: Výpočet optimální funkce $V(x,i) = V(0,9)$

$u(9)$	$g(x,u,i) = x^2 + u^2$	$x(10) = 0 + u(9)$	$V(x(10),10)$	$V + g$
-2	*	-2	*	*
-2	*	-1	*	*
0	0	0	10	10
1	1	1	2,5	3,5
2	4	2	0	4

Tabulka 8.2: Výpočet optimální funkce $V(x,i) = V(1,9)$

$u(9)$	$g(x,u,i) = x^2 + u^2$	$x(10) = 1 + u(9)$	$V(x(10),10)$	$V + g$
-2	*	-1	*	*
-2	2	0	10	12
0	1	1	2,5	3,5
1	2	2	0	2
2	*	3	*	*

a proto $V(0,10) = 10$, $V(1,10) = 2,5$, $v(2,10) = 0$. Bellmanova rovnice pro naši úlohu je

$$V(x,i) = \min_{u(i)\in U} \left\{ x^2 + u^2(i) + V\left((x + u(i)), (i+1)\right) \right\} \tag{8.28}$$

s okrajovou podmínkou $V(x,10)$, kterou jsme již určili.

Pro $i = 9$ vypočteme optimální funkci pro stav $x(9) = 0$ podle následující tabulky.

V posledním sloupci předchozí tabulky jsou vypočteny hodnoty výrazu $\{0 + u^2(9) + V\left((0 + u(9)), 10\right)\}$. Podle (8.28) hledáme minimum tohoto výrazu. V posledním sloupci předchozí tabulky je minimální prvek $3,5$ a proto $V(0,9) = 3,5$ a tomu odpovídá optimální řízení $u^*(x,i) = u^*(0,9) = 1$. Stavy $x(10) = -2$ a $x(10) = -1$ nesplňují omezení a proto výpočet pro ně neprovádíme (značíme je hvězdičkou).

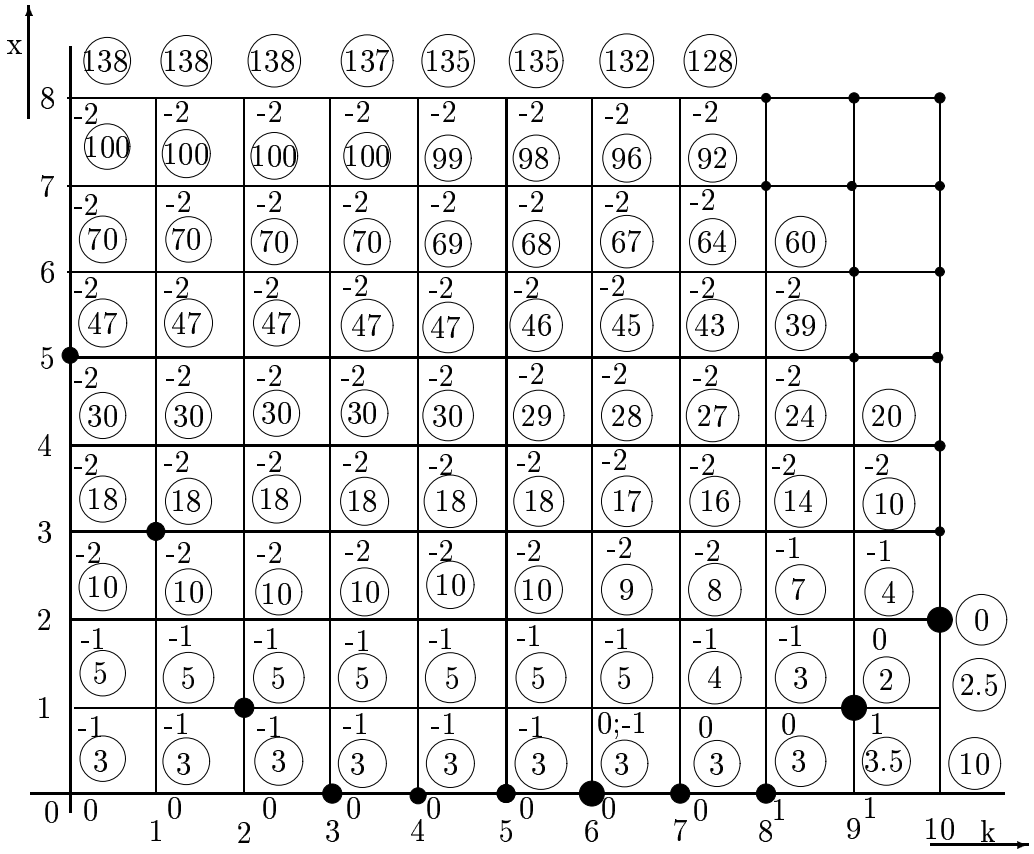
Podobně pro čas $i = 9$ a stav $x(9) = 1$ vypočteme optimální funkci $V(1,9)$ a optimální řízení $u^*(x,i) = u^*(1,9)$ podle následující tabulky: Optimální funkce je určena minimem v posledním sloupci, pak $V(1,9) = 2$ a optimální řízení je $u^*(x,i) = u^*(1,9) = 1$.

Podobně vypočteme stejnou tabulku pro $i = 9$ a $x(9) = 2$ až $x(9) = 8$. Stejně postupujeme pro čas $i = 8$ atd. až dospějeme k $i = 0$. Pro $i = 0$ a $x(0) = 5$ vyřešíme naši původní úlohu.

Ke každému bodu sítě stavů vypočteme optimální funkci $V(x,i)$ - viz obr. 8.4. V obr. 8.4 je hodnota optimální funkce uvedena u každého bodu sítě v kroužku. Optimální řízení $u^*(x,i)$ je uvedeno na obr. 8.4 v každém bodě sítě pod hodnotou optimální funkce. Na obr. 8.4 je silnou čarou zakreslena spojnice bodů optimální trajektorie z počátečního stavu $x(0) = 5$ do cíle. Našli jsme globální optimum a to je jediné. Optimální řídicí posloupnost je rovna

$$u^*(k) = \{-2,\ -2,\ -1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 1\}$$

Optimální trajektorie z počátečního stavu $x(3) = 6$ je na obr. 8.4 nakreslena čárkovaně. Je zřejmé, že v tomto případě existují dvě optimální trajektorie, neboť ve stavu $x^*(6) = 1$ lze volit optimální řízení $u^*(6) = 0$ nebo $u^*(6) = -1$ a hodnota kritéria je v obou případech stejná.



Obrázek 8.4: Optimální funkce a optimální řízení z příkladu 2.

Pokud bychom zvolili diskretizaci množiny dovolených hodnot řízení jiným způsobem, vznikly by obtíže, běžné při řešení většiny reálných problémů. Zvolme tedy množinu možných hodnot řízení

$$u = \{-2;\ -1;\ 0;\ 0,5;\ 1;\ 1,5;\ 2\}$$

Vypočtěme nyní pro $s = x(9) = 0$ optimální funkci. Řízení $u = -2$ a $u = -1$ neuvažujeme, neboť v bodě $x(9) = 0$ nejsou přípustná. Sestavíme opět tabulku (tabulka 8.3). Z tabulky je zřejmé, že optimální funkci $V(x(10), 10)$ jsme v bodě $x(10) = 0,5$ v předchozím kroku neurčili. Hodnota $x(10) = 0,5$ nám totiž padla mimo zvolenou síť . Abychom určili hodnotu $V(0,5,\ 10)$, která je v předchozí tabulce označena α a podobně $V(1,5,\ 10)$, musíme

Tabulka 8.3: Jiná diskretizace přípustných řízení, Výpočet $V(x, i) = V(0, 9)$

$u(9)$	$g(x, u, i) = x^2 + u^2$	$x(10) = 0 + u(9)$	$V(x(10), 10)$	$V + g$
0	0	0	10	10
0,5	0,25	0,25	α	0,25 + α
1	1	1	2,5	3,5
1,5	2,25	1,5	β	2,25 + β
2	4	2	0	0

provést interpolaci mezi sousedními hodnotami optimální funkce. Potom platí

$$\begin{aligned} V(0,5,\ 10) &= \frac{1}{2}\left[V(0,\ 10) + v(1,\ 10)\right] = 6,25 = \alpha \\ V(1,5,\ 10) &= \frac{1}{2}\left[V(1,\ 10) + v(2,\ 10)\right] = 1,25 = \beta \end{aligned}$$

Je zřejmé, že při větší dimenzi stavového prostoru je interpolace obtížná. Uvědomme si zároveň, že jakékoliv omezení na stavy či řízení lze při numerickém řešení dobře respektovat a každé omezení zjednodušuje řešení. To je podstatný rozdíl proti variačním metodám, kde se omezení respektují velmi obtížně.

8.2.5 Řešení některých speciálních úloh dynamickým programováním

Princip optimality a princip invariantního vnoření jsou obecné principy metody dynamického programování a mohou být využity při řešení různých optimalizačních problémů. Ukážeme na následujících třech příkladech řešení různých úloh statické optimalizace.

Volba optimálního nákladu

Máme n typů zboží a chceme naložit přepravní jednotku o dovoleném zatížení z_{max} takovým nákladem, jehož cena je nejvyšší. Podobné problémy vznikají při volbě optimálního nákladu lodi, kontejnerů a pod. tak, aby přepravní zisk byl maximální.

Označme: v_i - cena jednoho předmětu i-tého typu, w_i - váha jednoho předmětu i-tého typu, x_i - počet předmětů i-tého typu.

Hledáme tedy maximum lineární formy

$$J_n = \sum_{i=1}^{n} x_i v_i$$

při omezujících podmínkách

$$\sum_{i=1}^{n} x_i w_i \leq z_{max}, \qquad x_i = 0, 1, 2, \ldots$$

Jedná se tedy o lineární problém, ve kterém proměnné x_i jsou přirozená čísla.

Pokud bychom netrvali na celočíselnosti proměnných x_i, pak bychom za náklad vybrali to zboží, jehož cena na jednotku váhy je nejvyšší. Vybrali bychom tedy zboží typu k, pro které platí

$$k = \arg \max_{i=1,\ldots,n} \frac{v_i}{w_i}$$

a maximální cena nákladu je $J^* = \frac{v_k}{w_k} z_{max}$. Při požadavku celočíselnosti se optimální řešení může od předchozího značně lišit.

Úlohu budeme řešit pomocí dynamického programování tak, že ji vnoříme do celé třídy úloh - počet druhů zboží je postupně $j = 1,\ 2$ až n a dovolená zátěž je z, $0 \leq z \leq z_{max}$.

Hledáme tedy maximum

$$J_j(x,z) = \sum_{i=1}^{j} x_i v_i, j = 1, 2, \ldots, n$$

při omezení

$$\sum_{i=1}^{j} x_i w_i \leq z_{max}, \quad 0 \leq z \leq z_{max}, \quad x_i = 0, 1, 2, \ldots$$

Zavedeme si opět optimální funkci

$$V_j(z) = \max_{x_i} J_j(x,z),$$

která je rovna maximální ceně nákladu o váze z, složeného z j druhů zboží.

Rekurentní vztah pro výpočet optimální funkce $V_j(z)$ odvodíme následující úpravou předchozích vztahů. Optimální funkci můžeme vyjádřit ve tvaru

$$V_j(z) = \max_{x_j} \left(x_j v_j + \max_{x_1,\ldots,x_{j-1}} \sum_{i=1}^{j-1} x_i v_i \right)$$

Omezení na dovolenou nosnost z upravíme do tvaru

$$\sum_{i=1}^{j-1} x_i w_i \leq z - x_j w_j.$$

Předchozí vztahy vedou na Bellmanovu rovnici

$$V_j(z) = \max_{x_j} \left(x_j v_j + V_{j-1}\left(z - x_j w_j\right) \right)$$

s okrajovou podmínkou $V_0(z) = 0$. Bellmanovu rovnici řešíme postupně pro $j = 1$, 2 až n. Maximalizaci hledáme pro přirozená x_j, která se mění v mezích $x_j \in [0, \left[\frac{z_{max}}{w_i}\right]]$, kde $[z_{max}/w_i]$ značí nejvyšší přirozené číslo menší nebo rovné z_{max}/w_i. Pro $j = n$ a $z = z_{max}$ vyřešíme tímto postupem naši původní úlohu a při tom máme k dispozici všechna řešení při jiných omezení.

Nejkratší cesta sítí

Mějme n bodů očíslovaných 1, 2 až n. Nechť čas potřebný k překonání vzdálenosti z bodu i do j je roven t_{ij}. Obecně může platit $t_{ij} \neq t_{ji}$. Předpokládejme, že $t_{ij} \geq 0$. Není-li přechod z bodu i do j možný, pak $t_{ij} \to \infty$ a pro úplnost zřejmě $t_{ii} = 0$. Všechny body jsou tedy navzájem spojeny orientovanými spojnicemi, které jsou ohodnoceny veličinou t_{ij}. Body a spojovací cesty mezi nimi tvoří síť - orientovaný graf. Naším úkolem je určit takovou cestu sítí spojující dva dané body, pro jejíž překonání je potřeba nejmenší čas.

Tento problém vzniká při letecké, lodní i automobilové dopravě, při předávání zpráv ve sdělovacích sítích a pod.. Úloha je totožná s problémem časově optimálního řízení, interpretujeme-li uzly grafu jako stavy systému a větve grafu jako transformace z jednoho stavu do stavu druhého. Veličina t_{ij} je čas přechodu ze stavu i do j. Zřejmě t_{ij} může být

i jiné ohodnocení přechodu ze stavu i do j (např. spotřeba energie). Uzly grafu - body 1, 2 až n - jsou tedy stavy procesu, počáteční stav je stav p a koncový stav je stav m.

Budeme tuto úlohu opět řešit dynamickým programováním. Naši jedinou úlohu - určení optimální trajektorie ze stavu p do m vnoříme do následující třídy úloh.

Hledáme optimální cestu z libovolného stavu j, kde $j = 1, 2, \ldots, n$, do pevně určeného cílového stavu m tak, aby počet mezilehlých stavů byl nejvýše k ($k = 0, 1, \ldots, (n-2)$). Protože žádná trajektorie nemá smyčky, může mít libovolná trajektorie nejvýše $(n-2)$ mezilehlých stavů.

Optimální čas přechodu ze stavu j do cíle při nejvýše k mezilehlých stavech označíme jako $V_j(k)$ - to je optimální funkce pro naši úlohu. Zřejmě platí

$$V_j(0) = t_{jm},$$

což je přímý přechod z j do m. To je okrajová podmínka pro rekurentní výpočet $V_j(k)$. Pro optimální funkci platí

$$V_j(k) = \min_{i=1,\ldots,(n-1)} \left(t_{ji} + V_i(k-1)\right), \qquad k = 1, 2, \ldots, (n-2)$$

Předchozí rekurentní vztah je Bellmanova rovnice pro řešení naší úlohy. Zřejmě $V_p(n-2)$ je rovno minimální hodnotě kritéria pro přechod ze stavu p do cíle m.

Po výpočtu optimální funkce $V_j(k)$ konstruujeme optimální trajektorii následujícím postupem. Z počátečního stavu p se nejprve pohybujeme do stavu i_1 (první mezilehlý stav), platí-li pro něj

$$t_{pi_1} = V_p(n-2) - V_{i_1}(n-2-1)$$

Z bodu i_1 se potom dále pohybujeme do bodu i_2, platí-li pro něj

$$t_{i_1 i_2} = V_{i_1}(n-3) - V_{i_2}(n-4)$$

Tak postupujeme dále, obecně z bodu i_α se pohybujeme do $i_{\alpha+1}$, platí-li

$$t_{i_\alpha i_{\alpha+1}} = V_{i_\alpha}(n-2-\alpha) - V_{i_{\alpha+1}}(n-2-(\alpha+1))$$

Nejvýše po $(n-1)$ krocích se tímto postupem dostaneme po optimální trajektorii do cíle.

Hledejme nyní suboptimální trajektorii - například druhou nejlepší. Druhou nejmenší hodnotu kritéria z bodu j do cíle při nejvýše k mezilehlých stavech si označíme $U_j(k)$. Zřejmě $U_j(0)$ neexistuje, neboť přímý přechod je jediný. Zřejmě

$$U_j(1) = \overset{2}{\min}_{i=1,2,\ldots,(n-1)} \left[t_{ji} + V_i(k-1)\ ;\ t_{ji} + U_i(k-1)\right]$$

kde $\min^2$ je označení pro výběr druhé nejmenší hodnoty, která je různá od minima a $V_j(k)$ je optimální funkce pro původní úlohu.

Hledáme-li trajektorii, po které největší hodnota t_{ij} je co možno nejmenší, pak opět vnoříme naši úlohu do třídy úloh. Optimální hodnotu kritéria při cestě z bodu j do cíle m přes nejvýše k mezilehlých bodů, označíme nyní jako $S_j(k)$. Zřejmě $S_j(0) = t_{jn}$ a rekurentní vztah pro $S_j(k)$ je

$$S_j(k) = \min_{i=1,2,\ldots,(n-1)} \left[\max\left(t_{ji}\ ,\ S_i(k-1)\right)\right]$$

Z optimální funkce $S_j(k)$ můžeme opět rekonstruovat optimální trajektorii.

Úloha o falešné minci

Máme m mincí, mezi nimiž je jedna mince těžší (je falešná). Chceme použitím vahadlových vah bez závaží určit onu falešnou minci pomocí minimálního počtu vážení.

Úlohu řešíme tak, že z daných mincí vybereme dvě skupiny o u mincích. Ty dáme na misky vah. Budou-li misky v rovnováze, je falešná mince ve zbylých $(x - 2u)$ mincích, které jsme neváželi. Nebudou-li misky v rovnováze, je falešná mince v u mincích na těžší misce. Zřejmě platí omezení $1 \leq u \leq m/2$. Tento pokus opakujeme se skupinou mincí, mezi nimiž je falešná, tak dlouho, až určíme jednoznačně falešnou minci.

Proces opakovaného vážení je vlastně diskrétní proces, kde počet mincí m je výchozí stav procesu, u je řídicí veličina procesu a přirozené číslo k je pořadové číslo vážení. Zřejmě se jedná o úlohu časově optimálního diskrétního řízení. Hledáme minimální počet pokusů - označíme ho k^* - takový, abychom z počátečního stavu $x(0) = m$ přešli do koncového stavu $x(k^*) = 1$. Pro stav procesu v etapě $(k + 1)$ (po $(k + 1)$ vážení) platí

$$x(k+1) = \begin{cases} x(k) - 2u(k) & \text{jsou-li váhy v rovnováze} \\ u(k) & \text{nejsou-li váhy v rovnováze} \end{cases}$$

Toto je stavová rovnice procesu. Všimněme si, že je v ní neurčitost. Řídicí veličina $u(k)$ je rovna počtu mincí na miskách vah při $(k + 1)$ vážení, je omezena $1 \leq u(k) \leq x(k)/2$.

Pro řešení úlohy dynamickým programováním vnoříme naši úlohu do třídy úloh s libovolným počátečním stavem $x(0) = x$ (počet mincí). Zavedeme si optimální funkci $V(x)$, která je rovna minimálnímu počtu vážení, v nejméně příznivém případě, pro určení jedné falešné mince mezi x mincemi. Zřejmě platí

$$V(0) = 0, \quad V(1) = 0, \quad V(2) = 1, \quad V(3) = 1$$

což budou okrajové podmínky pro Bellmanovu rovnici, kterou nyní odvodíme. Z x mincí, které máme k dispozici, vezmeme dvě skupiny po u mincích, uděláme vážení a v nejhorším případě bude falešná mince mezi $\max\{x - 2u \ , \ u\}$ mincemi. Další postup již volíme optimální. Vybereme-li u v prvním pokusu optimální, pak podle principu optimality postupujeme optimálním způsobem. Proto pro optimální funkci $V(x)$ platí rekurentní vztah

$$V(x) = 1 + \min_{1 \leq u(k) \leq \frac{x(k)}{2}} \{\max [V(x - 2u) \ , \ V(u)]\}$$

Poznámka: Úloha v této formulaci je velmi jednoduchá. Optimální u bude zřejmě takové , že počet mincí x rozdělíme vždy na tři pokud možno stejné skupiny $(u, \ u, \ x-2u)$. Proto

$$\begin{aligned} u^* &= \frac{x}{3} && \text{pro} \quad x = 3i \\ u^* &= \frac{x-1}{3} && \text{pro} \quad x = 3i + 1 \\ u^* &= \frac{x+1}{3} && \text{pro} \quad x = 3i + 2 \end{aligned}$$

kde i je celé číslo. □

8.2.6 Řešení spojité úlohy optimálního řízení dynamickým programováním

Mějme problém optimálního řízení spojitého systému

$$\dot{\mathbf{x}}(t) = \mathbf{f}\left(\mathbf{x}(t), \mathbf{u}(t), t\right) \tag{8.29}$$

Chceme nalézt optimální řízení $\mathbf{u}^*(t) \in \mathbf{U}$ pro čas $t \in [t_0\ ,\ t_1]$ takové, které převádí počáteční stav $\mathbf{x}(t_0)$ do cílového stavu $\mathbf{x}(t_1)$ a minimalizuje kritérium kvality řízení

$$J(t_0, \mathbf{x}(t_0), \mathbf{u}(t)) = h(\mathbf{x}(t_1)) + \int_{t_0}^{t_1} g(\mathbf{x}(t), \mathbf{u}(t), t)\ dt \tag{8.30}$$

Při řešení této úlohy dynamickým programováním vnoříme tuto úlohu do třídy úloh tím, že uvolníme počáteční bod. Budeme tedy hledat optimální řízení $\mathbf{u}^*(t) \in \mathbf{U}$ na intervalu $t \in [\tau\ ,\ t_1]$, které systém (8.29) převádí z obecného počátečního stavu $\mathbf{x}(\tau) = \mathbf{s}$ v čase $\tau \in [t_0\ ,\ t_1]$ do koncového stavu $\mathbf{x}(t_1)$ v čase t_1 a minimalizuje kritérium (8.30). Kritérium má ale nyní dolní integrační mez rovnu počátečnímu času τ, je tedy závislé na $J(\tau, \mathbf{s}, \mathbf{u}(t))$.

Podobně jako v předchozích odstavcích si zavedeme optimální (Bellmanovu) funkci

$$V(\mathbf{s}, \tau) = \min_{\mathbf{u}(t),\ t\in[\tau,t_1]} J(\tau, \mathbf{s}, \mathbf{u}(t)) = J(\tau, \mathbf{s}, \mathbf{u}^*(t)) \tag{8.31}$$

Interval řízení $t \in [\tau, t_1]$ rozdělíme na dva subintervaly $[\tau\ ,\ \tau + \Delta\tau]$ a $[\tau + \Delta\tau\ ,\ t_1]$. Potom při optimálním řízení $\mathbf{u}^*(t)$ platí pro Bellmanovu funkci

$$V(\mathbf{s}, \tau) = \int_{\tau}^{\tau+\Delta\tau} g(\mathbf{x}, \mathbf{u}^*, t)\ dt + \ h(\mathbf{x}(t_1)) + \int_{\tau+\Delta\tau}^{t_1} g(\mathbf{x}, \mathbf{u}^*, t)\ dt. \tag{8.32}$$

Odtud plyne

$$V(\mathbf{s}, \tau) = \int_{\tau}^{\tau+\Delta\tau} g(\mathbf{x}, \mathbf{u}^*, t)\ dt + V(\mathbf{s} + \Delta\mathbf{s}, \tau + \Delta\tau). \tag{8.33}$$

Z úpravy (8.33) ve tvaru

$$V(\mathbf{s} + \Delta\mathbf{s}, \tau + \Delta\tau) - V(\mathbf{s}, \tau) = -\int_{\tau}^{\tau+\Delta\tau} g(\mathbf{x}, \mathbf{u}^*, t)\ dt \tag{8.34}$$

plyne při $\Delta\tau \to 0$ vztah

$$\frac{dV_+(\mathbf{s}, \tau)}{d\tau} = -g(\mathbf{x}, \mathbf{u}^*, t), \tag{8.35}$$

kde symbolem + značíme derivaci zprava.

Zvolíme řízení $\mathbf{u}(t)$ takové, že na prvním intervalu $t \in [\tau\ ,\ \tau + \Delta\tau]$ je libovolné, ale na druhém intervalu $t \in [\tau + \Delta\tau\ ,\ t_1]$ je optimální vzhledem ke stavu $\mathbf{x}(\tau + \Delta\tau) = \mathbf{s} + \overline{\Delta\mathbf{s}}$, který je výsledkem řízení v prvním intervalu. Pak podobně jako v (8.33) platí pro Bellmanovu funkci

$$V(\mathbf{s}, \tau) \leq \int_{\tau}^{\tau+\Delta\tau} g(\mathbf{x}, \mathbf{u}, t)\ dt + V(\mathbf{s} + \overline{\Delta\mathbf{s}}, \tau + \Delta\tau). \tag{8.36}$$

Odtud pro libobovolné $\mathbf{u}(t) \in \mathbf{U}$ platí

$$\frac{dV_+(\mathbf{s}, \tau)}{d\tau} \geq -g(\mathbf{x}, \mathbf{u}, t). \tag{8.37}$$

Porovnáním (8.37) a (8.35) dostaneme

$$\frac{dV_+(\mathbf{s},\tau)}{d\tau} + g(\mathbf{x},\mathbf{u}^*,t) = \min_{\mathbf{u}\in\mathbf{U}} \left(\frac{dV_+(\mathbf{s},\tau)}{d\tau} + g(\mathbf{x},\mathbf{u},t) \right) = 0 \tag{8.38}$$

což je implicitní tvar Bellmanovy rovnice. Má-li optimální funkce $V(\mathbf{s},\tau)$ spojité derivace, pak platí

$$\frac{dV_+(\mathbf{s},\tau)}{d\tau} = \frac{dV(\mathbf{s},\tau)}{d\tau} = \frac{\partial V}{\partial \tau} + \frac{\partial V}{\partial \mathbf{s}}\ \dot{\mathbf{s}} = \frac{\partial V}{\partial \tau} + \frac{\partial V}{\partial \mathbf{s}}\ \mathbf{f}(\mathbf{s},\mathbf{u},\tau) \tag{8.39}$$

Potom z (8.38) dostaneme konečně explicitní tvar Bellmanovy rovnice

$$\min_{\mathbf{u}(\tau)\in\mathbf{U}} \left(g(\mathbf{s},\mathbf{u},\tau) + \frac{\partial V}{\partial \tau} + \frac{\partial V}{\partial \mathbf{s}}\ \mathbf{f}(\mathbf{s},\mathbf{u},\tau) \right) = 0 \tag{8.40}$$

což je obecně nelineární parciální diferenciální rovnice. Okrajová podmínka pro její řešení je zřejmě

$$V(\mathbf{s},t_1) = h(\mathbf{s}). \tag{8.41}$$

Řešením Bellmanovy rovnice (8.40) z okrajové podmínky (8.41) dostaneme optimální funkci $V(\mathbf{s},\tau)$ a optimální řízení $\mathbf{u}^*$, které je funkcí okamžitého stavu systému, čili $\mathbf{u}^* = \mathbf{u}^*(\mathbf{s},\tau)$. Tím je provedena **syntéza řízení - syntéza optimálního zpětnovazebního regulátoru**.

Je-li systém časově invariantní, jádro funkcionálu g v kritériu jakosti řízení není závislé na čase a koncový čas t_1 je buď volný, nebo $t_1 \to \infty$, pak optimální funkce také není explicitně závislá na čase. Platí tedy

$$V(\mathbf{s},\tau) = V(\mathbf{s}) \quad \text{a proto} \quad \frac{\partial V}{\partial \tau} = 0$$

Pro existenci Bellmanovy rovnice pro všechny stavy $\mathbf{s} \in \mathbf{X}$ je nutno učinit předpoklad o diferencovatelnosti Bellmanovy funkce. Vlivem tohoto předpokladu je Bellmanova rovnice pouze postačující podmínkou optimality. To znamená, že najdeme-li z Bellmanovy rovnice řídicí funkci $\mathbf{u}(\mathbf{x},t)$, která je jejím řešením, pak takové řízení je optimální řízení.

Bellmanovu rovnici můžeme řešit pouze ve speciálních případech. Obvykle ji řešíme diskretizací, nebo přímo diskretizujeme výchozí problém.

Uvedeme nyní řešení tří příkladů, v prvních dvou příkladech je ukázáno, že předpoklad diferencovatelnosti Bellmanovy funkce není někdy splněn ani v nejjednodušších případech.

Příklad 1: Mějme systém druhého řádu se stavovou rovnicí

$$\begin{aligned} \dot{x}_1 &= u_1, \quad x_1(0) = y_1 \quad |u_1(t)| \le \alpha \\ \dot{x}_2 &= u_2, \quad x_2(0) = y_2 \quad |u_2(t)| \le \beta \end{aligned}$$

Hledáme optimální řízení takové, které převede počáteční stav do nuly $x_1(t_1) = x_2(t_1) = 0$ za minimální čas t_1.

Bellmanova rovnice pro tento problém je zřejmě

$$\min_{\mathbf{u}(t)\in\mathbf{U}} \left(1 + \frac{\partial V}{\partial y_1} u_1 + \frac{\partial V}{\partial y_2} u_2 \right) = 0, \quad V(0) = 0$$

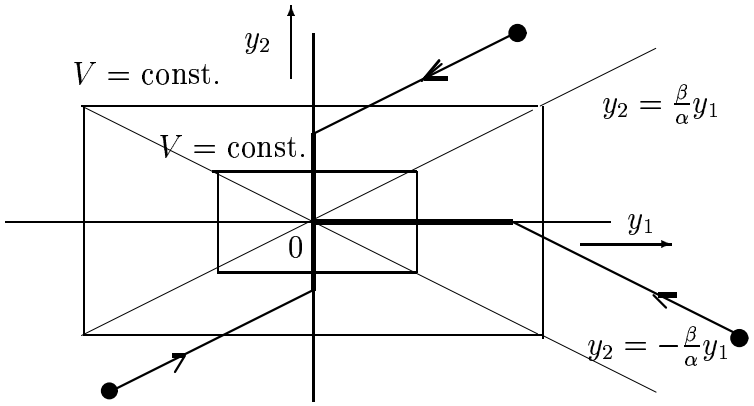
kde $V(y_1, y_2)$ je optimální funkce. Protože se jedná o dva nezávislé integrátory, bude zřejmě optimální řízení takové, aby vstup integrátorů byl maximální a tím se výstup největší možnou rychlostí blížil nule. Proto

$$u_1^*(\mathbf{y}, t) = -\alpha \text{sign } y_1, \qquad u_2^*(\mathbf{y}, t) = -\beta \text{sign } y_2$$

Optimální Bellmanova funkce je rovna většímu z obou časů, za které se oba integrátory vynulují

$$V(\mathbf{y}) = \max\left(\frac{1}{\alpha}|y_1| \ ; \ \frac{1}{\beta}|y_2|\right)$$

Křivky konstantní hodnoty Bellmanovy funkce jsou vyneseny na obr. 8.5. Optimální tra-



Obrázek 8.5: Křivky konstantní hodnoty optimální funkce a optimální trajektorie

jektorie jsou zakresleny na obr. 8.5 silnými čarami,. Je zřejmé, že na přímkách $y_2 = \pm\frac{\beta}{\alpha} y_1$ je Bellmanova funkce nediferencovatelná. Protože optimální trajektorie, které nezačínají na zmíněných přímkách, leží celé mimo ně, Bellmanova funkce je na nich diferencovatelná a předchozí Bellmanovu rovnici lze použít.

Příklad 2: Hledejme časově optimální řízení systému

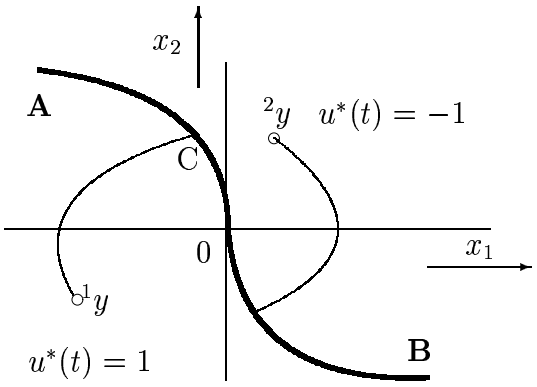
$$\begin{aligned} \dot{x}_1 &= x_2, \quad x_1(0) = y_1, \qquad |u(t)| \leq 1 \\ \dot{x}_2 &= u, \quad x_2(0) = y_2. \end{aligned}$$

Je možno ukázat, že optimální řízení je $u^*(t) = +1$, je-li stav $\mathbf{x}$ vlevo od křivky AOB na obr. 8.6 a $u^*(t) = -1$, je-li stav $\mathbf{x}$ vpravo od křivky AOB. Křivka AOB se nazývá **přepínací křivka**.

Je-li počáteční bod $\mathbf{y}(0) = {}^1\mathbf{y}$ (viz obr. 8.6), je optimální řízení nejprve $u^*(t) = +1$, až trajektorie dosáhne v bodě C přepínací křivky. Potom řízení bude $u^*(t) = -1$ a optimální trajektorie se po přepínací křivce blíží k počátku. Podobně pro počáteční bod $\mathbf{y} = {}^2\mathbf{y}$ - viz obr. 8.6. Počáteční bod můžeme tedy dosáhnou pouze po přepínací křivce.

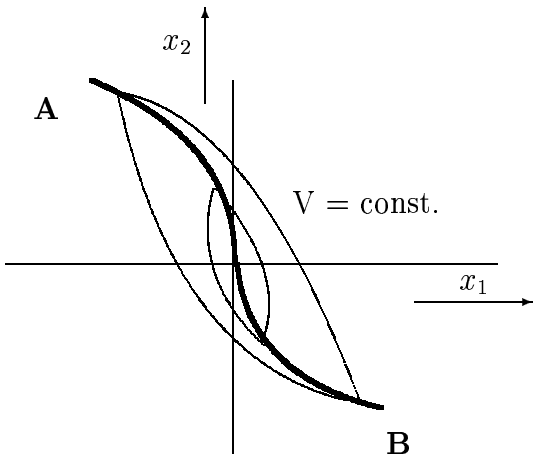
Na přepínací křivce v části AO je řízení $u^*(t) = -1$ a v části BO je řízení $u^*(t) = +1$. Bellmanova funkce $V(\mathbf{y})$ pro tento problém je

$$V(\mathbf{y}) = \begin{cases} -y_2 + 2\sqrt{-y_1 + \frac{1}{2}y_2^2}, & \text{je-li } \mathbf{y} \text{ vlevo od } AOB \\ +y_2 + 2\sqrt{y_1 + \frac{1}{2}y_2^2}, & \text{je-li } \mathbf{y} \text{ vpravo od } AOB \end{cases}$$



Obrázek 8.6: Přepínací křívka a optimální trajektorie

Body konstantní hodnoty Bellmanovy funkce $V(\mathbf{y})$ jsou vyneseny na obr. 8.7. Z obrázku je zřejmé, že na přepínací křivce jsou derivace optimální funkce nespojité. Při tom každá optimální trajektorie probíhá částečně po přepínací křivce. Bellmanovu rovnici nelze v tomto případě použít.



Obrázek 8.7: Geometrické místo bodů konstantní hodnoty Bellmanovy funkce

Příklad 3: Nyní dynamickým programováním vyřešíme úlohu o optimálním doběhu stejnosměrného motoru. Tato úloha byla formulována a vyřešena variačními metodami v předchozí kapitole. Jedná se tedy o problém

$$\min_u \left\{ J(u) = \int_0^\infty \left(u^*(y) - \omega(t)u(t) + \alpha\omega^2(t) \right) \, dt, \, : \, \dot{\omega}(t) = u(t) - \omega(t) \right\}$$

Okrajové podmínky jsou $\omega(0) = \omega_0$, $\omega(\infty) = 0$. Bellmanova rovnice pro tuto úlohu je

$$0 = \min_u \left[u^2(t) - \omega(t)u(t) + \alpha\omega^2(t) + \frac{\partial V}{\partial t} + \frac{\partial V}{\partial \omega} \left(u(t) - \omega(t) \right) \right]$$

Optimální funkce nezávisí na čase a proto $\frac{\partial V}{\partial t} = 0$. Protože řízení u není omezeno, nalezneme minimum v Bellmanově rovnici derivováním výrazu v závorce podle u, pak

$$2u^*(t) - \omega(t) + \frac{\partial V}{\partial \omega} = 0$$

Dosadíme-li za $\frac{\partial V}{\partial \omega} = -2u^* - \omega$ z předchozí rovnice zpět do Bellmanovy rovnice dostaneme

$$(u^*)^2 - \omega u + \alpha\omega^2 + (\omega - 2u^*)(u^* - \omega) = 0$$

Odtud plyne optimální řízení

$$u^* = \omega\left(1 \pm \sqrt{\alpha}\right)$$

Znamémko + v předchozím výrazu nemá fyzikální význam a proto platí

$$\begin{aligned} u^* &= \omega\left(1 - \sqrt{\alpha}\right) \\ \dot{\omega} &= -\sqrt{\alpha}\omega \end{aligned}$$

Výsledek souhlasí s řešením stejného příkladu variačními metodami v předchozí kapitole.

8.2.7 Příklady

1. Modifikujte úlohu o optimálním průchodu sítí z prvního odstavce této kapitoly tak, že
   a) vynecháme předpoklad, že souřadnice trajektorie neklesají
   b) hledáme druhou nejlepší cestu
   c) změníme kritérium tak, aby maximální hodnota kritéria na úsečce, po které prochází trajektorie, byla maximální.

2. Řešte úlohu optimálního diskrétního řízení, je-li koncový čas volný a leží v intervalu $k_1 \in [\alpha\ ,\ \beta]$. Určete Bellmanovu rovnici pro tento případ.

3. Napište rekurentní vztahy pro řešení úlohy lineárního programování pomocí dynamického programování. Vnořte úlohu lineárního programování do třídy úloh a sestavte Bellmanovu rovnici. Je zde výhodné málo omezení či málo proměnných?

4. Modifikujte úlohu na optimální průchod sítí tak, aby druhá suboptimální cesta se s optimální cestou nestýkala nikde, kromě počátečního a koncového bodu.

5. Modifikujte úlohu o falešné minci:
   a) o falešné minci víme pouze to, že má jinou váhu než ostatní mince.
   b) ve skupině m mincí máme dvě falešné mince.

6. Pomocí interpolace odvoďte vztah pro určení hodnoty optimální funkce $V(\mathbf{x}, k)$ z vypočtených hodnot $V(\mathbf{x}_i, k)$, kde body $\mathbf{x}_i$ jsou body zvolené sítě stavů. Řešte nejprve pro $\mathbf{x} \in R^2$, $\mathbf{x} \in R^3$ atd.

7. Úloha o jeepu:
   Naším úkolem je překonat v pustině vzdálenost d mezi výchozím bodem a cílem.

Vozidlo plně naložené palivem ujede však pouze vzdálenost $d/2$. Proto je nutno postupovat k cíli tak, že na trati vytváříme mezisklady paliva.

Jak je nutno naplánovat cestu, aby celková spotřeba paliva na rozvoz paliva do meziskladů a na dosažení cíle byla minimální. Jinými slovy chceme, aby celková ujetá dráha byla minimální.

Úlohu řešte úvahou a také dynamickým programováním. Optimum je jediné, optimální trajektorie není jediná.

Modifikujte úlohu tak, že vozidlo plně naložené palivem ujede pouze do vzdálenosti $a < d$. Uvažujte také, že poloha některých mezistanic je předem určena - byly zřízeny předchozími expedicemi.

8. Sestavte program pro řešení obecného problému optimálního řízení nelineárního diskrétního dynamického systému s obecným kritériem kvality řízení. Proveďte nejprve hrubou diskretizaci možných hodnot stavů a řízení. Kolem hrubě vypočtené optimální trajektorie proveďte zjemnění diskretizace stavů i řízení a určete novou optimální trajektorii ve zjemněné síti. Toto zjemnění několikráte opakujte až dostanete vyhovující řešení. Tímto způsobem lze zmírnit ono "prokletí rozměrnosti" při řešení problému diskrétní optimalizace dynamickým programováním.

Kapitola 9

Princip maxima

V této kapitole odvodíme Pontrjaginův princip maxima, který je nutnou podmínkou řešení problémů dynamické optimalizace a ukážeme jeho použití pro řešení optimalizačních problémů. Nejprve ale ukážeme na souvislosti různých metod dynamické optimalizace.

9.1 Souvislost dynamického programování a variačních metod

Mějme tedy opět základní variační úlohu. Hledáme minimum funkcionálu

$$J\left(\mathbf{x}(t)\right) = \int_{t_0}^{t_1} g\left(\mathbf{x}(t), \dot{\mathbf{x}}(t), t\right)\ dt, \qquad \mathbf{x}(t_0) = \mathbf{x}_0, \mathbf{x}(t_1) = \mathbf{x}_1 \tag{9.1}$$

Pro řešení této úlohy dynamickým programováním si zavedeme optimální funkci $V(\mathbf{x}, t)$ (viz předchozí kapitola), která je rovna

$$V(\mathbf{x}, t) = \min \int_{t}^{t_1} g\left(\mathbf{x}(t), \dot{\mathbf{x}}(t), t\right)\ dt \tag{9.2}$$

Známe-li derivaci $\dot{\mathbf{x}}(t)$ pro všechna t, známe potom i řešení $\mathbf{x}(t)$. Derivace $\dot{\mathbf{x}}(t)$ zde je vlastně jakousi "řídicí veličinou". Postupem provedeným v předchozí kapitole dostaneme Bellmanovu rovnici pro optimální funkci $V(\mathbf{x}, t)$. Zřejmě platí

$$V(\mathbf{x}, t) = \min_{\dot{\mathbf{x}}} \left\{ \int_{t}^{t+\Delta t} g\left(\mathbf{x}(t), \dot{\mathbf{x}}(t), t\right)\ dt + \int_{t+\Delta t}^{t_1} g\left(\mathbf{x}(t), \dot{\mathbf{x}}(t), t\right)\ dt \right\}$$

Odtud

$$V(\mathbf{x}, t) = \min_{\dot{\mathbf{x}}} \left\{ \int_{t}^{t+\Delta t} g\left(\mathbf{x}(t), \dot{\mathbf{x}}(t), t\right)\ dt + V(\mathbf{x} + \Delta\mathbf{x}, t + \Delta t) \right\} \tag{9.3}$$

Upravíme výrazy ve složené závorce následujícím způsobem

$$\begin{aligned} \int_{t}^{t+\Delta t} g\left(\mathbf{x}(t), \dot{\mathbf{x}}(t), t\right)\ dt &= g\left(\mathbf{x}(t), \dot{\mathbf{x}}(t), t\right)\Delta t + \varepsilon_1(\Delta t), \\ V(\mathbf{x} + \Delta\mathbf{x}, t + \Delta t) &= V(\mathbf{x}, t) + \frac{\partial V}{\partial t}\Delta t + \frac{\partial V}{\partial \mathbf{x}}\frac{d\mathbf{x}}{dt}\Delta t + \varepsilon_2(\Delta t), \\ \lim_{\Delta t \to 0} \frac{\varepsilon_1(\Delta t)}{\Delta t} &= 0, \qquad \lim_{\Delta t \to 0} \frac{\varepsilon_2(\Delta t)}{\Delta t} = 0 \end{aligned}$$

Předpokládáme existenci a omezenost druhých parciálních derivací $V(\mathbf{x},t)$ a existenci derivace $\dot{\mathbf{x}}(t)$. Po úpravě vztahu (9.3) je Bellmanova rovnice pro základní variační úlohu ve tvaru

$$0 = \min_{\dot{\mathbf{x}}} \left\{ g\left(\mathbf{x}(t), \dot{\mathbf{x}}(t), t\right) + \frac{\partial V}{\partial t} + \frac{\partial V}{\partial \mathbf{x}} \dot{\mathbf{x}} \right\} \tag{9.4}$$

Z ní zřejmě plynou dvě rovnice

$$\begin{aligned} -\, g\left(\mathbf{x}(t), \dot{\mathbf{x}}(t), t\right) &= \frac{\partial V}{\partial t} + \frac{\partial V}{\partial \mathbf{x}} \dot{\mathbf{x}} \\ -\frac{\partial g\left(\mathbf{x}(t), \dot{\mathbf{x}}(t), t\right)}{\partial \dot{\mathbf{x}}} &= \frac{\partial V}{\partial \mathbf{x}} \end{aligned} \tag{9.5}$$

Rovnice (9.5a) ukazuje, že výraz v závorce v (9.4) je roven nule. Rovnice (9.5b) je nutná podmínka pro minimum v (9.4) a sice, že derivace výrazu v závorce v (9.4) podle $\dot{\mathbf{x}}$ je nulová.

Z Bellmanovy rovnice (9.4) nebo z (9.5) již snadno odvodíme Eulerovu - Lagrangeovu rovnici. Derivujeme (9.5b) podle času, pak

$$-\frac{d}{dt} g_{\dot{\mathbf{x}}}\left(\mathbf{x}(t), \dot{\mathbf{x}}(t), t\right) = \frac{\partial^2 V}{\partial t \partial \mathbf{x}} + \frac{\partial^2 V}{\partial \mathbf{x}^2} \dot{\mathbf{x}} \tag{9.6}$$

Nyní derivujeme parciálně podle $\mathbf{x}$ rovnici (9.5a), pak

$$-\, g_{\mathbf{x}} - g_{\dot{\mathbf{x}}} \frac{d\dot{\mathbf{x}}}{d\mathbf{x}} = \frac{\partial^2 V}{\partial t \partial \mathbf{x}} + \frac{\partial^2 V}{\partial \mathbf{x}^2} \dot{\mathbf{x}} + \frac{\partial V}{\partial \mathbf{x}} \frac{d\dot{\mathbf{x}}}{d\mathbf{x}} \tag{9.7}$$

Dosadíme z (9.6) do (9.7), pak

$$-g_{\mathbf{x}} - g_{\dot{\mathbf{x}}} \frac{d\dot{\mathbf{x}}}{d\mathbf{x}} = -\frac{d}{dt} g_{\dot{\mathbf{x}}} + \frac{\partial V}{\partial \mathbf{x}} \frac{d\dot{\mathbf{x}}}{d\mathbf{x}}$$

Předchozí rovnici upravíme do následujícího tvaru

$$g_{\mathbf{x}} - \frac{d}{dt} g_{\dot{\mathbf{x}}} + \left(\frac{\partial V}{\partial \mathbf{x}} + g_{\dot{\mathbf{x}}} \right) \frac{d\dot{\mathbf{x}}}{d\mathbf{x}} = 0 \tag{9.8}$$

Protože výraz v závorce na levé straně předchozí rovnice je podle (9.5b) roven nule, je (9.8) totožná s Eulerovou - Lagrangeovou rovnicí $g_{\mathbf{x}} - \frac{d}{dt} g_{\dot{\mathbf{x}}} = 0$.

Legendrovu podmínku odvodíme z Bellmanovy rovnice také snadno. Aby v (9.4) byl výraz ve složené závorce minimální, musí být jeho druhá derivace podle $\dot{\mathbf{x}}$ nezáporná. Proto

$$\frac{\partial^2}{\partial \dot{\mathbf{x}}^2} \left\{ g\left(\mathbf{x}(t), \dot{\mathbf{x}}(t), t\right) + \frac{\partial V}{\partial t} + \frac{\partial V}{\partial \mathbf{x}} \dot{\mathbf{x}} \right\} \geq 0$$

Odtud přímo plyne Legendrova podmínka

$$g_{\dot{\mathbf{x}}\dot{\mathbf{x}}} \geq 0, \tag{9.9}$$

neboť optimální funkce nezávisí na $\dot{\mathbf{x}}$.

Legendrova podmínka nevylučuje, že minimum je pouze relativní. Aby minimum existovalo vůči všem funkcím $\dot{\mathbf{x}} = \mathbf{z}(\mathbf{x}, t)$, musí podle (9.4) platit

$$g\left(\mathbf{x}(t), \mathbf{s}(t), t\right) + \frac{\partial V}{\partial t} + \frac{\partial V}{\partial \mathbf{x}}\mathbf{s} \leq g\left(\mathbf{x}(t), \mathbf{z}(t), t\right) + \frac{\partial V}{\partial t} + \frac{\partial V}{\partial \mathbf{x}}\mathbf{z}$$

kde $\dot{\mathbf{x}} = \mathbf{s}(\mathbf{x}, t)$ je optimální hodnota derivace a $\dot{\mathbf{x}} = \mathbf{z}(\mathbf{x}, t)$ je libovolná jiná hodnota derivace. Z předchozí rovnice plyne

$$g\left(\mathbf{x}(t), \mathbf{z}(t), t\right) - g\left(\mathbf{x}(t), \mathbf{s}(t), t\right) + \frac{\partial V}{\partial \mathbf{x}}\left(\mathbf{z} - \mathbf{s}\right) \geq 0$$

Odtud po dosazení z (9.5b) dostaneme

$$g\left(\mathbf{x}(t), \mathbf{z}(t), t\right) - g\left(\mathbf{x}(t), \mathbf{s}(t), t\right) - \frac{\partial g}{\partial \mathbf{s}}\left(\mathbf{z} - \mathbf{s}\right) \geq 0 \tag{9.10}$$

Levá strana (9.10) je rovna Weierstrassově funkci $E(\mathbf{x}, \mathbf{z}, \mathbf{s}, t)$ a vztah (9.10) je Weierstrassova postačující podmínka.

Není-li koncový bod $\mathbf{x}(t_1)$ trajektorie určen, je třeba určit podmínky platné v koncovém bodě - podmínky transverzality. Optimální koncový bod má tu vlastnost, že změníme-li jeho polohu, pak hodnota funkcionálu $J(\mathbf{x}(t))$ vzroste (spíše neklesne). Proto v optimálním koncovém bodě platí

$$\left.\frac{\partial V(\mathbf{x}, t)}{\partial \mathbf{x}}\right|_{t=t_1} = 0$$

Z (9.5b) potom plyne, že v optimálním koncovém bodě platí

$$g_{\dot{\mathbf{x}}}|_{t=t_1} = 0, \tag{9.11}$$

což jsou podmínky transverzality pro volný konec trajektorie a pevný koncový čas. Stejně bychom odvodili podmínky transverzality pro volný koncový čas.

Leží-li koncový bod na křivce $\mathbf{x} = \varphi(t)$, pak při pohybu koncového bodu po této křivce, bude změna funkcionálu $J(\mathbf{x}, t)$ i optimální funkce $V(\mathbf{x}, t)$ nulová. Platí tedy

$$\delta V(\mathbf{x}, t)|_{t=t_1, x=\varphi(t_1)} = 0$$

Proto

$$\frac{\partial V}{\partial t} + \frac{\partial V}{\partial \mathbf{x}}\frac{d\mathbf{x}}{dt} = \frac{\partial V}{\partial t} + \frac{\partial V}{\partial \mathbf{x}}\dot{\varphi}(t) = 0, \qquad \text{pro} \quad t = t_1$$

Dosadíme-li za $\frac{\partial V}{\partial t}$ z (9.5a) dostaneme

$$-g\left(\mathbf{x}, \dot{\mathbf{x}}, t\right) - \frac{\partial V}{\partial \mathbf{x}}\dot{\mathbf{x}} + \frac{\partial V}{\partial \mathbf{x}}\dot{\varphi} = 0, \qquad \text{pro} \quad t = t_1.$$

Nyní upravíme podle (9.5b), pak

$$g\left(\mathbf{x}, \dot{\mathbf{x}}, t\right) - g_{\dot{\mathbf{x}}}\left(\dot{\varphi} - \dot{\mathbf{x}}\right) = 0, \qquad \text{pro} \quad t = t_1, \tag{9.12}$$

což jsou opět podmínky transverzality pro tento případ.

Podobným způsobem bychom ukázali souvislost Bellmanovy rovnice a Eulerovy - Lagrangeovy rovnice pro problém optimálního řízení.

9.2 Dynamické programování a princip maxima

V této sekci ukážeme souvislost dynamického programování a principu maxima. Pricip maxima byl vysloven jako nutná podmínka řešení úlohy optimálního řízení v roce 1956 L. S. Pontrjaginem. Jeho odvození nevyžaduje některé předpoklady platné pro dynamické programování (diferencovatelnost optimální funkce $V(\mathbf{x}, t)$).

Pokud však je optimální funkce diferencovatelná, je možno snadno ukázat souvislost principu maxima a dynamického programování. Mějme tedy problém optimálního řízení

$$\min_{\mathbf{u}(t)} \left\{ J = h(\mathbf{x}(t_1)) + \int_{t_0}^{t_1} g(\mathbf{x}, \mathbf{u}, t)\ dt, \quad \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, \mathbf{u}, t),\ x(t_0) = x_0 \right\} \tag{9.13}$$

Tuto Bolzovu úlohu převedeme na Mayerovu úlohu a odstraníme explicitní závislost na čase. Rozšíříme stav systému o dvě složky, nejprve zavedeme $x_{n+1}(t) = t$, která je pak určena diferenciální rovnicí

$$\frac{dx_{n+1}(t)}{dt} = f_{n+1} = 1, \qquad x_{n+1}(t_0) = t_0 \tag{9.14}$$

Podobně zavedeme další pomocnou proměnnou $x_0(t)$, která je rovna integrálnímu členu v kritériu optimality v (9.13). Pak platí

$$\frac{dx_0(t)}{dt} = f_0 = g(\mathbf{x}, \mathbf{u}, t), \qquad x_0(t_0) = 0 \tag{9.15}$$

a proto $x_0(t_1)$ je rovno

$$x_0(t_1) = \int_{t_0}^{t_1} g(\mathbf{x}, \mathbf{u}, t)\ dt$$

Minimalizace kritéria v (9.13) - Bolzova úloha - je tím změněna na úlohu minimalizace koncového stavu - Mayerova úloha

$$\min_{\mathbf{u}(t)} \{ J = h(\mathbf{x}(t_1)) + x_0(t_1), \quad \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, \mathbf{u}, t),\ x(t_0) = x_0 \} \tag{9.16}$$

Složky x_0 a x_{n+1} připojíme ke složkám stavového vektoru $\mathbf{x}$ systému a dostaneme zobecněný stavový vektor $\overline{\mathbf{x}}$ v $(n+2)$-rozměrném prostoru

$$\overline{\mathbf{x}} = [x_0, x_1, \ldots, x_n, x_{n+1}]^T \tag{9.17}$$

Podobně zavedeme zobecněný vektor pravých stran stavové rovnice systému (omezení v (9.13) a diferenciálních rovnic (9.14) a (9.15)

$$\overline{\mathbf{f}} = [f_0, f_1, \ldots, f_n, f_{n+1}]^T \tag{9.18}$$

Optimální řízení můžeme určit dynamickým programováním řešením Bellmanovy rovnice

$$-\frac{\partial V}{\partial t} = \min_{\mathbf{u}(t) \in \mathbf{U}} \left(g(\mathbf{x}, \mathbf{u}, t) + \frac{\partial V}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \right) \tag{9.19}$$

s okrajovou podmínkou $V(\mathbf{x}, t_1) = h(\mathbf{x}(t_1))$. Dále zavedeme vektor $\overline{\mathbf{p}}(t)$ rozměru $(n+2)$

$$\overline{\mathbf{p}}(t) = \left[\overline{p}_0, \overline{p}_1, \ldots, \overline{p}_n, \overline{p}_{n+1} \right]^T = \left[-1, -\frac{\partial V}{\partial x_1}, \ldots, -\frac{\partial V}{\partial x_n}, -\frac{\partial V}{\partial x_{n+1}} \right]^T \tag{9.20}$$

Vektor $\overline{\mathbf{p}}(t)$ se nazývá **konjugovaný vektor**. Jeho složky jsou rovny zápornému gradientu optimální funkce $V(\overline{\mathbf{x}}, t)$.

Bellmanovu rovnici (9.19) upravíme tak, abychom mohli dosadit zobecněné vektory $\overline{\mathbf{x}}$, $\overline{\mathbf{p}}$ a funkci $\overline{\mathbf{f}}$. Z (9.19) přímo plyne

$$0 = \min_{\mathbf{u}(t)\in\mathbf{U}} \left(g(\mathbf{x}, \mathbf{u}, t) + \frac{\partial V}{\partial x_1} f_1 + \ldots + \frac{\partial V}{\partial x_n} f_n + \frac{\partial V}{\partial t} \right)$$

Minimalizaci předchozího výrazu zaměníme na maximalizaci jeho negace, pak

$$0 = \max_{\mathbf{u}(t)\in\mathbf{U}} \left((-1) g(\mathbf{x}, \mathbf{u}, t) - \frac{\partial V}{\partial x_1} f_1 - \ldots - \frac{\partial V}{\partial x_n} f_n - \frac{\partial V}{\partial x_{n+1}} (+1) \right) \tag{9.21}$$

Dosazením $\overline{\mathbf{p}}$ podle (9.20) a $\overline{\mathbf{f}}$ podle (9.18) dostaneme z předchozí rovnice

$$0 = \max_{\mathbf{u}(t)\in\mathbf{U}} \left(\overline{\mathbf{f}}^T \overline{\mathbf{p}} \right) = \max_{\mathbf{u}(t)\in\mathbf{U}} \sum_{i=0}^{n+1} f_i \; p_i \tag{9.22}$$

Zavedením skalární **Hamiltonovy funkce**

$$\overline{\mathbf{H}} \left(\overline{\mathbf{x}}, \overline{\mathbf{p}}, \mathbf{u} \right) = \overline{\mathbf{f}}^T \; \overline{\mathbf{p}} = \sum_{i=0}^{n+1} f_i \; p_i \tag{9.23}$$

dostaneme z (9.22) podmínkovou rovnici

$$0 = \max_{\mathbf{u}(t)\in\mathbf{U}} \overline{\mathbf{H}} \left(\overline{\mathbf{x}}, \overline{\mathbf{p}}, \mathbf{u} \right) \tag{9.24}$$

což je tak zvaný **Pontrjaginův princip maxima**.

Pontrjaginův princip maxima vyžaduje, aby při optimálním řízení $\mathbf{u}^*(t)$ byla Hamiltonova funkce $\overline{\mathbf{H}} \left(\overline{\mathbf{x}}, \overline{\mathbf{p}}, \mathbf{u} \right)$ v každém časovém okamžiku t maximální vůči řízení $\mathbf{u}(t)$.

Algoritmus nalezení optimálního řízení je tedy v principu dosti jednoduchý. Optimální řízení $\mathbf{u}^*(t)$ je takové řízení, které maximalizuje Hamiltonián. Navíc platí, že Hamiltonián je při optimálním řízení konstantní a je roven nule. Jednoduchou aplikaci principu maxima znemožňuje to, že neznáme optimální funkci $V(\mathbf{x}, t)$, ani její gradienty.

Zavedeme si rozšířenou optimální funkci

$$\overline{V}(\overline{\mathbf{x}}) = x_0 + V(\mathbf{x}, t) = x_0 + V(x_1, \ldots, x_n, x_{n+1}) \tag{9.25}$$

V $(n+2)$-rozměrném prostoru $\overline{\mathbf{x}}$ existují izoplochy $\overline{V}(\overline{\mathbf{x}}) = \text{konst.}$. Je zřejmé, že podle (9.20) je konjugovaný vektor $\overline{\mathbf{p}}$ roven zápornému gradientu funkce $\overline{V}(\overline{\mathbf{x}})$

$$\overline{\mathbf{p}} = -\text{grad}_{\overline{x}} \overline{V}(\overline{\mathbf{x}}) \tag{9.26}$$

Vektor $\overline{\mathbf{f}}$ zavedený v (9.18) je vektor stavových rychlostí $\overline{\mathbf{f}} = \dfrac{d\overline{\mathbf{x}}}{dt}$. Princip maxima tedy vyžaduje, aby skalární součin $\overline{\mathbf{p}}^T \overline{\mathbf{f}} = \overline{\mathbf{H}}$ byl maximální, tedy aby projekce vektoru rychlosti $\overline{\mathbf{f}}$ na vektor $\overline{\mathbf{p}}$ byla maximální. Tato projekce je nekladná a pro optimální řízení je nulová - vektory $\overline{\mathbf{p}}$ a $\overline{\mathbf{f}}$ jsou tedy, při optimálním řízení, na sebe kolmé.

Názornější představě brání ta okolnost, že funkce $\overline{V}(\overline{\mathbf{x}})$ podle (9.25) není rovna optimální funkci $V(\mathbf{x}, t)$. Podle (9.25) je o $x_0 = \int_{t_0}^{t_1} g\ dt$ větší a je definována v prostoru větší dimenze.

Poznámka: Zavedením vektoru $\overline{\mathbf{p}}$ s opačným znaménkem se zřejmě maximalizace Hamiltoniánu změní na jeho minimalizaci. Dostaneme tedy tzv. **princip minima**, který se od principu maxima liší pouze znaménkem konjugovaného vektoru $\overline{\mathbf{p}}$. Používání principu minima má logiku v tom, že minimalizaci kritéria odpovídá minimalizace Hamiltoniánu. Zde zachováme Pontrjaginem zavedený princip maxima.

□

Určení optimální funkce $V(\mathbf{x}, t)$ vyžaduje řešit Bellmanovu parciální diferenciální rovnici. Při použití principu maxima potřebujeme znát konjugovaný vektor $\overline{\mathbf{p}}(t)$, který však lze určit, aniž bychom počítali optimální funkci $\overline{V}(\overline{\mathbf{x}})$ a její gradient. Nyní odvodíme diferenciální rovnici pro výpočet konjugovaného vektoru $\overline{\mathbf{p}}$. Vektor $\overline{\mathbf{p}} = \overline{\mathbf{p}}(\overline{\mathbf{x}}(t))$ závisí na zobecněném stavu $\overline{\mathbf{x}}$, který je funkcí času. Hledejme tedy vztah pro derivaci vektoru $\overline{\mathbf{p}}$ podle času. Z (9.26) platí

$$\frac{d\overline{p}_i}{dt} = -\frac{d}{dt}\left(\frac{\partial \overline{V}}{\partial x_i}\right) = -\sum_{j=0}^{n+1} \frac{\partial}{\partial x_j}\left(\frac{\partial \overline{V}}{\partial x_i}\right)\frac{dx_j}{dt} = -\sum_{j=0}^{n+1} \frac{\partial^2 \overline{V}}{\partial x_i \partial x_j}\overline{f}_j, \tag{9.27}$$

pro $i = 1, \ldots, (n+1)$. Souřadnice $\overline{p}_0$ je podle (9.20) rovna (-1) a proto $\dfrac{d\overline{p}_0}{dt} = 0$.

Podle principu maxima optimální řízení $\mathbf{u}^*(t)$ v každém okamžiku maximalizuje Hamiltonián. Jsme-li v čase t ve stavu $\overline{\mathbf{x}}$, je principem maxima určeno optimální řízení. Je zřejmé, že změníme-li v čase t stav $\overline{\mathbf{x}}$, řízení $\mathbf{u}^*(t)$ již nebude optimální pro změněný stav. Principem maxima můžeme spočíst jinou hodnotu optimálního řízení. Z této úvahy plyne, že derivace Hamiltonovy funkce $\overline{H}$ podle $\overline{\mathbf{x}}$ jsou na optimální trajektorii nulové. Platí tedy

$$\frac{\partial}{\partial x_i}\overline{H}\left(\overline{\mathbf{x}}, \overline{\mathbf{p}}(\overline{\mathbf{x}}), \mathbf{u}^*\right) = 0, \qquad i = 1, 2, \ldots, (n+1)$$

Zde je Hamiltonián chápán jako funkce $\overline{H}\left(\overline{\mathbf{x}}, \overline{\mathbf{p}}(\overline{\mathbf{x}}), \mathbf{u}^*\right) = \overline{H}\left(\overline{\mathbf{x}}, \mathbf{u}^*\right)$. Po dosazení za Hamiltonovu funkci $\overline{H}$ z (9.23) a (9.26) dostaneme

$$\frac{\partial}{\partial x_i}\left(-\sum_{j=0}^{n+1}\frac{\partial \overline{V}}{\partial x_j}\overline{f}_j\right) = -\sum_{j=0}^{n+1}\frac{\partial^2 \overline{V}}{\partial x_i \partial x_j}\overline{f}_j - \sum_{j=0}^{n+1}\frac{\partial \overline{V}}{\partial x_j}\frac{\partial \overline{f}_j}{\partial x_i} = 0, \quad i = 1, \ldots, (n+1)$$

Z předchozího výrazu plyne

$$-\sum_{j=0}^{n+1}\frac{\partial^2 \overline{V}}{\partial x_i \partial x_j}\overline{f}_j = \sum_{j=0}^{n+1}\frac{\partial \overline{V}}{\partial x_j}\frac{\partial \overline{f}_j}{\partial x_i}, \quad i = 1, \ldots, (n+1)$$

Upravíme-li (9.27) podle předchozího výrazu, dostaneme pro konjugovaný vektor $\overline{\mathbf{p}}$ diferenciální rovnici ve tvaru

$$\frac{d\overline{p}_i}{dt} = \sum_{j=0}^{n+1}\frac{\partial \overline{V}}{\partial x_j}\frac{\partial \overline{f}_j}{\partial x_i} = \sum_{j=0}^{n+1}\overline{p}_j\frac{\partial \overline{f}_j}{\partial x_i}, \quad i = 1, \ldots, (n+1) \tag{9.28}$$

Diferenciální rovnice (9.28) spolu s diferenciální rovnicí $\frac{d\overline{p}_0}{dt} = 0$ tvoří soustavu konjugovaných rovnic určující změnu konjugovaného vektoru na optimální trajektorii - jsou to rovnice tzv. **konjugovaného systému**. Soustavu (9.28) můžeme ještě dále upravit. Vyjádříme-li $\overline{H} = \overline{H}(\overline{\mathbf{x}}, \overline{\mathbf{p}}, \mathbf{u})$, pak platí pro parciální derivaci $\overline{H}$ podle x_i

$$\frac{\partial \overline{H}}{\partial x_i} = \frac{\partial}{\partial x_i}\left(\sum_{j=0}^{n+1} \overline{p}_j \overline{f}_j\right) = \sum_{j=0}^{n+1} \overline{p}_j \frac{\partial \overline{f}_j}{\partial x_i}$$

Dosadíme-li z předchozí rovnice do (9.28), je možno diferenciální rovnici konjugovaného systému zapsat ve tvaru

$$\frac{d\overline{p}_i}{dt} = -\frac{\partial \overline{H}}{\partial x_i}, \qquad i = 0, 1, \ldots, (n+1) \tag{9.29}$$

Princip maxima tedy vede na podmínkové rovnice

$$\begin{aligned} \frac{d\overline{\mathbf{x}}}{dt} &= \left(\frac{\partial \overline{H}}{\partial \overline{\mathbf{p}}}\right)^T \\ \mathbf{u}^*(t) &= \arg\max_{\mathbf{u}\in\mathbf{U}} \overline{H}(\overline{\mathbf{x}}, \overline{\mathbf{p}}, \mathbf{u}) \\ \frac{d\overline{\mathbf{p}}}{dt} &= -\left(\frac{\partial \overline{H}}{\partial \overline{\mathbf{x}}}\right)^T \end{aligned} \tag{9.30}$$

První rovnice v předchozí soustavě je stavová rovnice systému, neboť $\overline{H} = \overline{\mathbf{p}}^T \overline{\mathbf{f}}$, druhá rovnice je Pontrjaginův princip maxima a třetí rovnice je stavová rovnice konjugovaného systému.

Z okrajové podmínky $V(\mathbf{x}, t_1) = h(\mathbf{x}(t_1))$ Bellmanovy rovnice (9.19) plynou koncové podmínky diferenciální rovnice konjugovaného systému (9.29) či (9.30c). Pro konjugovaný vektor $\overline{\mathbf{p}}$ podle (9.20) plyne koncová podmínka

$$\overline{p}_0(t_1) = -1, \quad \overline{p}_{n+1}(t_1) = 0, \quad \overline{p}_i(t_1) = -\frac{\partial h(\mathbf{x}(t_1)}{\partial x_i}, \quad i = 1, \ldots, n. \tag{9.31}$$

Koncová podmínka (9.31) platí pro problém optimálního řízení s volným koncovým stavem a pevným koncovým časem t_1. Počáteční podmínky systému (9.30a) jsou podle (9.15), (9.14) a (9.13)

$$x_0(t_0) = 0, \quad x_{n+1}(t_0) = t_0, \quad x_i(t_0) = x_{i0}, \quad i = 1, \ldots, n. \tag{9.32}$$

Předpoklad o diferencovatelnosti funkce $V(\mathbf{x}, t)$, který provázel naše odvození, není obecně nutný.

Na závěr tohoto odstavce uveďme formulaci principu maxima:

Věta: Princip maxima.
Mějme problém optimálního řízení (9.13). Aby řízení $\mathbf{u}^*(t) \in \mathbf{U}$ bylo optimální řízení minimalizující kritérium kvality řízení, musí platit, že existuje nenulový vektor $\overline{\mathbf{p}}(t)$, vyhovující diferenciální rovnici (9.29) či (9.30c) s okrajovou podmínkou (9.31), kde Hamiltonova

funkce $\overline{H}$ je určena dle (9.23). Podle (9.24) či (9.30b) platí, že v každém časovém okamžiku t je Hamiltonova funkce maximální vzhledem k řízení $\mathbf{u}(t)$ a její maximum je rovno nule.

□

K této formulaci principu maxima připojíme několik poznámek:

1. Je-li pevný koncový bod trajektorie $\mathbf{x}(t_1)$, potom nejsou určeny okrajové podmínky (9.31) konjugovaného systému (9.29) či (9.30c). Místo nich známe koncové podmínky $\mathbf{x}(t_1)$ systému (9.30a).

2. Je-li řízený systém v (9.13) stacionární (časově invariantní), to znamená, že tvořící funkce $\mathbf{f}$ není explicitně závislá na čase, a také funkce g v kritériu (9.13) nezávisí explicitně na čase, není třeba zavádět novou souřadnici x_{n+1} podle (9.14). Optimalizační problém řešíme potom pouze v $(n+1)$-rozměrném prostoru stavů $\overline{\mathbf{x}}$ i $(n+1)$-rozměrném prostoru konjugovaných stavů $\overline{\mathbf{p}}$.

3. Pro problém časově optimálního řízení, ve kterém je kritérium

$$J = \int_{t_0}^{t_1} 1(t)\ dt = t_1 - t_0, \tag{9.33}$$

je jádro funkcionálu $g(\mathbf{x}, \mathbf{u}, t) = 1$ a proto není ani třeba zavádět nultou souřadnici $x_0(t)$ podle (9.14). Pro stacionární systém optimalizační problém řešíme potom pouze v n-rozměrném prostoru stavů $\mathbf{x}(t)$ a n-rozměrném prostoru konjugovaných stavů $\mathbf{p}(t)$ - pruhy nad $\mathbf{x}$, $\mathbf{p}$ i H můžeme tedy vynechat.

4. Hamiltonovu funkci můžeme definovat

$$H = \sum_{i=1}^{n} p_i f_i - g. \tag{9.34}$$

Pro stacionární systém i kritérium můžeme potom optimalizační problém řešit v n-rozměrném prostoru stavů systému i konjugovaného systému, neboť uměle zavedená složka $p_0(t)$ je konstantní a rovna (-1) a funkce $x_0(t)$ se v Hamiltoniánu explicitně nevyskytuje.

5. Protože neznáme počáteční podmínky konjugovaného systému, je třeba řešit okrajovou úlohu pro systém diferenciálních rovnic (9.30a) a (9.30c). Princip maxima převádí tedy problém optimalizace dynamických systémů na maximalizaci skalární Hamiltonovy funkce a okrajovou úlohu pro soustavy diferenciálních rovnic (9.30a) a (9.30c), které popisují dynamické vlastnosti daného systému a tzv. konjugovaného systému.

6. V kapitole o variačních metodách byl uveden kanonický tvar Eulerovy - Lagrangeovy rovnice. Tam se také vyskytovala Hamiltonova funkce H a konjugovaný vektor (který tam byl značen $\boldsymbol{\lambda}$). Tam jsme požadovali, aby na optimální trajektorii (extremále) platilo $\frac{\partial H}{\partial \mathbf{u}} = 0$. To platí v případě, že řízení $\mathbf{u}(t)$ není omezeno. Požadavek nulovosti derivace Hamiltoniánu vůči řízení nahrazuje princip maxima požadavkem maximalizace Hamiltonovy funkce vzhledem k řízení. Při tom řízení může být omezeno.

7. Je-li systém lineární vůči řízení $\mathbf{u}(t)$ a v kritériu kvality řízení je jádro funkcionálu $g(\mathbf{x}, \mathbf{u}, t)$ také lineární vůči řízení $\mathbf{u}(t)$, je i Hamiltonián $H(\mathbf{x}, \mathbf{p}, \mathbf{u})$ lineární vůči řízení $\mathbf{u}(t)$. Maximum lineární formy nastává vždy na hranici oblasti dovolených hodnot řízení $\mathbf{U}$. Je-li tedy omezena maximální i minimální hodnota složek řídicího vektoru, pak optimální řízení nabývá pouze maximální nebo minimální hodnotu - říkáme, že optimální řízení je typu "bang - bang". Okamžiky změny hodnot řídicích veličin z jedné krajní hodnoty na druhou nazýváme "okamžiky přepnutí".

9.3 Nutná podmínka optimality - princip maxima

V tomto odstavci odvodíme princip maxima jako nutnou podmínku pro optimální řízení. Nebudeme vycházet z dynamického programování, neboť Bellmanova rovnice je postačující podmínkou pro existenci optimálního řízení. Uvažujme optimalizační problém s volným koncem a pevným koncovým časem - viz problém (9.13). Funkce $h(\mathbf{x}(t_1))$ v kritériu optimality nechť je rovna nule.

Stejně jako v předchozím odstavci budeme optimalizační problém řešit v $(n + 2)$ rozměrném prostoru. Pro jednoduchost zápisu, na rozdíl od předchozího odstavce, vynecháme pruhy nad $(n + 2)$ rozměrným vektorem $\mathbf{x}$ i $\mathbf{p}$. Podle (9.13), (9.14) a (9.15) je rozšířený systém popsán rovnicí

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, \mathbf{u}), \qquad \mathbf{x} = [x_0, x_1, \ldots, x_n, x_{n+1}]^T \tag{9.35}$$

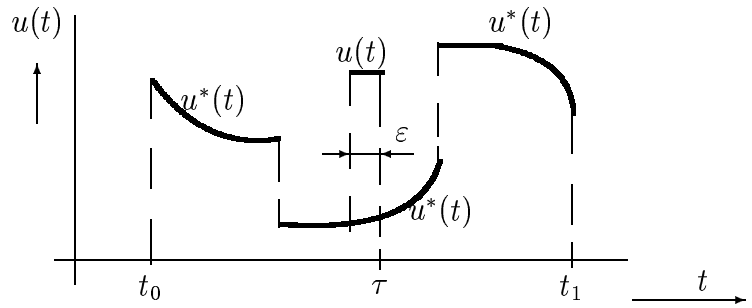
Podle (9.16) minimalizujeme kritérium

$$J(\mathbf{u}) = x_0(t_1), \tag{9.36}$$

neboť podle předpokladu je neintegrální člen $h(\mathbf{x}(t_1))$ v kritériu nulový. O funkci $\mathbf{f}$ v (9.35) budeme předpokládat, že je ohraničená a spojitá vzhledem ke svým argumentům a diferencovatelná podle $\mathbf{x}$. Řízení $\mathbf{u}(t) \in \mathbf{U}$ nechť je po částech spojitá funkce s konečným počtem nespojitostí prvního druhu. Pro přehlednost odvození předpokládejme jedinou řídicí veličinu.

Myšlenka odvození je následující:
Předpokládejme, že jsme nalezli optimální řízení $u^*(t)$ i optimální trajektorii $\mathbf{x}^*(t)$. Porušíme-li nějakým způsobem optimální řízení $u^*(t)$, kritérium se ze své minimální hodnoty může jenom zvětšit (lépe - nemůže se zmenšit). Nezápornost přírůstku kritéria, při změně řízení od optimální hodnoty, nás povede k principu maxima.

Porušení optimálního řízení provedeme zde speciálním způsobem. Na nekonečně malém časovém intervalu $\tau - \varepsilon < t < \tau$, kde ε je nekonečně malá veličina a libovolný časový okamžik τ, $t_0 < \tau \leq t_1$, dopustíme libovolně velkou (ale přípustnou) změnu řízení $u(t)$ od optimálního průběhu $u^*(t)$. Na ostatním intervalu $t \in [t_0, \tau - \varepsilon]$ a $t \in [\tau, t_1]$ zůstane řízení optimální. Této změně optimálního řízení říkáme **"jehlová variace"** - viz obr. 9.1. Vyšetříme vliv této jehlové variace na trajektorii a kritérium. Rozdíl mezi optimální trajektorií $\mathbf{x}^*(t)$ a trajektorií $\mathbf{x}(t)$, odpovídající neoptimálnímu řízení $u(t)$, bude v čase τ



Obrázek 9.1: Jehlová variace optimálního řízení $\mathbf{u}^*(t)$.

úměrný ε a s přesností na nekonečně malé veličiny druhého řádu platí

$$\mathbf{x}(\tau) - \mathbf{x}^*(\tau) = \varepsilon \left(\frac{d\mathbf{x}}{dt} - \frac{d\mathbf{x}^*}{dt} \right)_{t=\tau} = \varepsilon \left[\mathbf{f}(\mathbf{x}(\tau), \mathbf{u}(\tau)) - \mathbf{f}(\mathbf{x}^*(\tau), \mathbf{u}^*(\tau)) \right] \tag{9.37}$$

Rozmyslete si podrobně předchozí vztah. Spíše bychom v předchozím vztahu čekali, že derivaci stavů budeme uvažovat v čase $t = \tau - \varepsilon$. Rozdíl je ale nekonečně malá veličina druhého řádu. Vztah (9.37) je pro odvození principu maxima klíčový.

Pro $t \geq \tau$ je řízení již optimální, ale vlivem jehlové změny řízení je variace trajektorie $\delta\mathbf{x}(t) = \mathbf{x}(t) - \mathbf{x}^*(t)$ nenulová i pro čas t, kde $\tau \leq t \leq t_1$. Nalezneme linearizovaný vztah pro variaci $\delta\mathbf{x}(t)$. Z rovnice systému $\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, \mathbf{u})$ dostaneme pro přírůstek $\delta\mathbf{x}$ vztah

$$\frac{d(\mathbf{x} + \delta\mathbf{x})}{dt} = \mathbf{f}(\mathbf{x} + \delta\mathbf{x}, \mathbf{u}) = \mathbf{f}(\mathbf{x}, \mathbf{u}) + \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \delta\mathbf{x} + \varepsilon(\delta\mathbf{x}),$$

kde $\varepsilon(\delta\mathbf{x})$ je nekonečně malá veličina alespoň druhého řádu. Odtud plyne lineární rovnice pro přírůstek $\delta\mathbf{x}$

$$\frac{d(\delta\mathbf{x})}{dt} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \delta\mathbf{x}, \tag{9.38}$$

neboli po složkách

$$\frac{d(\delta x_i)}{dt} = \sum_{j=0}^{n+1} \frac{\partial f_i}{\partial x_j} \delta x_j, \qquad i = 0, 1, \ldots, (n+1). \tag{9.39}$$

Počáteční podmínky soustavy (9.38) jsou určeny podle (9.37). Variace kritéria je podle (9.36) rovna

$$\delta J(\mathbf{u}^*) = \delta x_0(t_1) \geq 0 \tag{9.40}$$

Variace je nezáporná, protože optimální řízení $\mathbf{u}^*(t)$ zajišťuje minimální hodnotu kritéria. Při neoptimálním řízení $\mathbf{u}(t)$ (s jehlovou variací) nemůže hodnota kritéria klesnout.

Nyní uděláme jeden ze dvou umělých kroků v odvození principu maxima. Výraz (9.40) zapíšeme ve tvaru

$$\delta J(\mathbf{u}^*) = \delta x_0(t_1) = -\delta\mathbf{x}^T(t_1)\mathbf{p}(t_1) \geq 0, \tag{9.41}$$

kde zavedený $(n+2)$-rozměrný vektor $\mathbf{p}(t_1)$ je roven

$$\mathbf{p}(t_1) = [-1, 0, \ldots, 0]^T. \tag{9.42}$$

Podmínka pro to, aby $\mathbf{u}^*(t)$ bylo optimální řízení je dána vztahem (9.40) nebo (9.41), kde variace $\delta\mathbf{x}$ je určena rovnicí (9.38) resp. (9.39) s počáteční podmínkou (9.37). Abychom vliv jehlové variace řízení nemuseli vyšetřovat až v koncovém čase t_1, je třeba udělat druhý umělý krok.

Budeme hledat vektor $\mathbf{p}(t)$ takový, abychom variaci kritéria podle (9.41) nemuseli zjišťovat až v čase t_1, ale v libovolném čase $\tau \le t \le t_1$. Chceme tedy, aby platilo

$$-\delta J(\mathbf{u}^*) = \delta\mathbf{x}^T(t_1)\mathbf{p}(t_1) = \delta\mathbf{x}^T(t)\mathbf{p}(t), \quad \tau \le t \le t_1. \tag{9.43}$$

Potom vliv jehlové změny řízení $\mathbf{u}^*(t)$ na kritérium kvality řízení zjistíme bezprostředně v čase τ, kdy jehlová variace skončila.

Nyní zbývá nalézt diferenciální rovnici pro vektor $\mathbf{p}(t)$. Ze vztahu (9.43) plyne $\delta\mathbf{x}^T(t)\mathbf{p}(t) = konst.$ pro $\tau \le t \le t_1$. Odtud tedy

$$\frac{d}{dt}\left(\delta\mathbf{x}^T(t)\mathbf{p}(t)\right) = 0, \qquad \tau \le t \le t_1$$

Provedeme derivaci skalárního součinu, pak

$$\left(\frac{d(\delta\mathbf{x}(t))}{dt}\right)^T \mathbf{p}(t) + \delta\mathbf{x}^T(t)\frac{d\mathbf{p}(t)}{dt} = 0, \qquad \tau \le t \le t_1.$$

Předchozí rovnici vyjádříme po složkách

$$\sum_{i=0}^{n+1} \frac{d(\delta x_i(t))}{dt} p_i(t) + \sum_{j=0}^{n+1} \delta x_j(t)\frac{dp_j(t)}{dt} = 0, \qquad \tau \le t \le t_1.$$

Do předchozího výrazu dosadíme z (9.39), pak

$$\sum_{i=0}^{n+1} p_i(t) \sum_{j=0}^{n+1} \frac{\partial f_i}{\partial x_j}\delta x_j(t) + \sum_{j=0}^{n+1} \delta x_j(t)\frac{dp_j(t)}{dt} = 0.$$

Vytkneme variaci δx_j z obou členů předchozí rovnice a dostaneme

$$\sum_{j=0}^{n+1} \delta x_j(t) \left[\frac{dp_j(t)}{dt} + \sum_{i=0}^{n+1} \frac{\partial f_i}{\partial x_j} p_i(t)\right] = 0.$$

Protože variace $\delta x_j(t)$ je obecně nenulová, musí být výraz v hranaté závorce roven nule. Odtud plyne diferenciální rovnice pro konjugovaný vektor $\mathbf{p}(t)$

$$\frac{dp_j(t)}{dt} = -\sum_{i=0}^{n+1} \frac{\partial f_i(\mathbf{x},\mathbf{u})}{\partial x_j} p_i(t), \quad j = 0, 1, \ldots, (n+1) \tag{9.44}$$

Vztah (9.44) je diferenciální rovnice tzv. **konjugovaného systému**, která je shodná s rovnicí (9.29) odvozenou v předchozím odstavci.

Ze vztahu (9.43) plyne, že přírůstek kritéria δJ můžeme vyšetřovat okamžitě v čase τ. Platí tedy

$$-\delta J = \delta\mathbf{x}^T(\tau)\mathbf{p}(\tau) \le 0 \tag{9.45}$$

Podle (9.39) a (9.37) je variace $\delta\mathbf{x}(\tau)$ úměrná rozdílu rychlostí a proto

$$\left[\mathbf{f}\left(\mathbf{x}(\tau),\mathbf{u}(\tau)\right)\right]^T\mathbf{p}(\tau)-\left[\mathbf{f}\left(\mathbf{x}^*(\tau),\mathbf{u}^*(\tau)\right)\right]^T\mathbf{p}(\tau)\leq 0. \tag{9.46}$$

Definujeme si Hamiltonovu funkci $H\left(\mathbf{x},\mathbf{p},\mathbf{u}\right)$

$$H\left(\mathbf{x},\mathbf{p},\mathbf{u}\right)=\left[\mathbf{f}\left(\mathbf{x}(\tau),\mathbf{u}(\tau)\right)\right]^T\mathbf{p}(\tau), \tag{9.47}$$

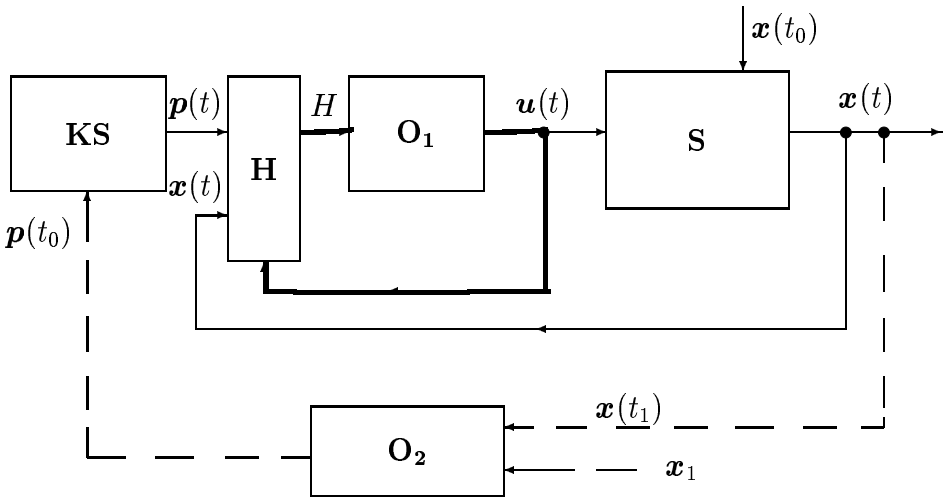
pak z (9.46) je zřejmé, že při optimálním řízení je na optimální trajektorii Hamiltonova funkce maximální.

Tímto postupem jsme odvodili princip maxima jako nutnou podmínku řešení problému optimálního řízení spojitého systému s omezením řízení.

9.4 Řešení některých problémů optimálního řízení principem maxima

9.4.1 Obecný postup řešení

Podle principu maxima optimální řízení optimalizuje Hamiltonián. Přitom je třeba řešit diferenciální rovnici systému a tzv. konjugovaného systému. Počáteční podmínky systému obvykle známe, ale neznáme počáteční podmínku konjugovaného systému. V úloze s volným koncem trajektorie známe koncovou podmínku konjugovaného systému a v úloze s pevným koncem trajektorie známe naopak koncovou podmínku systému. Ideové schéma řešení problému optimálního řízení s pevným koncem trajektorie je uvedeno na obr. 9.2.



Obrázek 9.2: Schéma výpočtu optimálního řízení a optimální trajektorie podle principu maxima.

Na obr. 9.2 je S řízený systém, KS konjugovaný systém, blok H vytváří Hamiltonovu funkci H. Optimizátor O_1 určuje optimální řídicí veličinu tak, aby Hamiltonián byl v každém časovém okamžiku maximální vzhledem k přípustnému řízení $\mathbf{u}(t)\in\mathbf{U}$. Optimizátor O_1 musí být velmi rychlý. Rychlá smyčka s optimizátorem O_1, sloužící k výpočtu

optimálního řízení, je na obr. 9.2 vyznačena silnou čarou. Často lze optimální řízení z tvaru Hamiltoniánu a omezení $\mathbf{u}(t) \in \mathbf{U}$ analyticky vypočítat.

Optimizátor O_2 určuje počáteční podmínku konjugovaného systému. V koncovém čase t_1 skutečnou hodnotu stavu $\mathbf{x}(t_1)$ systému S porovnáme s požadovaným koncovým stavem $\mathbf{x}_1$. Není-li skutečný koncový stav $\mathbf{x}(t_1)$ totožný s požadovaným koncovým stavem $\mathbf{x}_1$, změníme optimizátorem O_2 počáteční podmínku $\mathbf{p}(t_0)$ konjugovaného systému a v následující iteraci počítáme znovu optimální řízení a optimální trajektorii maximalizací Hamiltonovy funkce. Optimální postup hledání počáteční podmínky $\mathbf{p}(t_0)$ konjugovaného systému optimizátorem O_2 není obecně známý.

Přitom je optimální trajektorie velmi citlivá na volbu počáteční podmínky $\mathbf{p}(t_0)$ konjugovaného systému. Konvergence iteračního postupu hledání $\mathbf{p}(t_0)$ je zaručena pouze pro některé speciální třídy problémů optimalizace - například pro časově optimální řízení lineárního systému, nebo kvadraticky optimální řízení lineárního systému.

Jiný způsob iteračního výpočtu optimálního řízení podle principu maxima je následující:

Mějme problém optimálního řízení s volným koncem trajektorie a pevným koncovým časem. Označíme $\mathbf{u}^{(i)}(t)$ i-tou iteraci výpočtu optimálního řízení.

Zvolíme první odhad optimálního řízení $\mathbf{u}^{(1)}(t)$ a vypočteme trajektorii systému vycházející z dané počáteční podmínky $\mathbf{x}(t_0)$. Pro zvolené řízení $\mathbf{u}^{(1)}(t)$ a odpovídající řešení $\mathbf{x}(t)$ vypočteme řešení diferenciální rovnice konjugovaného systému v obráceném čase vycházející z koncové podmínky $\mathbf{p}(t_1) = 0$ (volný konec trajektorie).

Z výrazu pro Hamiltonovu funkci $H = \sum_{i=1}^{n} p_i f_i(\mathbf{x}, \mathbf{u}) - g(\mathbf{x}, \mathbf{u})$ (viz. (9.34)), můžeme vypočíst derivaci Hamiltonovy funkce $\mathrm{grad}_u H = \left(\frac{\partial H}{\partial \mathbf{u}}\right)^T$.

Uvnitř dovolené oblasti řízení je gradient kritéria vzhledem k řízení úměrný záporně vzatému gradientu Hamiltoniánu vůči řízení. Toto tvrzení snadno prokážeme, neboť podle (9.43) platí $\delta_u J = -\delta\mathbf{x}^T(t)\mathbf{p}(t)$ a proto

$$\begin{aligned}
\delta_u J &= \frac{\partial J}{\partial \mathbf{u}}\delta\mathbf{u} = -\delta\mathbf{x}^T(t)\mathbf{p}(t) = -\left(\mathbf{x} - \mathbf{x}^*\right)^T \mathbf{p} \\
&= -\varepsilon \left(\mathbf{f}(\mathbf{x}, \mathbf{u}) - \mathbf{f}(\mathbf{x}^*, \mathbf{u}^*)\right)^T \mathbf{p} = -\varepsilon \left[H(\mathbf{x}, \mathbf{u}, \mathbf{p}) - H(\mathbf{x}, \mathbf{u}^*, \mathbf{p})\right] \\
&= -\frac{\partial H}{\partial \mathbf{u}}\delta\mathbf{u}\ \varepsilon
\end{aligned}$$

Proto platí

$$\mathrm{grad}_u J = -\mathrm{grad}_u H. \tag{9.48}$$

Nový odhad optimálního řízení $\mathbf{u}^{(i+1)}(t)$ dostaneme podle iteračního předpisu

$$\mathbf{u}^{(i+1)}(t) = \begin{cases} \mathbf{u}^{(i)}(t) + \alpha\ \mathrm{grad}_u H & \text{pro} \quad \mathbf{u}^{(i)}(t) + \alpha\ \mathrm{grad}_u H \in \mathbf{U} \\ \mathbf{u}^{(i)}(t) & \text{pro} \quad \mathbf{u}^{(i)}(t) + \alpha\ \mathrm{grad}_u H \notin \mathbf{U} \end{cases} \tag{9.49}$$

kde α je vhodně volený koeficient. Jeho volba je vlastně algoritmus jednorozměrového hledání. Můžeme použít libovolný algoritmus, případně použít i jinou metodu - například metodu konjugovaných gradientů. Optimální řízení je určeno limitou

$$\mathbf{u}^*(t) = \lim_{i \to \infty} \mathbf{u}^{(i)}(t). \tag{9.50}$$

Výhodou této přímé metody výpočtu optimálního řízení je, že v každé iteraci máme přípustné řešení. Iterační výpočet ukončíme, je-li gradient Hamiltoniánu malý, případně se řízení dle (9.49) již nemění.

V reálných problémech optimalizace existují i omezení na stavy systému $\mathbf{x}(t) \in \mathbf{X} \subset R^n$. Tyto problémy se principem maxima řeší obtížně, lze je iteračně řešit pomocí pokutových či barierových funkcí zahrnutých v kritériu optimality.

Oblíbená a velmi efektivní metoda, která umožňuje respektovat omezení na stavy systému, je tzv. metoda "několikanásobného nástřelu" - **multiple shooting method**. V této metodě se interval řízení rozdělí na několik subintervalů a řeší se problém optimálního řízení na každém intervalu zvlášť.

Při tom se řídicí veličina na každém intervalu vhodně aproximuje, například po částech konstantní funkcí, nebo po částech lineární funkcí a pod.. Tím převedeme problém optimálního řízení na problém matematického programování - hledání koeficientů aproximační funkce řízení.

Nyní musíme respektovat hlavní myšlenku metody. Koncový stav na každém intervalu musí být přípustný a navíc musí být shodný s počátečním stavem na následujícím intervalu. Tyto omezující podmínky tvoří omezení problému nelineárního programování. Na jeho řešení můžeme použít libovolnou numerickou metodu nelineárního programování s omezením, na příklad metodu SQP (sekvenčního kvadratického programování) nebo metodu IPM (metodu vnitřního bodu).

Omezení na stavy respektujeme pouze v krajních bodech intervalů, na které jsme rozdělili celou dobu řízení. Proto můžeme sledovat a vhodně omezovat i derivace stavů v krajních bodech intervalů, abychom uvnitř zvolených intervalů nepřekročili omezení.

Při řešení optimalizačního problému principem maxima může nastat případ, že Hamiltonián není závislý na řízení a tudíž určení optimálního řízení maximalizací Hamiltoniánu nelze provést. Nastává tzv. **singulární případ**, při kterém nutné podmínky prvního řádu nedají řešení. Proto je nutno vyšetřovat podmínky druhého řádu (druhé a vyšší variace). Zde se singulárními řešeními nebudeme zabývat.

9.4.2 Časově optimální řízení

Pro lineární stacionární systém popsaný stavovou rovnicí $\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$ hledáme takové řízení $\mathbf{u}(t) \in \mathbf{U}$, aby doba přechodu z daného počátečního stavu $\mathbf{x}(t_0) = \mathbf{x}_0$ do daného koncového stavu $\mathbf{x}(t_1) = \mathbf{x}_1$ byla minimální. Kritériem je tedy doba přechodu

$$J(\mathbf{u}(t)) = \int_{t_0}^{t_1} 1(t)\ dt = t_1 - t_0 \tag{9.51}$$

Řízení $\mathbf{u}(t)$ je omezeno v každé složce a platí $(U_i)_{min} \leq u_i(t) \leq (U_i)_{max}$. Podle (9.34) si zavedeme Hamiltonovu funkci

$$H\left(\mathbf{x}, \mathbf{p}, \mathbf{u}\right) = \left(\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}\right)^T \mathbf{p} - 1 \tag{9.52}$$

Konstanta (-1) je v Hamiltoniánu proto, že podle tvaru kritéria (9.51) je $g = 1$ a $p_0(t) = -1$. Na výpočet optimálního řízení nemá konstanta (-1) vliv. Hamiltonián je

lineární vzhledem k řízení $\mathbf{u}(t)$ a proto optimální řízení leží na hranici oblasti. Maximum Hamiltoniánu podle (9.52) nastane, bude-li maximální výraz

$$(\mathbf{Bu})^T \mathbf{p} = \mathbf{u}^T \mathbf{B}^T \mathbf{p} = \mathbf{p}^T \mathbf{B}\mathbf{u} = \left(\mathbf{B}^T \mathbf{p}\right)^T \mathbf{u}$$

Odtud zřejmě plyne optimální řízení

$$u_i^*(t) = \begin{cases} (U_i)_{max} & \text{pro} \quad \sum_{j=1}^n b_{ji} p_j(t) > 0 \\ (U_i)_{min} & \text{pro} \quad \sum_{j=1}^n b_{ji} p_j(t) < 0 \end{cases} \tag{9.53}$$

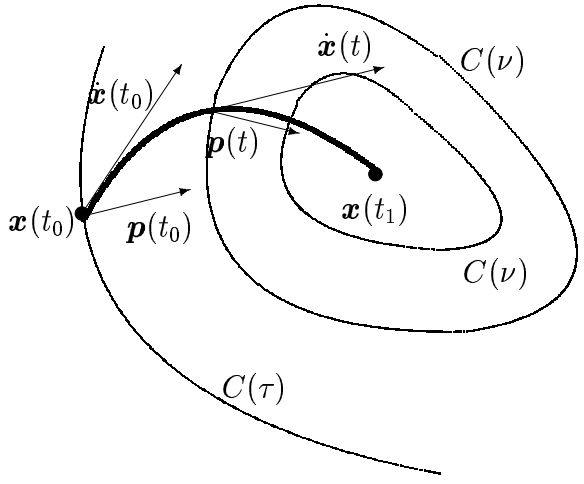
kde b_{ji} jsou prvky matice $\mathbf{B}$. Pokud $\sum_{j=1}^n b_{ji} p_j(t) = 0$, není optimální řízení definováno. Pokud je systém řiditelný každou složkou řídicí veličiny, pak to nastává v izolovaných časových okamžicích, ve kterých je přepnutí řídicí veličiny z jedné krajní polohy na druhou.

Rovnice konjugovaného systému odvodíme z $\dfrac{d\mathbf{p}}{dt} = -\left(\dfrac{\partial H}{\partial \mathbf{x}}\right)^T$, kde H je dle (9.52). Odtud

$$\frac{d\mathbf{p}}{dt} = -\mathbf{A}^T \mathbf{p} \tag{9.54}$$

Matice $-\mathbf{A}^T$ konjugovaného systému má vlastní čísla $-\lambda_A$, kde λ_A jsou vlastní čísla matice systému $\mathbf{A}$. Proto, bude-li původní systém stabilní, bude konjugovaný systém nestabilní.

Ve stavovém prostoru nalezneme množinu $C(\tau)$ bodů, ze kterých se lze dostat za čas $\tau = t_1 - t_0$ do koncového stavu $\mathbf{x}(t_1) = \mathbf{x}_1$ přípustným řízením $\mathbf{u}(t) \in \mathbf{U}$. Lze ukázat, že při optimálním řízení je počáteční stav $\mathbf{x}(t_0)$ na hranici množiny $C(\tau)$ a počáteční vektor konjugovaného systému $\mathbf{p}(t_0)$ je vnitřní normálový vektor množiny $C(\tau)$ - viz obr. 9.3.



Obrázek 9.3: Množina $C(\tau)$, $\tau = t_1 - t_0$ a množiny $C(\nu)$, $\nu = t_1 - t$, ze kterých se lze za čas τ resp. ν dostat do koncového stavu přípustným řízením.

Nyní uvolníme počáteční čas - místo t_0 budeme počáteční čas značit t. Potom množina $C(\nu)$ je množina bodů, ze kterých se lze za čas $\nu = t_1 - t$ dostat do cíle $\mathbf{x}(t_1)$. Vektor $\mathbf{p}(t)$ na optimální trajektorii je pro všechna t vnitřní normálový vektor hranice množiny $C(\nu)$

- viz obr. 9.3. Je to zřejmé z toho, že $\mathbf{p}(t) = -\text{grad}\, V(\mathbf{x}, t)$ a hranice množiny $C(\nu)$ je určena stavy $\mathbf{x}$, pro které platí $V(\mathbf{x}, t) = V(\mathbf{x}) = \nu$, kde $V(\mathbf{x}, t)$ je Bellmanova optimální funkce.

Navíc je při optimálním řízení skalární součin vektoru stavové rychlosti a vektoru konjugovaného systému po celou dobu řízení konstantní a roven 1 - viz vztah (9.52), kde $\max_u H = 0$.

Z řešení stavové rovnice konjugovaného systému (9.54)

$$\mathbf{p}(t) = e^{\mathbf{A}^T(t-t_0)}\mathbf{p}(t_0)$$

a vztahu (9.53) pro optimální řízení okamžitě plyne **věta o konečném počtu přepnutí**, která tvrdí, že optimální řízení $\mathbf{u}^*(t)$ je po částech konstantní s konečným počtem nespojitostí prvního druhu nastávajících v okamžicích přepnutí.

Navíc, má-li matice systému $\mathbf{A}$ pouze reálná vlastní čísla, je počet přepnutí nejvýše roven $(n-1)$, kde n je řád řízeného systému. Platí tedy **věta o n intervalech**, která tvrdí, že každá složka optimální řídicí veličiny $u_i^*(t)$ má nejvýše n intervalů, na kterých má konstantní hodnotu (počet přepnutí je tedy nejvýše $(n-1)$).

9.5 Diskrétní princip maxima

Také pro problém optimálního řízení diskrétních dynamických systémů platí nutné podmínky principu maxima. Pro diskrétní systémy platí princip maxima obecně ve "slabší" formě a podmínky principu maxima mají obecně pouze lokální charakter.

9.5.1 Podmínky optimálnosti

V tomto odstavci se budeme ještě trochu podrobněji zabývat úlohami statické optimalizace - problémy nelineárního programování. Odvodíme pro ně nutné podmínky, které jsou jakousi obdobou nutných podmínek principu maxima. Mějme tedy problém

$$\min\{f(\mathbf{x}) \ : \ g_j(\mathbf{x}) \leq 0\} \tag{9.55}$$

Omezení $g_j(\mathbf{x}) \leq 0$ určují množinu $\mathbf{X}$ přípustných bodů, mezi nimiž hledáme minimum funkce $f(\mathbf{x})$. Vybereme bod $\mathbf{x}_0 \in \mathbf{X}$ a určíme množinu $\mathcal{J}(\mathbf{x}_0)$ aktivních omezení

$$\mathcal{J}(\mathbf{x}_0) = \{j \ : \ g_j(\mathbf{x}_0) = 0\}$$

Je-li $\mathbf{x}_0$ hraniční bod množiny $\mathbf{X}$, je množina $\mathcal{J}(\mathbf{x}_0)$ neprázdná. V bodě $\mathbf{x}_0$ vypočteme gradienty aktivních omezení

$$\frac{\partial g_j(\mathbf{x}_0)}{\partial \mathbf{x}} = \left[\frac{\partial g_j(\mathbf{x}_0)}{\partial x_1}, \ldots, \frac{\partial g_j(\mathbf{x}_0)}{\partial x_n}\right] \qquad j \in \mathcal{J} \tag{9.56}$$

Budeme hledat podmínky, které musí platit, aby jiný bod $\mathbf{x}$, pro který platí $\mathbf{x} = \mathbf{x}_0 + \varepsilon\delta\mathbf{x}$, $\varepsilon > 0$, byl přípustný. Variace $\delta\mathbf{x}$ je přípustná, svírá-li tupý úhel s gradienty (9.56), tedy

$$\frac{\partial g_j(\mathbf{x}_0)}{\partial \mathbf{x}}\ \delta\mathbf{x} \leq 0, \qquad j \in \mathcal{J}(\mathbf{x}_0) \tag{9.57}$$

Předchozí nerovnost okamžitě plyne z toho, že $g_j(\mathbf{x}_0) \leq 0$ a také $g_j(\mathbf{x}_0 + \varepsilon\delta\mathbf{x}) \leq 0$. Z rozvoje $g_j(\mathbf{x}_0 + \varepsilon\delta\mathbf{x})$ v bodě $\mathbf{x}_0$ plyne (9.57).

Množina všech přípustných variací $\delta\mathbf{x}$, splňujících podmínku (9.57), tvoří **kužel přípustných variací** $K(\mathbf{x}_0)$

$$K(\mathbf{x}_0) = \left\{ \delta\mathbf{x} \ : \ \frac{\partial g_j(\mathbf{x}_0)}{\partial \mathbf{x}} \ \delta\mathbf{x} \leq 0, \ j \in \mathcal{J}(\mathbf{x}_0) \right\} \tag{9.58}$$

Tento kužel je tvořen průnikem poloprostorů tvořených nadrovinami $\frac{\partial g_j(\mathbf{x}_0)}{\partial \mathbf{x}} \ \delta\mathbf{x} = 0, \ j \in \mathcal{J}(\mathbf{x}_0)$ a jeho vrchol leží v bodě $\mathbf{x}_0$. Podmínky regularity vyžadují, aby kužel $K(\mathbf{x}_0)$ měl vnitřní bod a postačující podmínka pro to je lineární nezávislost gradientů (9.56).

Bod $\mathbf{x}^* \in \mathbf{X}$ je řešením problému (9.55), jestliže pro libovolnou variaci $\delta\mathbf{x} \in K(\mathbf{x}^*)$ v tomto bodě platí

$$f(\mathbf{x}^* + \varepsilon\delta\mathbf{x}) \geq f(\mathbf{x}^*), \quad \varepsilon > 0, \quad \delta\mathbf{x} \in K(\mathbf{x}^*) \tag{9.59}$$

Potom vskutku v bodě $\mathbf{x}^*$ je lokální minimum funkce $f(\mathbf{x})$ na množině $\mathbf{X}$. Je-li funkce $f(\mathbf{x})$ v bodě $\mathbf{x}^*$ diferencovatelná, pak platí

$$f(\mathbf{x}^* + \varepsilon\delta\mathbf{x}) = f(\mathbf{x}^*) + \varepsilon \frac{\partial f(\mathbf{x}^*)}{\partial \mathbf{x}} \ \delta\mathbf{x} + O(\varepsilon) \tag{9.60}$$

kde $O(\varepsilon)$ je nekonečně malá veličina druhého řádu. Dosadíme-li (9.60) do (9.59), dostaneme důležité tvrzení, které specifikuje optimální bod $\mathbf{x}^*$:

Je-li $\mathbf{x}^*$ *řešením problému (9.55), pak platí*

$$\frac{\partial f(\mathbf{x}^*)}{\partial \mathbf{x}} \ \delta\mathbf{x} \geq 0 \tag{9.61}$$

pro libovolnou variaci $\delta\mathbf{x}$ *splňující podle (9.57) nerovnost*

$$\frac{\partial g_j(\mathbf{x}^*)}{\partial \mathbf{x}} \ \delta\mathbf{x} \leq 0, \qquad j \in \mathcal{J}(\mathbf{x}^*) \tag{9.62}$$

□

Podmínky (9.61) a (9.62) jsou nutné podmínky. Platí-li v nějakém bodě nerovnosti (9.61) a (9.62), pak tento bod nemusí být řešením problému (9.55), ale naopak v každém bodě $\mathbf{x}$ řešícím (9.55) platí (9.61) a (9.62).

Nerovnost (9.62) je možno zapsat i v jiném tvaru. Z (9.61) plyne

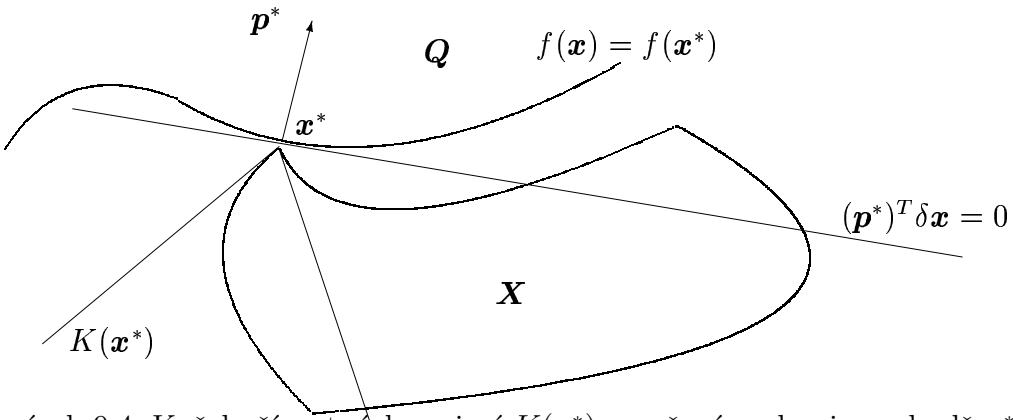
$$\max_{\delta\mathbf{x} \in K(\mathbf{x}^*)} (\mathbf{p}^*)^T \delta\mathbf{x} = 0 \tag{9.63}$$

kde vektor $\mathbf{p}^*$ je roven

$$\mathbf{p}^* = -\left(\frac{\partial f(\mathbf{x}^*)}{\partial \mathbf{x}} \right)^T \tag{9.64}$$

Nadrovina $(\mathbf{p}^*)^T \delta\mathbf{x} = 0$ je opěrná ke kuželi přípustných variací $K(\mathbf{x}^*)$ v jeho vrcholu - viz obr. 9.4. Aby platilo (9.63), je třeba poněkud rozšířit kužel přípustných variací.

Poznámka: Nadrovina $\left\{ \mathbf{x} \ : \ (\mathbf{p}^*)^T \mathbf{x} = (\mathbf{p}^*)^T \mathbf{x}^* = konst. \right\}$ je opěrná k množině $\mathbf{X}$ v bodě $\mathbf{x}^*$, leží-li $\mathbf{X}$ v jednom z poloprostorů určených příslušnou nadrovinou a má s množinou $\mathbf{X}$ alespoň jeden společný bod.



Obrázek 9.4: Kužel přípustných variací $K(\mathbf{x}^*)$ a opěrná nadrovina v bodě $\mathbf{x}^*$.

□

Protože $\delta\mathbf{x} = \mathbf{x} - \mathbf{x}^*$, pak z $(\mathbf{p}^*)^T\delta\mathbf{x} = 0$ plyne

$$(\mathbf{p}^*)^T\mathbf{x} = (\mathbf{p}^*)^T\mathbf{x}^* = konst. \tag{9.65}$$

Množina všech bodů $\mathbf{x}$ splňujících (9.65) je nadrovina procházející bodem $\mathbf{x}^*$, která je kolmá k vektoru $\mathbf{p}^*$ - viz obr. 9.4.

Z toho, že nadrovina (9.65) je opěrná ke kuželi přípustných variací v bodě $\mathbf{x}^* \in \mathbf{X}$ bohužel neplyne, že je také opěrná k celé množině $\mathbf{X}$ - viz obr. 9.4. Aby to platilo, je třeba požaovat, aby množina $\mathbf{X}$ byla konvexní. Platí tedy následující tvrzení:

Je-li $\mathbf{x}^* \in \mathbf{X}$ řešením problému (9.55) a množina $\mathbf{X}$ je konvexní množina, pak v $\mathbf{x}^*$ je maximum lineární funkce $(\mathbf{p}^*)^T\mathbf{x}$, to je

$$\max_{\mathbf{x}\in\mathbf{X}} (\mathbf{p}^*)^T\mathbf{x} = (\mathbf{p}^*)^T\mathbf{x}^* \tag{9.66}$$

kde vektor $\mathbf{p}^*$ je určen podle (9.64). □

Je-li množina $\mathbf{X}$ určena nerovnicemi $g_j(\mathbf{x}) \leq 0$, pak, jsou-li funkce $g_j(\mathbf{x})$ konvexní, je množina $\mathbf{X}$ konvexní množina. Je-li ještě navíc funkce $f(\mathbf{x})$ také konvexní, pak nadrovina (9.65) je opěrná i k množině

$$\mathbf{Q} = \{\ \mathbf{x}\ :\ f(\mathbf{x}) \leq f(\mathbf{x}^*)\} \tag{9.67}$$

Na body $\mathbf{x}$ množiny $\mathbf{Q}$ se nekladou žádná omezení. Množina $\mathbf{Q}$ má s množinou $\mathbf{X}$ alespoň jeden společný bod $\mathbf{x}^*$. Množina $\mathbf{Q}$ je vyznačena na obr. 9.4, kde však nadrovina $(\mathbf{p}^*)^T\delta\mathbf{x} = 0$ k ní není opěrná. Je-li funkce $f(\mathbf{x})$ konvexní funkce a množina $\mathbf{X}$ je konvexní, pak nerovnosti (9.61) a (9.62) resp. (9.63) a (9.64) jsou postačující pro to, aby bod $\mathbf{x}^*$ byl řešením problému (9.55).

Z podmínek, které jsme zde odvodili, plyne obecný postup určení optimálního řešení problému (9.55). Vybereme libovolný bod $\mathbf{x}_0 \in \mathbf{X}$ a najdeme jiný blízký bod $\mathbf{x}_1 \in \mathbf{X}$ takový, aby platilo $f(\mathbf{x}_1) < f(\mathbf{x}_0)$. To opakujeme tak dlouho, až v určitém bodě nelze nalézt přípustnou variaci, která zlepší (zmenší) kritérium. Tímto postupem nalezneme lokální minimum. Tento obecný postup určuje tzv. **přímé metody**. Přímé metody se také často nazývají gradientní nebo také metody přípustných směrů. Takovými metodami

dostaneme monotónně klesající posloupnost přípustných řešení, až konečně nalezneme lokální minimum.

Existuje i jiný postup řešení. Předpokládejme, že množina $\mathbf{X}$ je konvexní a ohraničená, také optimalizovaná funkce $f(\mathbf{x})$ je konvexní funkce a řešení problému (9.55) leží na hranici množiny $\mathbf{X}$. Potom vybereme nějaký vektor $\mathbf{p} \neq 0$ a řešíme úlohu

$$\max_{\mathbf{x}\in\mathbf{X}} \left(\mathbf{p}^T\mathbf{x}\right) = \mathbf{p}^T\mathbf{x}(\mathbf{p}) \tag{9.68}$$

Předpokládejme, že řešení $\mathbf{x}(\mathbf{p})$ předchozí úlohy je jediné pro každé $\mathbf{p} \neq 0$, pak $\mathbf{x}(\mathbf{p})$ je hraniční bod množiny $\mathbf{X}$ a nadrovina $\mathbf{p}^T\mathbf{x} = \mathbf{p}^T\mathbf{x}(\mathbf{p})$ je opěrná k množině $\mathbf{X}$. Původní problém (9.55) můžeme nahradit problémem nalezení takového vektoru $\mathbf{p}$, pro který funkce $\psi(\mathbf{p}) = f(\mathbf{x}(\mathbf{p}))$ je minimální, kde $\mathbf{x}(\mathbf{p})$ je řešením (9.68), pak

$$\min_{\mathbf{x}\in\mathbf{X}} f(\mathbf{x}) = \min_{\mathbf{p}} \psi(\mathbf{p}) \tag{9.69}$$

kde vektor $\mathbf{p}$ není omezen. Metody tohoto typu jsou tzv. **nepřímé metody**. Jejich použití je možné pouze v případě jediného extrému a regulárnosti problému.

9.5.2 Diskrétní princip maxima

V tomto odstavci odvodíme nutné podmínky diskrétního principu maxima. Z odvození bude zřejmé, že bez omezujících předpokladů má pouze lokální charakter. Optimální řízení $\mathbf{u}^*(t)$ zaručí stacionární hodnotu Hamiltonovy funkce. To znamená, že libovolný stacionární bod Hamiltonovy funkce (lokální minimum či maximum nebo dokonce pouze stacionární bod) je "podezřelý" v tom smyslu, že jeden z nich může určovat optimální řízení. Pouze ve zvláštních případech platí obecná formulace diskrétního principu maxima.

Mějme tedy problém optimálního řízení diskrétního systému popsaného stavovou rovnicí

$$\mathbf{x}(k+1) = f\left(\mathbf{x}(k), \mathbf{u}(k)\right), \qquad \mathbf{x}(k_0) = \mathbf{x}_0 \tag{9.70}$$

Hledáme takovou posloupnost řízení $\mathbf{u}^*(k) \in \mathbf{U}_k$, aby bylo minimální kritérium kvality řízení

$$J\left(\mathbf{u}(k)\right) = h(\mathbf{x}(k_1)) + \sum_{k=k_0}^{k_1-1} g\left(\mathbf{x}(k), \mathbf{u}(k)\right) \tag{9.71}$$

V dalším budeme předpokládat, že funkce $f(.)$ v (9.70) a funkce $g(.)$ i $h(.)$ v (9.71) jsou spojité a mají spojité první parciální derivace podle svých proměnných a množina $\mathbf{U}_k$ je ohraničená a uzavřená.

Kritérium (9.71) upravíme do jiného tvaru rozšířením stavového prostoru o souřadnici $\mathbf{x}_0(k)$, pro kterou platí

$$x_0(k+1) = x_0(k) + g\left(\mathbf{x}(k), \mathbf{u}(k)\right) = f_0(\mathbf{x}, \mathbf{u}), \qquad x_0(k_0) = 0 \tag{9.72}$$

Potom kritérium (9.71) je rovno

$$J\left(\mathbf{u}(k)\right) = h(\mathbf{x}(k_1)) + x_0(k_1). \tag{9.73}$$

Rovnici (9.72) připojíme ke stavové rovnici systému (9.70) a optimalizační problém budeme řešit v $(n+1)$-rozměrném prostoru stavů $\mathbf{x}(k) = [x_0(k), x_1(k), \ldots, x_n(k)]^T$. Rozšířením stavového prostoru jsme problém minimalizace kritéria (9.71) převedli na problém minimalizace koncového stavu (což je Mayerova úloha).

Problém optimálního diskrétního řízení je nyní vlastně problém statické optimalizace s omezením (9.70) a (9.72). Je to tedy úloha na vázaný extrém, kterou budeme řešit Lagrangeovou metodou. Zavedeme si Lagrangeovu funkci

$$L(\mathbf{x}, \mathbf{u}, \mathbf{p}) = J(\mathbf{u}) + \sum_{i=0}^{n} p_i(k+1)\left[x_i(k+1) - f_i(\mathbf{x}, \mathbf{u})\right] \tag{9.74}$$

kde $p_0(k)$ až $p_n(k)$ jsou Lagrangeovy koeficienty, závislé na diskrétním čase k. Hledáme extrém Lagrangeovy funkce. Z podmínky $\mathrm{grad}_x L = 0$ plynou následující vztahy

$$\begin{aligned}
\frac{\partial L}{\partial x_0(k_1)} &= 1 + p_0(k_1) = 0 && (9.75)\\
\frac{\partial L}{\partial x_i(k_1)} &= \frac{\partial h(\mathbf{x}(k_1))}{\partial x_i(k_1)} + p_i(k_1) = 0, \qquad i = 1, \ldots, n\\
\frac{\partial L}{\partial x_i(k)} &= p_i(k) - \sum_{j=0}^{n} \frac{\partial f_j(\mathbf{x}, \mathbf{u})}{\partial x_i(k)} p_j(k+1) = 0, \quad \begin{array}{rcl} i &=& 1, \ldots, n;\\ k &=& k_0, \ldots, (k_1 - 1)\end{array}
\end{aligned}$$

Z (9.75a) a (9.75b) plynou okrajové podmínky

$$\begin{aligned}
p_0(k_1) &= -1 && (9.76)\\
p_i(k_1) &= -\frac{\partial h}{\partial x_i}, \qquad i = 1, \ldots, n
\end{aligned}$$

Z (9.75c) dostaneme diferenční rovnici, jejíž vektorový zápis je

$$\mathbf{p}(k) = \left(\frac{\partial f(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}}\right)^T \mathbf{p}(k+1) \tag{9.77}$$

Diferenční rovnice (9.77) je rovnice konjugovaného systému, kterou řešíme v obráceném čase vycházejíce z koncové podmínky (9.76). Koncové podmínky (9.76) konjugovaného systému (9.77) můžeme zapsat ve tvaru

$$p_i(k_1) = -\frac{\partial J}{\partial x_i}, \qquad i = 0, 1, \ldots, n. \tag{9.78}$$

Definujme si nyní Hamiltonovu funkci

$$H\left(\mathbf{p}(k+1), \mathbf{x}(k), \mathbf{u}(k)\right) = \mathbf{p}^T(k+1)\; f\left(\mathbf{x}(k), \mathbf{u}(k)\right), \qquad k = k_0, \ldots, (k_1 - 1) \tag{9.79}$$

Pomocí Hamiltonovy funkce H podle (9.79) můžeme stavové rovnice systému (9.70) spolu s (9.72) a konjugovaného systému (9.77) zapsat ve tvaru

$$\begin{aligned}
\mathbf{x}(k+1) &= \left(\frac{\partial H}{\partial \mathbf{p}(k+1)}\right)^T, \qquad k = k_0, \ldots, (k_1 - 1) && (9.80)\\
\mathbf{p}(k) &= \left(\frac{\partial H}{\partial \mathbf{x}(k)}\right)^T, \qquad k = k_0, \ldots, (k_1 - 1)
\end{aligned}$$

Známe počáteční podmínky $\mathbf{x}(t_0)$ systému (9.80a) a koncové podmínky (9.76) konjugovaného systému (9.80b). Při pevném $\mathbf{p}(k)$ a $\mathbf{x}(k)$ je Hamiltonián H funkcí řízení $\mathbf{u}(k)$. Pro každé $\mathbf{u} \in \mathbf{U}$ sestrojíme kužel přípustných variací $K(\mathbf{u})$

$$K(\mathbf{u}) = \{\delta\mathbf{u} \ : \ \mathbf{u} + \varepsilon\delta\mathbf{u} \in \mathbf{U}, \ 0 \leq \varepsilon < \varepsilon_1\} \tag{9.81}$$

Předpokládáme splnění podmínek regularity - to znamená, že kužel $K(\mathbf{u})$ je konvexní a má vnitřní bod. Výrazem $\overline{K}(\mathbf{u})$ budeme dále označovat kužel $K(\mathbf{u})$, ke kterému připojíme jeho hranici; pak $\overline{K}(\mathbf{u})$ je uzavřená množina (uzavřený kužel). Abychom odvodili nutné podmínky optimality, budeme předpokládat, že známe optimální řízení $\mathbf{u}^*(t)$ a optimální trajektorii $\mathbf{x}^*(t)$ i $\mathbf{p}^*(t)$.

Ze stavových rovnic systému plyne vztah pro variace $\delta\mathbf{x}$ a $\delta\mathbf{u}$ na optimální trajektorii $\mathbf{x}^*$ a $\mathbf{u}^*$

$$\delta\mathbf{x}^*(k+1) = \frac{\partial\mathbf{f}}{\partial\mathbf{x}}\delta\mathbf{x}^*(k) + \frac{\partial\mathbf{f}}{\partial\mathbf{u}}\delta\mathbf{u}^*(k) \tag{9.82}$$

kde $\delta\mathbf{u}^*(k) \in \overline{K}(\mathbf{u})$. Vytvoříme skalární součin vektoru $\mathbf{p}^*(k+1)$ a vektoru $\delta\mathbf{x}^*(k+1)$

$$(\mathbf{p}^*(k+1))^T \ \delta\mathbf{x}^*(k+1) = (\mathbf{p}^*(k+1))^T \left(\frac{\partial\mathbf{f}}{\partial\mathbf{x}}\delta\mathbf{x}^*(k) + \frac{\partial\mathbf{f}}{\partial\mathbf{u}}\delta\mathbf{u}^*(k)\right) \tag{9.83}$$

Podle (9.77) upravíme předchozí vztah

$$(\mathbf{p}^*(k+1))^T \ \delta\mathbf{x}^*(k+1) = (\mathbf{p}^*(k))^T \ \delta\mathbf{x}^*(k) + (\mathbf{p}^*(k+1))^T \ \frac{\partial\mathbf{f}}{\partial\mathbf{u}}\delta\mathbf{u}^*(k) \tag{9.84}$$

Provedeme-li součet předchozího výrazu pro $k = k_0, \ldots (k_1 - 1)$, dostaneme

$$(\mathbf{p}^*(k_1))^T \ \delta\mathbf{x}^*(k_1) - (\mathbf{p}^*(k_0))^T \ \delta\mathbf{x}^*(k_0) = \sum_{k=k_0}^{k_1-1} (\mathbf{p}^*(k+1))^T \ \frac{\partial\mathbf{f}}{\partial\mathbf{u}}\delta\mathbf{u}^*(k) \tag{9.85}$$

Protože počáteční podmínky jsou dané, je variace $\delta\mathbf{x}(k_0) = 0$. Dosazením za $\mathbf{p}(k_1)$ z (9.78) upravíme levou stranu předchozí rovnice do tvaru

$$-\left(\frac{\partial J}{\partial\mathbf{x}}\right)^T \delta\mathbf{x}^*(k_1) = -\delta J(\mathbf{u}^*) \tag{9.86}$$

kde δJ je variace kritéria kvality řízení. Označme $\delta_{\mathbf{u}}H$ přípustný diferenciál Hamiltonovy funkce

$$\delta_{\mathbf{u}}H(\mathbf{x},\mathbf{u},\mathbf{p}) = \frac{\partial H}{\partial\mathbf{u}}\delta\mathbf{u} = \mathbf{p}^T(k+1)\frac{\partial\mathbf{f}}{\partial\mathbf{u}}\delta\mathbf{u} \tag{9.87}$$

kde $\delta\mathbf{u} \in \bar{K}(\mathbf{u}^*)$. Potom vztah (9.85) můžeme upravit do tvaru

$$-\delta J(\mathbf{u}^*) = \sum_{k=k_0}^{k_1-1} \delta_{\mathbf{u}}H\left(\mathbf{x}^*(k), \mathbf{u}^*(k), \mathbf{p}^*(k+1)\right) \tag{9.88}$$

Protože $J(\mathbf{u}^*)$ je minimální hodnota kritéria, platí $\delta J(\mathbf{u}^*) \geq 0$ pro libovolnou variaci $\delta\mathbf{u} \in \overline{K}(\mathbf{u}^*)$. Zvolíme-li variaci $\delta\mathbf{u}(i)$ nenulovou pouze v čase $k = i$, zbude ze sumy na pravé straně předchozí rovnice pouze jediný nenulový člen. Proto na optimální trajektorii platí

$$\delta_{\mathbf{u}}H\left(\mathbf{x}^*(k), \mathbf{u}^*(k), \mathbf{p}^*(k+1)\right) \leq 0, \qquad \delta\mathbf{u} \in \bar{K}(\mathbf{u}^*) \tag{9.89}$$

neboli

$$\frac{\partial H\left(\mathbf{x}^*(k),\mathbf{u}^*(k),\mathbf{p}^*(k+1)\right)}{\partial \mathbf{u}}\delta\mathbf{u}\leq 0,\qquad \delta\mathbf{u}\in\overline{K}(\mathbf{u}^*) \tag{9.90}$$

Vztah (9.89) resp. (9.90) je obecně platný závěr o vlastnostech Hamiltonovy funkce na optimální trajektorii při diskrétním řízení.

Je-li $\mathbf{u}^*$ vnitřní bod množiny přípustných řízení $\mathbf{U}$, pak variace $\delta\mathbf{u}$ je libovolná a z (9.90) plyne

$$\frac{\partial H\left(\mathbf{x}^*(k),\mathbf{u}^*(k),\mathbf{p}^*(k+1)\right)}{\partial \mathbf{u}}=0. \tag{9.91}$$

V tomto případě je Hamiltonián na optimální trajektorii stacionární.

Pro $\delta_u H\left(\mathbf{x}^*(k),\mathbf{u}^*(k),\mathbf{p}^*(k+1)\right)<0$, leží řízení $\mathbf{u}^*\in\mathbf{U}$ na hranici oblasti. Hamiltonova funkce dosahuje v tomto případě na optimální trajektorii lokální maximum. O globálním chování Hamiltonovy funkce nemůžeme bez dodatečných předpokladů nic tvrdit.

Je-li omezení řízení určeno nerovnostmi $g_i(\mathbf{u}(k))\leq 0$, pak podle (9.58) je kužel přípustných variací $\delta\mathbf{u}^*$ určen vztahy

$$\frac{\partial g_i\left(\mathbf{u}^*(k)\right)}{\partial \mathbf{u}}\delta\mathbf{u}\leq 0,\qquad i\in\mathcal{J}(\mathbf{u}^*). \tag{9.92}$$

Nyní si zavedeme množiny dosažitelnosti $R_k(\mathbf{x}_0)$. Jsou to množiny stavů $\mathbf{x}$, které jsme schopni dosáhnout z počátečního stavu $\mathbf{x}(k_0)=\mathbf{x}_0$ v čase $k\geq k_0$ přípustným řízením.

Je zřejmé, že zavedením množin dosažitelnosti jsme problém dynamické optimalizace převedli na úlohu statické optimalizace kritéria (9.71) na množině $R_{k_1}(\mathbf{x}_0)$.

V dalším budeme potřebovat, aby množina dosažitelnosti $R_{k_1}(\mathbf{x}_0)$ byla konvexní. Zavedeme si množiny $R(\mathbf{x})$, kde

$$R(\mathbf{x})=\{\mathbf{y}\ :\ \mathbf{y}=\mathbf{f}(\mathbf{x},\mathbf{u}],\ \mathbf{u}\in\mathbf{U}\} \tag{9.93}$$

což jsou množiny stavů dosažitelných z počátečního stavu $\mathbf{x}$ za jeden krok přípustným řízením. Je zřejmé, že množina $R_{k_1}(\mathbf{x}_0)$ je konvexní, je-li konvexní množina $R(\mathbf{x})$ pro všechny $\mathbf{x}\in\mathbf{X}$.

Je-li množina $R(\mathbf{x})$ konvexní, pak postupem provedeným v předchozím odstavci můžeme ukázat, že na optimální trajektorii platí princip maxima, to znamená, že na optimální trajektorii je Hamiltonián maximální.

Uvědomme si, že pro problém optimálního řízení spojitých systémů jsme neměli žádný požadavek na konvexnost množin dosažitelnosti. Množiny dosažitelnosti za malý časový interval Δt jsou vždy konvexní - odtud plyne tzv. konvexifikující účinek spojitého času.

Na závěr tohoto odstavce uvedeme formulaci diskrétního principu maxima:

Věta: Diskrétní princip maxima
Mějme diskrétní systém popsaný stavovou rovnicí (9.70) a hledejme optimální řízení $\mathbf{u}^*(k)\in\mathbf{U}$*, které minimalizuje kritérium kvality řízení (9.71). Podle (9.72) upravíme kritérium do tvaru (9.73) a problém řešíme v* $(n+1)$*-rozměrném prostoru stavů.*

Je-li $\mathbf{u}^*(k)$ *přípustné optimální řízení a* $\mathbf{x}^*(k)$ *optimální trajektorie a množina dosažitelnosti* $R(\mathbf{x})$ *stavů za jeden krok je konvexní, pak existuje nenulový* $(n+1)$*-rozměrný vektor* $\mathbf{p}(k)$*, popsaný diferenční rovnicí (9.80b) s koncovou podmínkou (9.76), kde Hamiltonova*

funkce H podle (9.79) je pro všechny k, $k_0 \leq k < k_1$ maximální vzhledem k řízení $\mathbf{u}(k)$*, neboli*

$$H\left(\mathbf{x}^*(k), \mathbf{u}^*(k), \mathbf{p}^*(k+1)\right) \geq H\left(\mathbf{x}^*(k), \mathbf{u}(k), \mathbf{p}^*(k+1)\right), \qquad \mathbf{u}(k) \in \mathbf{U} \tag{9.94}$$

Poznámka: Je-li systém lineární vůči řízení a množina přípustných řízení je konvexní množina a funkce g v kritériu (9.71) je také konvexní funkce, pak je i Hamiltonián konvexní a platí diskrétní princip maxima.

Poznámka: Obrácením znaménka koncové podmínky (9.76) konjugovaného systému se maximum Hamiltoniánu změní na jeho minimum - dostaneme potom často používaný diskrétní princip minima.

9.6 Příklady

1. Mějme jednoduchý systém druhého řádu popsaný stavovou rovnicí

   $$\dot{x}_1(t) = x_2(t), \quad \dot{x}_2(t) = u(t), \qquad |u(t)| \leq 1$$

   Nalezněte časově optimální řízení, které v minimálním čase převede libovolný počáteční stav $\mathbf{x}$ do stavu $\mathbf{x}(T) = [x_1(T), 0]^T$. Uvědomme si, že platí věta o n intervalech. Jaké budou přepínací křivky ve stavové rovině?

   Přepínací křivky lze získat řešením stavové rovnice v obráceném čase z koncové podmínky $\mathbf{x}(T) = [x_1(T), 0]^T$ a volbou řízení $u^*(t) = +1$ nebo $u^*(t) = -1$.

2. Uvažujte kmitavý systém druhého řádu popsaný stavovou rovnicí

   $$\dot{x}_1(t) = x_2(t), \quad \dot{x}_2(t) = -x_1(t) + u(t), \qquad |u(t)| \leq 1$$

   Nalezněte časově optimální řízení, které v minimálním čase převede libovolný počáteční stav $\mathbf{x}$ do stavu $\mathbf{x}(T) = [0\ ,\ 0]^T$.

   Uvědomme si, že zde sice neplatí věta o n intervalech, ale platí věta o konečném počtu přepnutí. Jaké budou přepínací křivky ve stavové rovině? Konjugovaný systém bude také kmitavý se stejnou periodou kmitů, proto přepínání řízení z jedné krajní hodnoty na druhou nastává pravidelně po půl periodě kmitů konjugovaného systému. Trajektorie systému při konstantním řízení jsou kružnice se středy určenými velikostí řízení. Zobecněte toto řešení na kmitavý systém druhého řádu s tlumením.

3. Určete energeticky optimální řízení systému

   $$\dot{x}_1(t) = x_2(t), \quad \dot{x}_2(t) = u(t), \qquad |u(t)| \leq 1,$$

   které daný počáteční stav $\mathbf{x}(0)$ převede do koncového stavu $\mathbf{x}(T)$ za pevnou dobu řízení T. Optimální řízení minimalizuje spotřebu řídicí energie, která je úměrná kritériu

   $$J(u(t)) = \int_0^T |u(t)|\ dt$$

   Proveďte nejprve fyzikální rozbor problému. Rozhodněte, zda existují omezení na dobu řízení a ukažte, že optimální řízení nabývá pouze hodnoty $(+1, -1, 0)$. Určete přepínací křivky.

4. Energeticky optimální řízení vlaku:
Stavové rovnice popisující dynamické vlastnosti vlaku jsou

$$\dot{x}_1(t) = x_2(t), \qquad \dot{x}_2(t) = u(t) - q(x_2, x_1)$$

kde x_1 a x_2 je poloha a rychlost vlaku, $u(t)$ je tažná síla elektrické pohonné jednotky, která je omezena $u_b \leq u(t) \leq u_t$, kde u_b je maximální brzdná síla a u_t je maximální tažná síla, $q(x_2, x_1)$ jsou brzdné odpory závislé na rychlosti a poloze vlaku. Platí $q(x_2, x_1) = a + bx_2 + c(x_2)^2 + s(x_1)$, kde $s(x_1)$ je síla vyvolaná sklonem trati.

Naším úkolem je přemístit vlak z jedné stanice do druhé za předepsaný čas tak, aby energetické kritérium ve tvaru

$$J = \int_0^T \frac{1}{2} \left(u(t) + |u(t)|\right) x_2(t)\ dt$$

bylo minimální. Počáteční a koncové body jsou pevné $\mathbf{x}_0 = [0\ ,\ 0]^T$, $\mathbf{x}_1 = [d\ ,\ 0]^T$, kde d je vzdálenost mezi stanicemi.

Uvažujte nejprve $a = b = c = s = 0$. Určete minimální dobu jízdy T. Ukažte, že v této úloze mohou existovat singulární řešení. Ověřte, že optimální trajektorie se skládá z rozjezdu maximální tažnou silou, jízdou konstantní rychlostí, dojezdu (tažná síla je nulová) a brzdění maximální silou. Ostatně takové jízdní režimy bychom pravděpodobně volili i bez použití teorie.

5. Optimální řízení meteorologické rakety:
Meteorologická raketa se pohybuje kolmo k povrchu zemskému. Raketa má omezené množství paliva a chceme řídit tah motorů rakety tak, aby raketa vyletěla do maximální výšky.

Nechť h je vertikální poloha rakety a v je její rychlost. Počáteční podmínky jsou $h(0) = v(0) = 0$ a koncový bod je $h(T) = h_{max}$, $v(T) = 0$, kde T je pevná doba pohybu rakety vzhůru. Hmotnost rakety $m(t)$ se v čase mění podle toho, jak ubývá paliva. Platí tedy

$$m(t) = M_r + M_p - \int_0^t \mu(\tau)\ d\tau$$

kde M_r je hmotnost rakety bez paliva, M_p je hmotnost paliva při startu a $\mu(t)$ je spotřeba paliva za $[s]$ v čase t.

Tah F_r motoru rakety se v čase mění a je roven $F_r = f(\mu(t)) \doteq C\mu(t)$. Uvažujme přibližně lineární závislost mezi tahem rakety a spotřebou paliva. Diferenciální rovnice pohybu rakety je rovna

$$d(m(t)v(t)) = \sum F(t)\ dt = (F_r - F_p(v, h) - mg(h))\ dt$$

kde $g(h)$ je gravitační zrychlení závislé na výšce rakety a F_p je síla odporu prostředí přibližně úměrná rychlosti $F_p \doteq Bv$. Z předchozího plyne diferenciální rovnice

$$\frac{dm(t)}{dt} = -\mu(t), \qquad m(0) = M_r + M_p, \quad m(T) = M_r$$

Pro dané konstanty M_r, M_p, C a B určete optimální spotřebu paliva $\mu(t)$ tak, aby výška $h(T)$ byla maximální.

6. Modifikujte předchozí úlohu tak, že je dána souřadnice $h(T)$ a hledejte optimální průběh $\mu(t)$ tak, aby celková spotřeba paliva byla minimální.

7. Časově optimální řízení lineárního systému.
   Naším úkolem je přemístit počáteční stav $\mathbf{x}(0)$ lineárního systému
   $\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$ do počátku $\mathbf{x}(T) = 0$ v minimálním čase T omezeným řízením $\mathbf{U}_{min} \leq \mathbf{u}(t) \leq \mathbf{U}_{max}$.

   Sestavte program pro výpočet počáteční podmínky $\mathbf{p}(0)$ konjugovaného systému řešící náš problém podle principu maxima.

   Rovnice konjugovaného systému je $\dot{\mathbf{p}}(t) = -\mathbf{A}^T\mathbf{p}(t)$. Iterační algoritmus podle Neustadta je následující:

   - Algoritmus startujeme volbou počáteční podmínky $\mathbf{p}_0(0)$ konjugovaného systému tak, aby $\mathbf{p}_0^T(0)\mathbf{x}(0) < 0$, na příklad
     $\mathbf{p}_0(0) = -\dfrac{\mathbf{x}(0)}{\|\mathbf{x}(0)\|}$.
   - V i-té iteraci pro $\mathbf{p}_i(0)$ řešíme stavovou rovnici systému i konjugovaného systému z daných počátečních podmínek a optimálním řízením podle (9.53). To řešíme až do času $t = t_i$, kdy $(\mathbf{p}_i(t_i))^T \mathbf{x}(t_i) = 0$
   - Nyní řešíme diferenciální rovnici $\dot{\mathbf{v}}(t) = -\mathbf{A}\mathbf{v}(t)$ s počáteční podmínkou $\mathbf{v}(0) = \mathbf{x}(t_i)$ a získáme řešení $\mathbf{v}(t_i) = e^{-\mathbf{A}t_i}\mathbf{x}(t_i) = \boldsymbol{\gamma}_i$.
   - Novou iteraci počáteční podmínky konjugovaného systému získáme podle vztahu
     $$\mathbf{p}_{i+1}(0) = \mathbf{p}_i(0) - \Delta_i\boldsymbol{\gamma}_i$$
     Krok Δ_i volíme takový, aby v další iteraci čas t_{i+1} byl větší než čas t_i v předchozí iteraci. Pokud čas roste, pak zvětšíme krok $\Delta_i := 2\Delta_i$, pokud čas neroste, krok zmenšíme $\Delta_i := 0.5\Delta_i$.
   - Algoritmus končí, pokud je $\boldsymbol{\gamma}_i$ malé, nebo $\mathbf{x}(t_i)$ je se zvolenou přesností blízko počátku (koncovému stavu).

   Zdůvodnění algoritmu je jednoduché.
   Určíme funkci $f(t, \mathbf{p}_i(0)) = \mathbf{p}_i^T(0)\,(\mathbf{x}(0) - \mathbf{z}(t, \mathbf{p}_i(0)))$, kde $\mathbf{z}(t, \mathbf{p}_i(0))$ je počáteční stav systému, který optimálním řízením (s počáteční podmínkou $\mathbf{p}_i(0)$ konjugovaného systému) převedeme do počátku za čas t.

   Z řešení stavových rovnic lineárního systému plyne, že $\mathbf{z}(t, \mathbf{p}_i(0)) =$
   $-\int_0^t e^{\mathbf{A}\tau}\mathbf{B}\mathbf{u}^*(t)\,d\tau$. Po dosazení je funkce

   $$\begin{aligned} f(t, \mathbf{p}_i(0)) &= \mathbf{p}_i^T(0)\,(\mathbf{x}(0) - \mathbf{z}(t, \mathbf{p}_i(0))) = \mathbf{p}_i^T(0)e^{-\mathbf{A}t}\mathbf{x}(t) \\ &= \left(e^{-\mathbf{A}^T t}\mathbf{p}_i(0)\right)^T \mathbf{x}(t) = \mathbf{p}^T(t)\mathbf{x}(t) \end{aligned}$$

   Vektor $(\mathbf{x}(0) - \mathbf{z}(t_i, \mathbf{p}_i(0))) = \boldsymbol{\gamma}_i$ je pro čas t_i tečný vektor k množině stavů $S_i(t_i)$, ze kterých se za čas $t \leq t_i$ dostanu do cíle (počátku souřadnic) přípustným řízením. Vektor $\mathbf{p}_i(0)$ je normálový vektor k této množině. Aby množina $S_{i+1}(t_{i+1})$ v další iteraci vzrostla ($S_i(t_i) \subset S_{i+1}(t_{i+1})$), je třeba opravit počáteční podmínku konjugovaného systému o vektor $\boldsymbol{\gamma}_i = e^{-\mathbf{A}t_i}\mathbf{x}(t_i)$.

8. Mějme diskrétní systém popsaný stavovými rovnicemi

$$x_1(k+1) = x_1(k) + 2u(k) \qquad x_2(k+1) = -x_1^2(k) + x_2(k) + u^2(k)$$

Určete optimální řízení minimalizující kritérium $J(u) = -x_2(2)$. Počáteční podmínka je $\mathbf{x}(0) = [3\ ,\ 0]^T$ a řízení je omezeno $|u(k)| \leq 5$.

Ukažte, že množiny dosažitelnosti nejsou konvexní. Optimální řízení můžeme vypočítat dosazením stavových rovnic do kritéria a dostaneme $u^*(0) = -2$, $u^*(1) = \pm 5$ - ověřte. Hamiltonián je pro $u^*(0)$ minimální a pro $u^*(1)$ je naopak maximální.

Kapitola 10

Stochasticky optimální řízení

Tato kapitola je věnována případové studii použití dynamického programování na syntézu optimálního řízení stochastických systémů. Tento přístup se také označuje jako **stochastické dynamické programování**. V této kapitole odvodíme algoritmus optimálního řízení stochastických lineárních diskrétních systémů. Kritérium optimality bude kvadratické ve stavech a řízení. Odvodíme tzv. důvěřivé i opatrné strategie pro dva stochastické modely - model ARMAX (AutoRegresive Moving Average with eXternal input - autoregresní model s pohyblivým průměrem a vnějším vstupem) a ARX (AutoRegresive with eXternal input).

10.1 Stochasticky optimální řízení ARMAX modelu

V literatuře je popsáno současné odhadování stavů a parametrů ARMAX modelu. Využijeme tyto výsledky a v této kapitole odvodíme kvadraticky optimální řízení lineárního ARMAX modelu a to tzv. důvěřivé i opatrné strategie LQG řízení (Linear system, Quadratic criterion, Gausian noise - řízení lineárního systému a kvadratickým kritériem a gausovským šumem). Nejprve uvedeme stavové rovnice ARMAX modelu v pozorovatelném kanonickém tvaru, který je vhodný pro syntézu řízení. Provedeme úpravu tohoto modelu do tvaru, který umožňuje provést současné odhadování stavů a parametrů.

10.1.1 ARMAX model a jeho pozorovatelný kanonický tvar

Nechť dynamické vlastnosti procesu jsou popsány časově proměnným ARMAX modelem n-tého řádu se známými parametry šumu c_i a známým rozptylem šumu σ_e^2

$$y(t) + \sum_{i=1}^{n} a_i(t-i)y(t-i) = \sum_{i=0}^{n} b_i(t-i)u(t-i) + \sum_{i=1}^{n} c_i(t-i)e(t-i) + e(t) \tag{10.1}$$

kde $y(t)$ je výstup modelu, $u(t)$ je řízení, $e(t) \sim \mathcal{N}(0,\, \sigma_e^2)$ je Gaussův bílý šum, nezávislý na hodnotách výstupů $y(t-i)$, $i \geq 1$, a vstupů $u(t-i)$, $i \geq 0$, a

$$\boldsymbol{\theta}(t) = [b_0(t), \ldots, b_n(t), a_1(t), \ldots, a_n(t)]^T \tag{10.2}$$

je vektor neznámých parametrů. Dále budeme používat vektor neznámých parametrů $a_i(t)$, vektor neznámých parametrů $b_i(t)$ a vektor známých parametrů šumu $c_i(t)$

$$\mathbf{a}(t) = \begin{bmatrix} a_1(t) \\ \vdots \\ a_n(t) \end{bmatrix}, \quad \mathbf{b}(t) = \begin{bmatrix} b_1(t) \\ \vdots \\ b_n(t) \end{bmatrix}, \quad \mathbf{c}(t) = \begin{bmatrix} c_1(t) \\ \vdots \\ c_n(t) \end{bmatrix}.$$

Pozorovatelný kanonický tvar časově proměnného ARMAX modelu je popsaný stavovou rovnicí

$$\begin{aligned} \mathbf{x}(t+1) &= \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)u(t) + \mathbf{C}(t)e(t) \\ y(t) &= \mathbf{h}_x\mathbf{x}(t) + d(t)u(t) + e(t) \end{aligned} \tag{10.3}$$

kde

$$\mathbf{A}(t) = \begin{bmatrix} -a_1(t) & 1 & 0 & & 0 \\ -a_2(t) & 0 & 1 & & 0 \\ & \vdots & & \ddots & \\ -a_{n-1}(t) & 0 & 0 & & 1 \\ -a_n(t) & 0 & 0 & & 0 \end{bmatrix}$$

$$\mathbf{B}(t) = \mathbf{b}(t) - b_0(t)\mathbf{a}(t) = \begin{bmatrix} b_1(t) - b_0a_1 \\ \vdots \\ b_n(t) - b_0a_n \end{bmatrix}, \quad \mathbf{C}(t) = \mathbf{c}(t) - \mathbf{a}(t)$$

$$\mathbf{h}_x = \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}, \qquad d = b_0$$

Tento tvar stavových rovnic ARMAX modelu budeme používat v dalším odstavci při odvození stochasticky optimálního řízení. Model v tomto tvaru není ale vhodný pro současné odhadování stavů a parametrů, protože se v něm vyskytují součiny těchto proměnných. Proto stavový model upravíme do tvaru - viz blokové schéma na obr. 10.1.

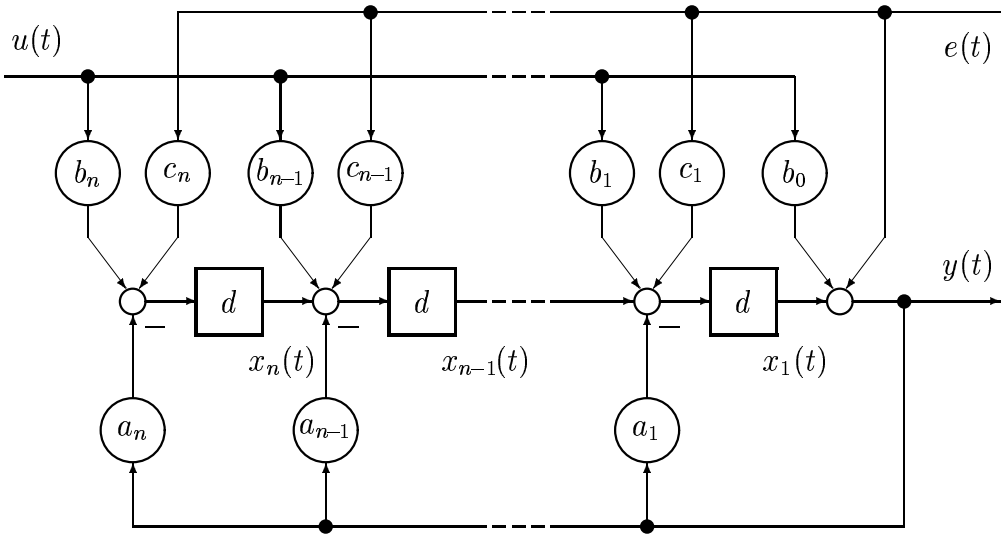
$$\begin{aligned} \mathbf{x}(t+1) &= \overline{\mathbf{F}}\mathbf{x}(t) - \mathbf{a}(t)y(t) + \mathbf{b}(t)u(t) + \mathbf{c}(t)e(t) \\ y(t) &= x_1(t) + b_0(t)u(t) + e(t) \end{aligned} \tag{10.4}$$

kde

$$\overline{\mathbf{F}} = \begin{bmatrix} 0 & 1 & 0 & & 0 \\ 0 & 0 & 1 & & 0 \\ & \vdots & & \ddots & \\ 0 & 0 & 0 & & 1 \\ 0 & 0 & 0 & & 0 \end{bmatrix}$$

Ve stavové rovnici (10.4) je také výstupní proměnná $y(t)$, proto v matici systému $\overline{\mathbf{F}}$ nejsou neznámé parametry. Šum $e(t)$ se vyskytuje ve stavové rovnici i v rovnici výstupní. Můžeme proto šum $e(t)$ vyjádřit z výstupní rovnice a výsledek dosadit do rovnice stavové. Stavový model se touto úpravou transformuje do tvaru

$$\begin{aligned} \mathbf{x}(t+1) &= \left[\overline{\mathbf{F}} - \mathbf{c}(t)\mathbf{h}_x\right]\mathbf{x}(t) + \left[\mathbf{b}(t) - b_0(t)\mathbf{c}(t)\right]u(t) - \left[\mathbf{a}(t) - \mathbf{c}(t)\right]y(t) \\ y(t) &= \mathbf{h}_x\mathbf{x}(t) + \mathbf{h}_\theta\boldsymbol{\theta}(t)u(t) + e(t) \end{aligned} \tag{10.5}$$



Obrázek 10.1: Pozorovatelný kanonický tvar realizace ARMAX modelu

kde $\mathbf{h}_x$ je řádkový vektor rozměru n, $\mathbf{h}_x = [\ 1, 0, \ldots, 0\ ]$ a $\mathbf{h}_\theta$ je řádkový vektor rozměru $(2n+1)$, $\mathbf{h}_\theta = [\ 1, 0, \ldots, 0\ ]$. Tím jsme vlastně provedli dekorelaci šumu stavu a šumu výstupu. Stavová rovnice po této úpravě není vůbec zatížena šumem.

Pro účely odhadu parametrů je třeba mít model vývoje parametrů. Pokud nemáme žádnou apriorní informaci o vývoji parametrů, modelujeme jejich vývoj ve tvaru náhodné procházky

$$\boldsymbol{\theta}(t+1) = \boldsymbol{\theta}(t) + \boldsymbol{\nu}(t) \tag{10.6}$$

kde $\boldsymbol{\nu}(t) \sim \mathcal{N}(0, \sigma_e^2 \mathbf{V}(t))$ je Gaussův bílý šum s nulovou střední hodnotou a známou normalizovanou kovarianční maticí $\mathbf{V}(t)$. Kovarianční matici $\mathbf{V}(t)$ uvažujeme obvykle pouze s nenulovými diagonálními prvky. Jejich velikost je dána naší apriorní představou o rychlosti změny parametrů.

Pro současné odhadování stavů a parametrů systému zavedeme rozšířený stavový vektor složený z vektoru parametrů a stavů

$$\mathbf{z}(t) = \left[\ \boldsymbol{\theta}^T(t) \quad \mathbf{x}^T(t)\ \right]^T.$$

Potom stavové rovnice rozšířeného systému vzniknou složením stavových rovnic ARMAX modelu a stavových rovnic vývoje parametrů

$$\begin{aligned} \mathbf{z}(t+1) &= \mathbf{F}(t)\mathbf{z}(t) + \mathbf{G}(t)y(t) + \overline{\boldsymbol{\nu}}(t) \\ y(t) &= \mathbf{h}(t)\mathbf{z}(t) + e(t), \end{aligned} \tag{10.7}$$

kde

$$\mathbf{F}(t) = \begin{bmatrix} \mathbf{I}_{2n+1} & 0 \\ \mathbf{H}(t) & \mathbf{J}(t) \end{bmatrix}, \quad \mathbf{G}(t) = \begin{bmatrix} 0 \\ \mathbf{c}(t) \end{bmatrix}, \quad \overline{\boldsymbol{\nu}}(t) = \begin{bmatrix} \boldsymbol{\nu}(t) \\ 0 \end{bmatrix} \tag{10.8}$$

a kde matice $\mathbf{H}$ a $\mathbf{J}$ ve stavové matici $\mathbf{F}$ a výstupní řádkový vektor $\mathbf{h}$ jsou rovny

$$\mathbf{H}(t) = [-u(t)\mathbf{c}(t),\ u(t)\mathbf{I}_n,\ -y(t)\mathbf{I}_n]$$

$$\begin{aligned}\mathbf{J}(t) &= \overline{\mathbf{F}} - \mathbf{c}(t)\mathbf{h}_x \\ \mathbf{h}(t) &= \begin{bmatrix} \mathbf{h}_\theta u(t) & \mathbf{h}_x \end{bmatrix}.\end{aligned}$$

Rozšířený stavový model v tomto tvaru je vhodný pro současné odhadování stavů a parametrů systému, neboť stavová matice $\mathbf{F}$ neobsahuje neznámé parametry. Zvláštností rozšířeného modelu v tomto tvaru je to, že ve stavové rovnici je výstupní vektor $y(t)$, což nevadí při odhadování rozšířeného stavu (parametrů a stavu původního systému). Matice systému $\mathbf{F}(t)$ závisí na datech (viz submatice $\mathbf{H}(t)$) a také výstupní vektor $\mathbf{h}(t)$ je závislý na datech. Systém (10.7) s rozšířeným stavem je tedy lineárním nestacionárním systémem.

10.1.2 Současné odhadování stavů a parametrů ARMAX modelu

V tomto odstavci odvodíme potřebné vztahy pro odhadování stavů i parametrů ARMAX modelu. Protože je systém ve stavech i parametrech lineární, výsledkem je Kalmanův filtr bez aproximací.

Předpokládejme, že pozorujeme vstup $u(\tau)$ a výstup $y(\tau)$ pro $\tau = 1, \ldots, t-1$ a naše znalost parametrů a stavu systému založená na množině dat

$$\mathcal{D}^{t-1} = \{u(1),\, y(1),\, \ldots,\, u(t-1),\, y(t-1)\}$$

je popsána podmíněnou hustotou

$$p\left(\mathbf{z}(t)|\mathcal{D}^{t-1}\right) = p\left(\begin{bmatrix}\boldsymbol{\theta}(t)\\ \mathbf{x}(t)\end{bmatrix}\middle|\ \mathcal{D}^{t-1}\right) \tag{10.9}$$

Naším problémem je aktualizace znalosti popsané podmíněnou hustotou pravděpodobnosti $p\left(\boldsymbol{\theta}(t), \mathbf{x}(t)|\, \mathcal{D}^{t-1}\right)$ na podmíněnou hustotu $p\left(\boldsymbol{\theta}(t+1), \mathbf{x}(t+1)|\, \mathcal{D}^{t}\right)$ poté, co získáme nová data o vstupu a výstupu $\{u(t),\, y(t)\}$. Výstupní rovnice (10.7b) definuje podmíněnou hustotu

$$p\left(y(t)|\, \mathbf{z}(t), u(t)\right) = p\left(y(t)|\, \boldsymbol{\theta}(t), \mathbf{x}(t), u(t)\right). \tag{10.10}$$

Stavová rovnice (10.7a) rozšířeného systému definuje podmíněnou hustotu

$$p\left(\mathbf{z}(t+1)|\ \mathbf{z}(t), u(t), y(t), \mathbf{V}(t)\right) = p\left(\begin{bmatrix}\boldsymbol{\theta}(t+1)\\ \mathbf{x}(t+1)\end{bmatrix}\middle|\ \boldsymbol{\theta}(t), \mathbf{x}(t), u(t), y(t), \mathbf{V}(t)\right). \tag{10.11}$$

Řešení našeho problému získáme následujícím postupem:

1. Je dána podmíněná hustota $p\left(\mathbf{z}(t)|\mathcal{D}^{t-1}\right) = p\left(\begin{bmatrix}\boldsymbol{\theta}(t)\\ \mathbf{x}(t)\end{bmatrix}\middle|\ \mathcal{D}^{t-1}\right)$.

2. Použitím výstupního modelu ve tvaru podmíněné hustoty $p\left(y(t)|\, \mathbf{z}(t), u(t)\right)$ určíme vzájemnou podmíněnou hustotu

$$p\left(y(t), \mathbf{z}(t)|\ \mathcal{D}^{t-1}, u(t)\right) = p\left(y(t)|\, \mathbf{z}(t), u(t)\right)\, p\left(\mathbf{z}(t)|\ \mathcal{D}^{t-1}, u(t)\right) \tag{10.12}$$

Přitom využijeme tzv. přirozené podmínky řízení

$$p\left(\mathbf{z}(t)|\ \mathcal{D}^{t-1}, u(t)\right) = p\left(\mathbf{z}(t)|\ \mathcal{D}^{t-1}\right)$$

3. Pomocí hodnoty výstupu $y(t)$, určíme podmíněnou hustotu

$$p\left(\mathbf{z}(t)|\ \mathcal{D}^t\right) = \frac{p\left(y(t), \mathbf{z}(t)|\ \mathcal{D}^{t-1}, u(t)\right)}{p\left(y(t)|\,\mathcal{D}^{t-1}, u(t)\right)} \tag{10.13}$$

kde

$$p\left(y(t)|\,\mathcal{D}^{t-1}, u(t)\right) = \int p\left(y(t), \mathbf{z}(t)|\,\mathcal{D}^{t-1}, u(t)\right)\, d\mathbf{z}(t). \tag{10.14}$$

Tím ukončíme datový krok koncepčního řešení.

4. Použitím rozšířeného stavového modelu vývoje stavů a parametrů (10.7)

$$p\left(\mathbf{z}(t+1)|\ \mathbf{z}(t), \mathcal{D}^t\right) = p\left(\mathbf{z}(t+1)|\ \mathbf{z}(t), u(t), y(t), \mathbf{V}(t)\right) \tag{10.15}$$

určíme prediktivní podmíněnou hustotu

$$p\left(\mathbf{z}(t+1)|\ \mathcal{D}^t\right) = \int p\left(\mathbf{z}(t+1)|\,\mathbf{z}(t), \mathcal{D}^t\right) p\left(\mathbf{z}(t)|\ \mathcal{D}^t\right) d\mathbf{z}(t) \tag{10.16}$$

čímž ukončíme časový krok koncepčního řešení.

Tím jsme provedli koncepční řešení průběžného odhadu stavů a parametrů ARMAX modelu ve specielním kanonickém tvaru. Podmíněnou hustotu stavů a parametrů (10.9) jsme získali pro inkrementální časový index $(t+1)$.

Pokud je podmíněná hustota pravděpodobnosti (10.9) normální, potom všechny předchozí podmíněné hustoty jsou také normální. Normalita je zachována (normální rozdělení se reprodukuje) a proto můžeme používat pouze první dva momenty normálního rozdělení, to znamená, že stačí v jednotlivých krocích aktualizovat pouze střední hodnotu a kovarianční matici.

Uvažujme tedy podmíněnou hustotu pravděpodobnosti rozšířeného stavu (parametrů a stavu původního modelu) ve tvaru

$$p(\mathbf{z}(t)|\,\mathcal{D}^{t-1}) = p\left(\begin{bmatrix}\boldsymbol{\theta}(t)\\ \mathbf{x}(t)\end{bmatrix}\middle|\ \mathcal{D}^{t-1}\right) = \mathcal{N}\left(\begin{bmatrix}\widehat{\boldsymbol{\theta}}(t|\,t-1)\\ \widehat{\mathbf{x}}(t|\,t-1)\end{bmatrix}; \sigma_e^2\mathbf{P}(t|\,t-1)\right) \tag{10.17}$$

Potom použitím výstupní rovnice (10.7b) získáme sdruženou podmíněnou hustotu

$$p\left(\begin{bmatrix}y(t)\\ \mathbf{z}(t)\end{bmatrix}\middle|\ \mathcal{D}^{t-1}\right) = p\left(\begin{bmatrix}y(t)\\ \boldsymbol{\theta}(t)\\ \mathbf{x}(t)\end{bmatrix}\middle|\ \mathcal{D}^{t-1}\right) = \mathcal{N}\left(\begin{bmatrix}\hat{y}(t|\,t-1)\\ \widehat{\boldsymbol{\theta}}(t|\,t-1)\\ \widehat{\mathbf{x}}(t|\,t-1)\end{bmatrix}; \sigma_e^2\mathbf{P}_y\right) \tag{10.18}$$

kde

$$\hat{y}(t|\,t-1) = \mathbf{h}(t)\widehat{\mathbf{z}}(t|\,t-1) = [\mathbf{h}_\theta u(t),\ \mathbf{h}_x]\begin{bmatrix}\widehat{\boldsymbol{\theta}}(t|\,t-1)\\ \widehat{\mathbf{x}}(t|\,t-1)\end{bmatrix}$$

a

$$\mathbf{P}_y = \begin{bmatrix}1+\mathbf{h}(t)\mathbf{P}(t|\,t-1)\mathbf{h}^T(t) & \mathbf{h}(t)\mathbf{P}(t|\,t-1)\\ \mathbf{P}(t|\,t-1)\mathbf{h}^T(t) & \mathbf{P}(t|\,t-1)\end{bmatrix}.$$

Za předpokladu normality podmíněných hustot spolu s využitím vztahů pro podmíněné hustoty provedeme datový krok Bayesovské rekurze. Podmíněná normální hustota je rovna

$$p(\mathbf{z}(t)|\mathcal{D}^t) = p\left(\begin{bmatrix}\boldsymbol{\theta}(t)\\ \mathbf{x}(t)\end{bmatrix}\middle|\ \mathcal{D}^t\right) = \mathcal{N}\left(\begin{bmatrix}\widehat{\boldsymbol{\theta}}(t|\,t)\\ \widehat{\mathbf{x}}(t|\,t)\end{bmatrix}; \sigma_e^2\mathbf{P}(t|\,t)\right) \tag{10.19}$$

kde

$$\begin{aligned} \widehat{\boldsymbol{\theta}}(t|\,t) &= \widehat{\boldsymbol{\theta}}(t|\,t-1) + \mathbf{k}_\theta \varepsilon(t|\,t-1) \\ \widehat{\mathbf{x}}(t|\,t) &= \widehat{\mathbf{x}}(t|\,t-1) + \mathbf{k}_x \varepsilon(t|\,t-1) \\ \varepsilon(t|\,t-1) &= y(t) - \hat{y}(t|\,t-1) \end{aligned} \tag{10.20}$$

a

$$\mathbf{P}(t|\,t) = \mathbf{P}(t|\,t-1) - \frac{\mathbf{P}(t|\,t-1)\mathbf{h}^T(t)\mathbf{h}(t)\mathbf{P}(t|\,t-1)}{1+\mathbf{h}(t)\mathbf{P}(t|\,t-1)\mathbf{h}^T(t)} \tag{10.21}$$

Kalmanův zisk datového kroku $\mathbf{k}_\theta(t)$ a $\mathbf{k}_x(t)$ je roven

$$\mathbf{k}(t) = \begin{bmatrix} \mathbf{k}_\theta(t) \\ \mathbf{k}_x(t) \end{bmatrix} = \frac{\mathbf{P}(t|\,t-1)\mathbf{h}^T(t)}{1+\mathbf{h}(t)\mathbf{P}(t|\,t-1)\mathbf{h}^T(t)}$$

Tím je proveden datový krok algoritmu. Časový krok provedeme podle stavové rovnice vývoje složeného stavu (10.7). Prediktivní podmíněná hustota je potom rovna

$$p\left(\mathbf{z}(t+1)|\;\mathcal{D}^t\right) = p\left(\begin{bmatrix} \boldsymbol{\theta}(t+1) \\ \mathbf{x}(t+1) \end{bmatrix} \middle|\; \mathcal{D}^t\right) = \mathcal{N}\left(\begin{bmatrix} \widehat{\boldsymbol{\theta}}(t+1|\,t) \\ \widehat{\mathbf{x}}(t+1|\,t) \end{bmatrix};\, \sigma_e^2 \mathbf{P}(t+1|\,t)\right) \tag{10.22}$$

kde

$$\widehat{\mathbf{z}}(t+1|\,t) = \begin{bmatrix} \widehat{\boldsymbol{\theta}}(t+1|\,t) \\ \widehat{\mathbf{x}}(t+1|\,t) \end{bmatrix} = \mathbf{F}(t) \begin{bmatrix} \widehat{\boldsymbol{\theta}}(t|\,t) \\ \widehat{\mathbf{x}}(t|\,t) \end{bmatrix} + \mathbf{G}(t) y(t) \tag{10.23}$$

a

$$\mathbf{P}(t+1|\,t) = \mathbf{F}(t)\mathbf{P}(t|\,t)\mathbf{F}^T(t) + \begin{bmatrix} \mathbf{V}(t) & 0 \\ 0 & 0 \end{bmatrix} \tag{10.24}$$

Popsaný datový a časový krok uzavírají algoritmus současného odhadování stavů a parametrů ARMAX modelu v pozorovatelném kanonickém tvaru, přičemž za předpokladu normality stačí aktualizovat pouze střední hodnoty a kovarianční matici.

10.1.3 Stochasticky optimální řízení

V tomto odstavci odvodíme stochasticky optimální řízení pozorovatelné realizace ARMAX modelu. Vyjdeme z výsledků předchozího odstavce a budeme předpokládat, že stav i parametry modelu jsou náhodné proměnné, při čemž známe jejich podmíněné střední hodnoty a podmíněné kovarianční matice.

Pozorovatelný kanonický tvar ARMAX modelu je popsán stavovou rovnicí (10.3)

$$\begin{aligned} \mathbf{x}(t+1) &= \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)u(t) + \mathbf{C}(t)e(t) \\ y(t) &= \mathbf{h}_x \mathbf{x}(t) + d(t)u(t) + e(t) \end{aligned}$$

Počáteční podmínky tohoto stavového modelu nemusí být známy přesně, předpokládáme, že počáteční podmínka je také náhodná proměnná a známe pouze její střední hodnotu a kovarianční matici $p(\mathbf{x}(0)) \sim \mathcal{N}(\widehat{\mathbf{x}}(0),\, \mathbf{P}_x(0))$. Naším úkolem bude určit optimální řízení $u^*(t)$, které minimalizuje kritérium kvality řízení

$$J = \mathcal{E}\left\{\mathbf{x}^T(N)\mathbf{S}\mathbf{x}(N) + \sum_{t=0}^{N-1} \mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + r u^2(t) | \mathcal{D}\right\} \tag{10.25}$$

Pozitivně semidefinitní matice $\mathbf{S}$, $\mathbf{Q}$ a nezápornou váhu r používáme jako ladicí parametry pro získání celkově vyhovujících odezev optimálního systému. Kritérium kvality řízení je rovno podmíněné střední hodnotě, kde v podmínce jsou změřená vstupní a výstupní data $u(t)$ a $y(t)$, což je v předchozím vztahu schematicky naznačeno v podmínce jako data $\mathcal{D}$.

Protože data získáváme postupně v diskrétních časových okamžicích t, je třeba jednotlivé členy v kritériu podmiňovat jinou množinou dat. Proto správný tvar kritéria kvality řízení stochastického systému je roven

$$J = \mathcal{E}\left\{\mathbf{x}^T(N)\mathbf{S}\mathbf{x}(N)|\mathcal{D}^{N-1}\right\} + \sum_{t=0}^{N-1} \mathcal{E}\left\{\mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + ru^2(t)|u(t), \mathcal{D}^{t-1}\right\} \quad (10.26)$$

Pokud v kritériu střední hodnotu podmiňujeme jinou množinou dat, dostaneme buď nerealizovatelný zákon řízení nebo nevyužíváme všechna dostupná data a proto řídíme pouze suboptimálně. Je zřejmé, že kritérium kvality řízení je funkcí počátečního stavu $\mathbf{x}(0)$, který je náhodnou proměnnou ($p(\mathbf{x}(0)) \sim \mathcal{N}(\hat{\mathbf{x}}(0), \mathbf{P}_x(0))$), počtu kroků řízení N a posloupnosti řízení $u(0)$ až $u(N)$, proto $J = J(\hat{\mathbf{x}}(0), \mathbf{P}_x(0), N, u(0), u(1), \ldots, u(N))$. Hodnota kritéria vedle toho závisí také na vlastnostech šumu procesu (jeho rozptylu) a také na jednotlivých realizacích výstupu $y(t)$, které ovlivňují průběžné odhady stavů a parametrů systému.

Optimální hodnota kritéria je tedy rovna

$$J^* = \min_{u(0),\ldots,u(N-1)} \left[\mathcal{E}\left\{\mathbf{x}^T(N)\mathbf{S}\mathbf{x}(N)|u(N), \mathcal{D}^{N-1}\right\} + \sum_{t=0}^{N-1} \mathcal{E}\left\{\mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + ru^2(t)|u(t), \mathcal{D}^{t-1}\right\} \right] \quad (10.27)$$

V [Aström 70] bylo ukázáno, že komutují operace minimalizace vzhledem k řízení a operace podmíněné střední hodnoty. Proto optimální kritérium můžeme psát ve tvaru

$$\begin{aligned} J^* = \min_{u(0)} \Big[& \mathcal{E}\left\{\mathbf{x}^T(0)\mathbf{Q}\mathbf{x}(0) + ru^2(0) + \right. \quad (10.28) \\ & \min_{u(1)} \Big[\mathcal{E}\left\{\mathbf{x}^T(1)\mathbf{Q}\mathbf{x}(1) + ru^2(1) + \ldots \right. \\ & \min_{u(N-1)} \Big[\mathcal{E}\left\{\mathbf{x}^T(N-1)\mathbf{Q}\mathbf{x}(N-1) + ru^2(N-1) + \right. \\ & \left. \left. \left. \mathcal{E}\left\{\mathbf{x}^T(N)\mathbf{S}\mathbf{x}(N)|u(N), \mathcal{D}^{N-1}\right\} |u(N-1), \mathcal{D}^{N-2}\right\} \Big] \ldots |u(1), \mathcal{D}^0\right\} \Big] |u(0)\right\} \Big] \end{aligned}$$

Abychom mohli využít průběžné odhady stavů a parametrů, je třeba kritérium počítat rekurzivně. Pro získání rekurzivního tvaru kritéria si zavedeme optimální funkci $\mathcal{V}_N(\mathbf{x}(t), t)$

$$\mathcal{V}_N(\mathbf{x}(t), t) = \min_{u(t),\ldots,u(N-1)} \left[\mathbf{x}^T(N)\mathbf{S}\mathbf{x}(N) + \sum_{\tau=t}^{N-1} \left(\mathbf{x}^T(\tau)\mathbf{Q}\mathbf{x}(\tau) + ru^2(\tau)\right) \right]$$

a její podmíněnou střední hodnotu

$$\begin{aligned} V_N(t) &= \mathcal{E}\left\{\mathcal{V}_N(\mathbf{x}(t), t)|\mathcal{D}^{t-1}\right\} = \\ & \min_{u(t),\ldots,u(N-1)} \left[\mathcal{E}\left\{\mathbf{x}^T(N)\mathbf{S}\mathbf{x}(N)|\mathcal{D}^{N-1}\right\} + \right. \\ & \left. \sum_{\tau=t}^{N-1} \mathcal{E}\left\{\mathbf{x}^T(\tau)\mathbf{Q}\mathbf{x}(\tau) + ru^2(\tau)|u(\tau), \mathcal{D}^{\tau-1}\right\} \right] \end{aligned}$$

Optimální funkce $\mathcal{V}_N(\mathbf{x}(t), t)$ je rovna hodnotě kritéria při počátečním čase $\tau = t$, proto je funkcí stavu v čase t, který je ale náhodnou proměnnou. My známe podmíněnou střední hodnotu a podmíněnou kovarianční matici této náhodné proměnné, kde v podmínce jsou dostupná data (to je stará data a současný vstup) $p(\mathbf{x}(t)|u(t), \mathcal{D}^{t-1}) \sim \mathcal{N}(\widehat{\mathbf{x}}(t|t-1),\ \mathbf{P}(t|t-1))$. Při tom jsme užili obvyklého značení $\widehat{\mathbf{x}}(t|\tau)$ pro odhad stavu v čase t podmíněném znalostí dat až do času τ. Proto $V_N(.)$, což je podmíněná střední hodnota optimální funkce $\mathcal{V}_N(.)$, je pro pevný koncový čas N závislá na podmíněné střední hodnotě a kovarianční matici stavu

$$V(t) = V(\widehat{\mathbf{x}}(t|t-1),\ \mathbf{P}(t|t-1), t).$$

V čase $t = 0$ je $V_N(0)$ rovno optimální hodnotě zvoleného kritéria kvality řízení (10.27). Optimální funkci $V_N(t)$ můžeme počítat rekurzivně

$$V_N(t) = \min_{u(t)}\ \mathcal{E}\left\{\mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + ru^2(t) + \mathcal{V}_N(\mathbf{x}(t+1), t+1)|u(t), \mathcal{D}^{t-1}\right\} \tag{10.29}$$

s počáteční (resp. koncovou) podmínkou

$$\begin{aligned} V_N(N) &= \mathcal{E}\left\{\mathbf{x}^T(N)\mathbf{S}\mathbf{x}(N)|u(N), \mathcal{D}^{N-1}\right\} = \\ &\quad \widehat{\mathbf{x}}^T(N|N-1)\ \mathbf{S}\ \widehat{\mathbf{x}}(N|N-1) + \mathrm{tr}\left[\mathbf{S}\mathbf{P}(N|N-1)\right]. \end{aligned}$$

Připomeňme, že $\mathrm{tr}(\mathbf{A})$ je stopa matice rovná součtu diagonálních prvků. Zde využijeme toho, že stopa součinu matic je komutativní. Abychom podmíněnou střední hodnotu v (10.29) vypočetli, musíme v $\mathcal{V}_N(\mathbf{x}(t+1), t+1)$ dosadit za stav $\mathbf{x}(t+1)$ ze stavové rovnice systému $\mathbf{x}(t+1) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) + \mathbf{C}e(t)$.

Protože kritérium je kvadratické a systém je lineární, má střední hodnota optimální funkce kvadratický tvar

$$V_N(t) = \widehat{\mathbf{x}}^T(t|t-1)\mathbf{G}(t)\widehat{\mathbf{x}}(t|t-1) + \boldsymbol{\gamma}^T(t)\widehat{\mathbf{x}}(t|t-1) + g(t) \tag{10.30}$$

kde $\mathbf{G}(t)$ je zatím neznámá maticová posloupnost, $\boldsymbol{\gamma}(t)$ je neznámá vektorová posloupnost a $g(t)$ je také zatím neznámá posloupnost. Ze vztahu pro koncovou podmínku optimální funkce okamžitě plynou koncové podmínky

$$\mathbf{G}(N) = \mathbf{S}, \qquad \boldsymbol{\gamma}(N) = \mathbf{0}, \qquad g(N) = \mathrm{tr}\left[\mathbf{S}\mathbf{P}(N|N-1)\right].$$

Pro výpočet optimálního řízení $u^*(t)$ provedeme minimalizaci v (10.29). Platí následující jednoduché úpravy

$$\begin{aligned} V_N(t) &= \min_{u(t)} \mathcal{E}\left\{\mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + ru^2(t) + V_N(\mathbf{x}(t+1), t+1)|\ u(t), \mathcal{D}^{t-1}\right\} \\ &= \min_{u(t)} \mathcal{E}\left\{\mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + ru^2(t) + \right. \\ &\quad (\mathbf{A}(t)\mathbf{x}(t)+\mathbf{B}(t)u(t)+\mathbf{C}(t)e(t))^T\,\mathbf{G}(t+1)\,(\mathbf{A}(t)\mathbf{x}(t)+\mathbf{B}(t)u(t)+\mathbf{C}(t)e(t)) \\ &\quad \left.+\boldsymbol{\gamma}^T(t+1)\,(\mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)u(t) + \mathbf{C}(t)e(t)) + g(t+1)|\ u(t), \mathcal{D}^{t-1}\right\} \\ &= \min_{u(t)}\ \mathcal{E}\left\{u(t)\left[r + \mathbf{B}^T(t)\mathbf{G}(t+1)\mathbf{B}(t)\right]u(t) + \right. \\ &\quad \left. u(t)\left[\mathbf{B}^T(t)\mathbf{G}(t+1)\,(\mathbf{A}(t)\mathbf{x}(t) + \mathbf{C}(t)e(t)) + \frac{1}{2}\mathbf{B}^T(t)\boldsymbol{\gamma}(t+1)\right] + \right. \end{aligned}$$

$$\left[\frac{1}{2}\mathbf{B}^T(t)\boldsymbol{\gamma}(t+1) + \left(\mathbf{x}^T(t)\mathbf{A}^T(t) + e(t)\mathbf{C}^T(t)\right)\mathbf{G}(t+1)\mathbf{B}(t)\right]u(t) +$$
$$\mathbf{x}^T(t)\left[\mathbf{Q} + \mathbf{A}^T(t)\mathbf{G}(t+1)\mathbf{A}(t)\right]\mathbf{x}(t) +$$
$$\mathbf{x}^T(t)\mathbf{A}^T(t)\mathbf{G}(t+1)\mathbf{C}(t)e(t) + e(t)\mathbf{C}^T(t)\mathbf{G}(t+1)\mathbf{A}(t)\mathbf{x}(t) +$$
$$e(t)\mathbf{C}^T(t)\mathbf{G}(t+1)\mathbf{C}(t)e(t) + \boldsymbol{\gamma}^T(t+1)\left[\mathbf{A}(t)\mathbf{x}(t) + \mathbf{C}(t)e(t)\right] +$$
$$g(t+1) \quad |u(t), \mathcal{D}^{t-1}\Big\}$$

Nyní naznačíme výpočet podmíněné střední hodnoty jednotlivých členů v předchozím výrazu

$$\begin{aligned} V_N(t) \;=\; \min_{u(t)} \Big[& \mathcal{E}\left\{\mathbf{x}^T(t)\left(\mathbf{Q} + \mathbf{A}^T(t)\mathbf{G}(t+1)\mathbf{A}(t)\right)\mathbf{x}(t)\right\} + \\ & u(t)\left(r + \mathcal{E}\left\{\mathbf{B}^T(t)\mathbf{G}(t+1)\mathbf{B}(t)\right\}\right)u(t) + \\ & \mathcal{E}\left\{\mathbf{x}^T(t)\mathbf{A}^T(t)\mathbf{G}(t+1)\mathbf{B}(t) + \frac{1}{2}\boldsymbol{\gamma}^T(t+1)\mathbf{B}(t)\right\}u(t) + \\ & u(t)\mathcal{E}\left\{\mathbf{B}^T(t)\mathbf{G}(t+1)\mathbf{A}(t)\mathbf{x}(t) + \frac{1}{2}\mathbf{B}^T(t)\boldsymbol{\gamma}(t+1)\right\} + \\ & \boldsymbol{\gamma}^T(t+1)\,\mathcal{E}\left\{\mathbf{A}(t)\mathbf{x}(t)\right\} + \\ & \mathcal{E}\left\{e(t)\mathbf{C}^T(t)\mathbf{G}(t+1)\mathbf{C}(t)e(t) + \boldsymbol{\gamma}^T(t+1)\mathbf{C}(t)e(t)\right\} + g(t+1)\Big] \end{aligned} \tag{10.31}$$

kde všechny podmíněné střední hodnoty jsou podmíněny daty $(u(t), \mathcal{D}^{t-1})$. Tuto podmíněnost budeme v dalších výrazech předpokládat a nebudeme ji pro jednoduchost vždy uvádět. Zatím nebudeme provádět výpočet podmíněné střední hodnoty předchozích výrazů, poznamenejme zde pouze, že v předchozích výrazech se vyskytují součiny tří a čtyř náhodných proměnných a proto při výpočtu jejich střední hodnoty budeme potřebovat třetí a čtvrté momenty jejich rozdělení. To v případě normálního rozdělení nebude nepřekonatelný problém.

Minimalizaci předchozího výrazu provedeme doplněním na úplný čtverec. Platí

$$\begin{aligned} V_N(t) \;=\; \min_{u(t)} \Big[& (u(t) - u^*(t))^T \left(r + \mathcal{E}\left\{\mathbf{B}^T(t)\mathbf{G}(t+1)\mathbf{B}(t)\right\}\right)(u(t) - u^*(t)) \\ & + \mathcal{E}\left\{\mathbf{x}^T(t)\left(\mathbf{Q} + \mathbf{A}^T(t)\mathbf{G}(t+1)\mathbf{A}(t)\right)\mathbf{x}(t)\right\} + \\ & \mathcal{E}\left\{e(t)\mathbf{C}^T(t)\mathbf{G}(t+1)\mathbf{C}(t)e(t) + \boldsymbol{\gamma}^T(t+1)\mathbf{C}\mathbf{e}(t)\right\} + g(t+1) + \\ & \boldsymbol{\gamma}^T(t+1)\,\mathcal{E}\left\{\mathbf{A}(t)\mathbf{x}(t)\right\} - u^*(t)\left(r + \mathcal{E}\left\{\mathbf{B}^T(t)\mathbf{G}(t+1)\mathbf{B}(t)\right\}\right)u^*(t)\Big] \end{aligned}$$

Porovnáním lineárních členů dostaneme

$$\begin{aligned} -u^T(t)\left[r + \mathcal{E}\left\{\mathbf{B}^T\mathbf{G}(t+1)\mathbf{B}\right\}\right]u^*(t) \;=\; & \\ u^T(t)\left[\mathcal{E}\left\{\mathbf{B}^T(t)\mathbf{G}(t+1)\mathbf{A}(t)\mathbf{x}(t) + \frac{1}{2}\mathbf{B}^T(t)\boldsymbol{\gamma}(t+1)\right\}\right] & \end{aligned}$$

Odtud plyne optimální řízení

$$\begin{aligned} u^*(t) \;=\; & -\left[r + \mathcal{E}\left\{\mathbf{B}^T(t)\mathbf{G}(t+1)\mathbf{B}(t)\right\}\right]^{-1} \times \\ & \times\, \mathcal{E}\left\{\mathbf{B}^T(t)\mathbf{G}(t+1)\mathbf{A}(t)\mathbf{x}(t) + \frac{1}{2}\mathbf{B}^T(t)\boldsymbol{\gamma}(t+1)\right\} \end{aligned} \tag{10.32}$$

kde opět naznačené střední hodnoty jsou podmíněny daty $(u(t), \mathcal{D}^{t-1})$. Optimální řízení podle předchozího vztahu dosadíme do optimální funkce a dostaneme vztahy pro výpočet maticové posloupnosti $\mathbf{G}(t)$, vektorové posloupnosti $\boldsymbol{\gamma}(t)$ a posloupnosti $g(t)$.

$$\begin{aligned} V_N(t) &= \hat{\mathbf{x}}^T(t|t-1)\mathbf{G}(t)\hat{\mathbf{x}}(t|t-1) + \boldsymbol{\gamma}^T(t)\hat{\mathbf{x}}^T(t|t-1) + g(t) = \\ &\quad \mathcal{E}\left\{e(t)\mathbf{C}^T(t)\mathbf{G}(t+1)\mathbf{C}(t)e(t) + \boldsymbol{\gamma}^T(t+1)\mathbf{C}(t)e(t)\right\} + g(t+1) + \\ &\quad \mathcal{E}\left\{\mathbf{x}^T(t)\left(\mathbf{Q} + \mathbf{A}^T(t)\mathbf{G}(t+1)\mathbf{A}(t)\right)\mathbf{x}(t)\right\} - \\ &\mathcal{E}\left\{\mathbf{x}^T(t)\mathbf{A}^T(t)\mathbf{G}(t+1)\mathbf{B}(t) + \frac{1}{2}\boldsymbol{\gamma}^T(t+1)\mathbf{B}(t)\right\}\left(r + \mathcal{E}\left\{\mathbf{B}^T(t)\mathbf{G}(t+1)\mathbf{B}(t)\right\}\right)^{-1} \times \\ &\mathcal{E}\left\{\mathbf{B}^T(t)\mathbf{G}(t+1)\mathbf{A}(t)\mathbf{x}(t) + \frac{1}{2}\mathbf{B}^T(t)\boldsymbol{\gamma}(t+1)\right\} + \boldsymbol{\gamma}^T(t+1)\ \mathcal{E}\left\{\mathbf{A}(t)\mathbf{x}(t)\right\} \end{aligned} \tag{10.33}$$

Z předchozího vztahu dostaneme rekurentní vztahy pro výpočet maticové posloupnosti $\mathbf{G}(t)$, vektorové posloupnosti $\boldsymbol{\gamma}(t)$ a posloupnosti $g(t)$ až po výpočtu podmíněných středních hodnot v předchozích výrazech. Těmto výpočtům věnujeme celý následující odstavec.

10.1.4 Střední hodnoty součinu závislých náhodných veličin

Momenty centrovaných náhodných veličin můžeme vypočíst pomocí charakteristické funkce náhodné veličiny. Charakteristická funkce $\varphi(t)$ náhodné veličiny x s hustotou pravděpodobnosti $p(x)$ je definována vztahem

$$\varphi(t) = \int_{-\infty}^{\infty} e^{jtx} p(x) dx \tag{10.34}$$

Existuje-li k-tý moment, potom jej můžeme počítat podle vztahu

$$\mathcal{E}\left\{x^k\right\} = \frac{1}{j^k}\varphi^{(k)}(0) \tag{10.35}$$

kde $\varphi^{(k)}(0)$ je k-tá derivace charakteristické funkce podle proměnné t počítaná v bodě $t = 0$.

V případě náhodného vektoru $\mathbf{x}$ s normálním rozdělením se střední hodnotou $\hat{\mathbf{x}} = \nu$ a kovariancí $\mathbf{P}$ je charakteristická funkce $\varphi(\mathbf{t})$ rovna

$$\varphi(\mathbf{t}) = e^{(j\mathbf{t}^T\nu - \frac{1}{2}\mathbf{t}^T\mathbf{P}\mathbf{t})}. \tag{10.36}$$

Vyšší momenty můžeme pak počítat podle vztahu

$$j^{(r_1+r_2+\ldots+r_n)}\mathcal{E}\left\{x_1^{r_1}, \ldots, x_n^{r_n}\right\} = \frac{\partial}{\partial t_1^{r_1} \ldots \partial t_n^{r_n}}\varphi(\mathbf{t})\ \Big|_{\mathbf{t}=\mathbf{0}}. \tag{10.37}$$

Použitím symbolických výpočtů vypočteme podle předchozího vztahu střední hodnotu součinu čtyř centrovaných náhodných veličin x_1 až x_4. Platí

$$\mathcal{E}\left\{x_1\ x_2\ x_3\ x_4\right\} = P_{12}\ P_{34} + P_{13}\ P_{24} + P_{14}\ P_{23}$$

a podobně

$$\begin{aligned}\mathcal{E}\left\{x_1^2\ x_2^2\right\} &= P_{11}\ P_{22} + 2\ P_{12}^2 \\ \mathcal{E}\left\{x_1^2\ x_2\ x_3\right\} &= P_{12}\ P_{23} + 2\ P_{12}\ P_{13}\end{aligned}$$

Uvažujme nyní závislé náhodné veličiny x, y, z, w, které mají normální rozdělení, pak

$$p\left(\left[\begin{array}{c} x \\ y \\ z \\ w \end{array}\right]\right) = \mathcal{N}\left(\left[\begin{array}{c} \hat{x} \\ \hat{y} \\ \hat{z} \\ \hat{w} \end{array}\right]; \left[\begin{array}{cccc} P_{xx} & P_{xy} & P_{xz} & P_{xw} \\ P_{yx} & P_{yy} & P_{yz} & P_{yw} \\ P_{zx} & P_{zy} & P_{zz} & P_{zw} \\ P_{wx} & P_{wy} & P_{wz} & P_{ww} \end{array}\right]\right) \tag{10.38}$$

Potom platí pro necentrované střední hodnoty součinu proměnných následující vztahy

$$\begin{aligned}\mathcal{E}\{x\ y\} &= \hat{x}\ \hat{y} + P_{xy} \\ \mathcal{E}\{x\ y\ z\} &= \hat{x}\ \hat{y}\ \hat{z} + \hat{x}\ P_{yz} + \hat{y}\ P_{xz} + \hat{z}\ P_{xy} \\ \mathcal{E}\{x\ y\ z\ w\} &= \hat{x}\ \hat{y}\ \hat{z}\ \hat{w} + \hat{x}\ \hat{y}\ P_{zw} + \hat{x}\ \hat{z}\ P_{yw} + \hat{x}\ \hat{w}\ P_{yz} + \hat{y}\ \hat{z}\ P_{xw} + \\ &\quad \hat{y}\ \hat{w}\ P_{xz} + \hat{z}\ \hat{w}\ P_{xy} + P_{xy}\ P_{zw} + P_{xz}\ P_{yw} + P_{xw}\ P_{yz}\end{aligned} \tag{10.39}$$

Podobně platí

$$\begin{aligned}\mathcal{E}\left\{x^2\ y^2\right\} &= \hat{x}^2\ \hat{y}^2 + \hat{x}^2\ P_{yy} + \hat{y}^2\ P_{xx} \\ &\quad +4\ \hat{x}\ \hat{y}\ P_{xy} + 2\ P_{xy}^2 + P_{xx}\ P_{yy} \\ \mathcal{E}\left\{x^2\ y\ z\right\} &= \hat{x}^2\ \hat{y}\hat{z} + 2\ \hat{x}\ \hat{y}\ P_{zx} + 2\ \hat{x}\ \hat{z}\ P_{yx} + \hat{x}^2\ P_{yz} + \hat{y}\ \hat{z}\ P_{xx} \\ &\quad +P_{xx}\ P_{yz} + 2\ P_{xy}\ P_{xz}\end{aligned} \tag{10.40}$$

Předchozí vztahy snadno dokážeme, napíšeme-li každou náhodnou proměnnou ve tvaru $x = \hat{x} + \tilde{x}$, kde $\hat{x}$ je střední hodnota náhodné proměnné x a $\tilde{x}$ je její odchylka od střední hodnoty (která má samozřejmě nulovou střední hodnotu). Pak platí

$$\mathcal{E}\{x\ y\} = \mathcal{E}\{(\hat{x} + \tilde{x})\ (\hat{y} + \tilde{y})\} = \hat{x}\ \hat{y} + \mathcal{E}\{\tilde{x}\ \tilde{y}\} = \hat{x}\ \hat{y} + P_{xy}$$

Podobně

$$\begin{aligned}\mathcal{E}\{x\ y\ z\} &= \mathcal{E}\{(\hat{x} + \tilde{x})\ (\hat{y} + \tilde{y})\ (\hat{z} + \tilde{z})\} \\ &= \hat{x}\ \hat{y}\ \hat{z} + \hat{x}\ \mathcal{E}\{\tilde{y}\ \tilde{z}\} + \hat{y}\ \mathcal{E}\{\tilde{x}\tilde{z}\} + \hat{z}\ \mathcal{E}\{\tilde{x}\tilde{y}\} + P_{xyz} \\ &= \hat{x}\ \hat{y}\ \hat{z} + \hat{x}\ P_{yz} + \hat{y}\ P_{xz} + \hat{z}\ P_{xy}\end{aligned}$$

protože třetí moment P_{xyz} nazývaný šikmost rozdělení je u normálně rozdělených veličin nulový. Podobně počítáme střední hodnotu součinu čtyř normálně rozdělených veličin. Platí

$$\begin{aligned}\mathcal{E}\{x\ y\ z\ w\} &= \mathcal{E}\{(\hat{x} + \tilde{x})\ (\hat{y} + \tilde{y})\ (\hat{z} + \tilde{z})\ (\hat{w} + \tilde{w})\} \\ &= \hat{x}\ \hat{y}\ \hat{z}\ \hat{w} + \hat{x}\ \hat{y}\ \mathcal{E}\{\tilde{z}\tilde{w}\} + \hat{x}\ \hat{z}\ \mathcal{E}\{\tilde{y}\tilde{w}\} + \hat{x}\ \hat{w}\ \mathcal{E}\{\tilde{y}\tilde{z}\} + \\ &\quad \hat{y}\ \hat{z}\ \mathcal{E}\{\tilde{x}\tilde{w}\} + \hat{y}\ \hat{w}\ \mathcal{E}\{\tilde{x}\tilde{z}\} + \hat{z}\ \hat{w}\ \mathcal{E}\{\tilde{x}\tilde{y}\} + \mathcal{E}\{\tilde{x}\tilde{y}\tilde{z}\tilde{w}\} \\ &= \hat{x}\ \hat{y}\ \hat{z}\ \hat{w} + \hat{x}\ \hat{y}\ P_{zw} + \hat{x}\ \hat{z}\ P_{yw} + \hat{x}\ \hat{w}\ P_{yz} + \hat{y}\ \hat{z}\ P_{xw} + \\ &\quad +P_{xy}\ P_{zw} + P_{xz}\ P_{yw} + P_{xw}\ P_{yz}\end{aligned}$$

kde jsme opět využili toho, že první i třetí moment centrovaných normálních veličin je nulový.

Obdobně vypočteme střední hodnotu matice

$$\begin{aligned}\mathcal{E}\left\{xy\mathbf{a}\mathbf{a}^T\right\} &= \hat{x}\hat{y}\hat{\mathbf{a}}\hat{\mathbf{a}}^T + \hat{x}\hat{\mathbf{a}}\mathbf{P}_{ya} + \hat{x}\mathbf{P}_{ay}\hat{\mathbf{a}}^T + \hat{y}\hat{\mathbf{a}}\mathbf{P}_{xa} + \hat{y}\mathbf{P}_{ax}\hat{\mathbf{a}}^T + \\ &\quad \hat{\mathbf{a}}\hat{\mathbf{a}}^T\mathbf{P}_{xy} + \hat{x}\hat{y}\mathbf{P}_{aa} + \mathbf{P}_{aa}\mathbf{P}_{xy} + \mathbf{P}_{ay}\mathbf{P}_{xa} + \mathbf{P}_{ax}\mathbf{P}_{ya}\end{aligned}$$

kde x, y jsou skalární náhodné veličiny, $\mathbf{a}$ je náhodný vektor. Všechny veličiny jsou vzájemně závislé a jejich závislost je vyjádřena vzájemnými kovariancemi.

Tyto vztahy využijeme při výpočtu optimálního řízení a tomuto problému věnujeme následující odstavec.

10.1.5 Výpočet optimálního řízení

Ve vztahu (10.32) pro optimální řízení při respektování neurčitosti ve stavech i parametrech a ve vztahu pro optimální funkci (10.33) se vyskytují střední hodnoty následujících výrazů

$$\begin{aligned}&\mathcal{E}\left\{e(t)\mathbf{C}^T(t)\mathbf{G}(t+1)\mathbf{C}(t)e(t)\right\};\\ &\mathcal{E}\left\{\boldsymbol{\gamma}^T(t+1)\mathbf{A}(t)\mathbf{x}(t) + \boldsymbol{\gamma}^T(t+1)\mathbf{C}(t)e(t)\right\};\\ &\mathcal{E}\left\{\mathbf{B}^T(t)\mathbf{G}(t+1)\mathbf{B}(t)\right\};\\ &\mathcal{E}\left\{\mathbf{x}^T(t)\left(\mathbf{Q} + \mathbf{A}^T(t)\mathbf{G}(t+1)\mathbf{A}(t)\right)\mathbf{x}(t)\right\};\\ &\mathcal{E}\left\{\mathbf{B}^T(t)\mathbf{G}(t+1)\mathbf{A}(t)\mathbf{x}(t) + \frac{1}{2}\mathbf{B}^T(t)\boldsymbol{\gamma}(t+1)\right\};\\ &\mathcal{E}\left\{\mathbf{x}^T(t)\mathbf{A}^T(t)\mathbf{G}(t+1)\mathbf{B}(t) + \frac{1}{2}\boldsymbol{\gamma}^T(t+1)\mathbf{B}(t)\right\}.\end{aligned}$$

Je zřejmé, že poslední dva výrazy jsou pouze transpozicí jeden druhého. S použitím vztahů odvozených v předchozím odstavci budeme počítat uvedené střední hodnoty součinu náhodných vektorů a matic. Při tom parametry $\boldsymbol{\theta}(t)$ ARMAX modelu jsou složeny z koeficientů $b_i(t)$ a $a_i(t)$. Dále pro jednoduchost budeme vynechávat argument času t. Platí

$$\boldsymbol{\theta}^T = \left[\begin{array}{cccccccc} b_0 & b_1 & \dots & b_n & a_1 & \dots & a_n \end{array}\right] = \left[\begin{array}{ccc} b_0 & \mathbf{b}^T & \mathbf{a}^T \end{array}\right] \tag{10.41}$$

Složený vektor parametrů a stavů $\mathbf{z} = \left[\begin{array}{cc} \boldsymbol{\theta}^T, & \mathbf{x}^T \end{array}\right]^T$ má normální rozdělení

$$p\left(\left[\begin{array}{c}\boldsymbol{\theta}\\ \mathbf{x}\end{array}\right]\right) = p\left(\left[\begin{array}{c} b_0\\ \mathbf{b}\\ \mathbf{a}\\ \mathbf{x}\end{array}\right]\right) = \mathcal{N}\left(\left[\begin{array}{c}\hat{b}_0\\ \hat{\mathbf{b}}\\ \hat{\mathbf{a}}\\ \hat{\mathbf{x}}\end{array}\right];\ \left[\begin{array}{cccc} P_{b_0b_0} & P_{b_0b} & P_{b_0a} & P_{b_0x}\\ P_{bb_0} & P_{bb} & P_{ba} & P_{bx}\\ P_{ab_0} & P_{ab} & P_{aa} & P_{ax}\\ P_{xb_0} & P_{xb} & P_{xa} & P_{xx}\end{array}\right]\right). \tag{10.42}$$

Připomeňme, že vektor $\mathbf{B} = (\mathbf{b} - b_0\ \mathbf{a})$, vektor $\mathbf{C} = (\mathbf{c} - \mathbf{a})$ a součin matice systému $\mathbf{A}$ s vektorem stavu $\mathbf{x}$ můžeme vyjádřit ve tvaru

$$\mathbf{A}\mathbf{x} = \mathbf{w} - \mathbf{a}x_1 \tag{10.43}$$

kde vektor $\mathbf{w}$ je roven

$$\mathbf{w} = \left[\begin{array}{cccc} x_2 & \dots & x_n & 0 \end{array} \right]^T$$

Zřejmě platí

$$x_1 = \mathbf{h}_x \mathbf{x} = \mathbf{x}^T \mathbf{h}_x^T, \qquad \mathbf{w} = \overline{\mathbf{F}} \mathbf{x} \tag{10.44}$$

kde

$$\mathbf{h}_x^T = \left[\begin{array}{c} 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{array} \right]; \qquad \overline{\mathbf{F}} = \left[\begin{array}{ccccc} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & & 0 \\ & \vdots & \ddots & & \\ 0 & 0 & 0 & & 1 \\ 0 & 0 & 0 & \dots & 0 \end{array} \right]$$

Nyní budeme počítat střední hodnoty jednotlivých výrazů. Platí

$$\begin{aligned} \mathcal{E}\left\{ e(t)\mathbf{C}^T(t)\mathbf{G}(t+1)\mathbf{C}(t)e(t) \right\} &= \operatorname{tr}\left[\mathbf{G}(t+1)\mathcal{E}\left\{ \mathbf{C}(t)e^2(t)\mathbf{C}^T(t) \right\} \right] \\ &= \operatorname{tr}\left[\mathbf{G}(t+1)\mathcal{E}\left\{ (\mathbf{c}-\mathbf{a})\, e^2(t)\, (\mathbf{c}-\mathbf{a})^T \right\} \right] \\ &= \operatorname{tr}\left[\mathbf{G}(t+1)\left(\sigma_e^2 \widehat{\mathbf{C}}\widehat{\mathbf{C}}^T + \sigma_e^2 \mathbf{P}_{aa} \right) \right] \\ = \sigma_e^2 \widehat{\mathbf{C}}^T \mathbf{G}(t+1)\widehat{\mathbf{C}} + \sigma_e^2 \operatorname{tr}\left[\mathbf{G}(t+1)\mathbf{P}_{aa} \right] \end{aligned} \tag{10.45}$$

kde všechny podmíněné střední hodnoty a podmíněné kovariance mají argument $(t|t-1)$, jsou to tedy odhady v čase t, podmíněné daty až do času $t-1$. Tyto argumenty budeme dále pro jednoduchost vynechávat. Zde jsme použili následující úpravy

$$\begin{aligned} &\mathcal{E}\left\{ (\mathbf{c}-\mathbf{a})\, \mathbf{e}\mathbf{e}^T\, (\mathbf{c}-\mathbf{a})^T \right\} = \\ &\mathcal{E}\left\{ \mathbf{c}\mathbf{e}\mathbf{e}^T\mathbf{c}^T - \mathbf{a}\mathbf{e}\mathbf{e}^T\mathbf{c}^T - \mathbf{c}\mathbf{e}\mathbf{e}^T\mathbf{a}^T + \mathbf{a}\mathbf{e}\mathbf{e}^T\mathbf{a}^T \right\} = \\ &\mathbf{c}\mathbf{P}_e\mathbf{c}^T - \hat{\mathbf{a}}\mathbf{P}_e\mathbf{c}^T - \mathbf{c}\mathbf{P}_e\hat{\mathbf{a}}^T + \hat{\mathbf{a}}\mathbf{P}_e\hat{\mathbf{a}}^T + \mathbf{P}_e\mathbf{P}_{aa} \end{aligned}$$

Předchozí výpočet byl jednoduchý, protože šum $e(t)$ je skalární veličina s rozptylem $\sigma_e^2 = \mathbf{P}_e$ a šum $e(t)$ navíc není korelovaný s parametry i stavy v čase t. Výraz $(\sigma_e^2 \operatorname{tr}\left[\mathbf{G}(t+1)\mathbf{P}_{aa}\right])$ představuje zvýšení kritéria vlivem nejistoty v parametrech. Pro střední hodnotu dalšího výrazu platí

$$\mathcal{E}\left\{ \boldsymbol{\gamma}^T(t+1)\left(\mathbf{A}(t)\mathbf{x}(t) + \mathbf{C}(t)e(t) \right) \right\} = \boldsymbol{\gamma}^T(t+1)\left(\widehat{\mathbf{A}}\hat{\mathbf{x}} - \mathbf{P}_{ax_1} \right) \tag{10.46}$$

Podobně budeme počítat střední hodnotu dalšího výrazu. Platí

$$\begin{aligned} \mathcal{E}\left\{ \mathbf{x}^T(t)\left(\mathbf{Q} + \mathbf{A}^T(t)\mathbf{G}(t+1)\mathbf{A}(t) \right)\mathbf{x}(t) \right\} &= \\ \mathcal{E}\left\{ \mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) \right\} &+ \mathcal{E}\left\{ \mathbf{x}^T(t)\mathbf{A}^T(t)\mathbf{G}(t+1)\mathbf{A}(t)\mathbf{x}(t) \right\} \end{aligned}$$

První člen na pravé straně spočteme snadno

$$\mathcal{E}\left\{ \mathbf{x}^T\mathbf{Q}\mathbf{x} \right\} = \mathcal{E}\left\{ \operatorname{tr}\left[\mathbf{Q}\mathbf{x}\mathbf{x}^T \right] \right\} = \hat{\mathbf{x}}^T\mathbf{Q}\hat{\mathbf{x}} + \operatorname{tr}\left[\mathbf{Q}\mathbf{P}_{xx} \right]$$

kde druhý člen na pravé straně opět zvyšuje pouze hodnotu kritéria vlivem nejistoty ve znalosti stavu a nemá vliv na řízení. Podobně

$$\begin{aligned} \mathcal{E}\left\{ \mathbf{x}^T(t)\mathbf{A}^T\mathbf{G}(t+1)\mathbf{A}\mathbf{x}(t) \right\} &= \operatorname{tr}\left(\mathbf{G}\ \mathcal{E}\left\{ \mathbf{A}\mathbf{x}\mathbf{x}^T\mathbf{A}^T \right\} \right) \\ &= \operatorname{tr}\left[\mathbf{G}\ \mathcal{E}\left\{ (\mathbf{w} - \mathbf{a}x_1)\ (\mathbf{w} - \mathbf{a}x_1)^T \right\} \right] \end{aligned}$$

Střední hodnotu v předchozím výrazu spočteme podle následujících vztahů

$$\mathcal{E}\left\{\mathbf{A}\mathbf{x}\mathbf{x}^T\mathbf{A}^T\right\} = \mathcal{E}\left\{x_1^2\mathbf{a}\mathbf{a}^T - \mathbf{w}\mathbf{a}^T x_1 - \mathbf{a}\mathbf{w}^T x_1 + \mathbf{w}\mathbf{w}^T\right\}$$

Zřejmě platí

$$\begin{aligned}
\mathcal{E}\left\{\mathbf{w}\mathbf{w}^T\right\} &= \hat{\mathbf{w}}\hat{\mathbf{w}}^T + \mathbf{P}_{ww} \\
\mathcal{E}\left\{\mathbf{w}\mathbf{a}^T x_1\right\} &= \hat{\mathbf{w}}\hat{\mathbf{a}}^T\hat{x}_1 + \hat{x}_1\mathbf{P}_{wa} + \hat{\mathbf{w}}\mathbf{P}_{x_1a} + \mathbf{P}_{wx_1}\hat{\mathbf{a}}^T \\
\mathcal{E}\left\{\mathbf{a}\mathbf{w}^T x_1\right\} &= \hat{\mathbf{a}}\hat{\mathbf{w}}^T\hat{x}_1 + \hat{x}_1\mathbf{P}_{aw} + \hat{\mathbf{a}}\mathbf{P}_{x_1w} + \mathbf{P}_{ax_1}\hat{\mathbf{w}}^T \\
\mathcal{E}\left\{x_1^2\mathbf{a}\mathbf{a}^T\right\} &= \hat{x}_1^2\hat{\mathbf{a}}\hat{\mathbf{a}}^T + \hat{x}_1^2\mathbf{P}_{aa} + 2\hat{x}_1\hat{\mathbf{a}}\mathbf{P}_{x_1a} + 2\hat{x}_1\mathbf{P}_{ax_1}\hat{\mathbf{a}}^T + \\
&\quad \hat{\mathbf{a}}\hat{\mathbf{a}}^T\mathbf{P}_{x_1x_1} + \mathbf{P}_{x_1x_1}\mathbf{P}_{aa} + 2\,\mathbf{P}_{ax_1}\mathbf{P}_{x_1a}
\end{aligned}$$

Potom je hledaná střední hodnota

$$\begin{aligned}
&\mathcal{E}\left\{\mathbf{x}^T(t)\mathbf{A}^T\mathbf{G}(t+1)\mathbf{A}\mathbf{x}(t)\right\} = \hat{\mathbf{x}}^T\widehat{\mathbf{A}}^T\mathbf{G}\widehat{\mathbf{A}}\hat{\mathbf{x}} + \\
&\mathrm{tr}\left[\mathbf{G}\left(4\,\hat{x}_1\hat{\mathbf{a}}\mathbf{P}_{x_1a} + \hat{x}_1^2\mathbf{P}_{aa} + \hat{\mathbf{a}}\hat{\mathbf{a}}^T\mathbf{P}_{x_1x_1} + \mathbf{P}_{x_1x_1}\mathbf{P}_{aa} + 2\,\mathbf{P}_{ax_1}\mathbf{P}_{x_1a}\right.\right. \\
&\left.\left.-\hat{\mathbf{a}}\mathbf{P}_{x_1w} - \mathbf{P}_{wx_1}\hat{\mathbf{a}}^T - \mathbf{P}_{ax_1}\hat{\mathbf{w}}^T - \hat{\mathbf{w}}\mathbf{P}_{x_1a} - \hat{x}_1(\mathbf{P}_{aw} + \mathbf{P}_{wa}) + \mathbf{P}_{ww}\right)\right]
\end{aligned}$$

Po úpravě dostaneme

$$\begin{aligned}
&\mathcal{E}\left\{\mathbf{x}^T(t)\mathbf{A}^T\mathbf{G}(t+1)\mathbf{A}\mathbf{x}(t)\right\} = \hat{\mathbf{x}}^T\widehat{\mathbf{A}}^T\mathbf{G}\widehat{\mathbf{A}}\hat{\mathbf{x}} + \\
&\hat{x}_1^2\,\mathrm{tr}\left[\mathbf{G}\mathbf{P}_{aa}\right] + \hat{x}_1\left(4\,\mathbf{P}_{x_1a}\mathbf{G}\hat{\mathbf{a}} - \mathrm{tr}\left[\mathbf{G}\left(\mathbf{P}_{wa} + \mathbf{P}_{aw}\right)\right]\right) - 2\,\mathbf{P}_{x_1a}\mathbf{G}\hat{\mathbf{w}} + \\
&-2\,\mathbf{P}_{x_1w}\mathbf{G}\hat{\mathbf{a}} + \mathbf{P}_{x_1x_1}\left(\hat{\mathbf{a}}^T\mathbf{G}\hat{\mathbf{a}} + \mathrm{tr}\left[\mathbf{G}\mathbf{P}_{aa}\right]\right) + 2\,\mathbf{P}_{x_1a}\mathbf{G}\mathbf{P}_{ax_1} + \mathrm{tr}\left[\mathbf{G}\mathbf{P}_{ww}\right]
\end{aligned}$$

Všechny členy kromě prvního pouze zvyšují kritérium vlivem nejistoty a nemají vliv na optimální řízení. Pomocí vektoru $\mathbf{h}_x$ a matice $\mathbf{A}$ vyjádříme předchozí výraz jako součet kvadratické a lineární formy v $\hat{\mathbf{x}}$ a absolutního členu

$$\begin{aligned}
\mathcal{E}\left\{\mathbf{x}^T(t)\mathbf{A}^T(t)\mathbf{G}(t+1)\mathbf{A}(t)\mathbf{x}(t)\right\} &= \hat{\mathbf{x}}^T\left(\widehat{\mathbf{A}}^T\mathbf{G}\widehat{\mathbf{A}} + \mathbf{h}_x\,\mathrm{tr}\left[\mathbf{G}\mathbf{P}_{aa}\right]\mathbf{h}_x^T\right)\hat{\mathbf{x}} + \\
&\left(2\,\mathbf{P}_{x_1a}\mathbf{G}\left(\hat{\mathbf{a}}\mathbf{h}_x^T - \mathbf{A}\right) - \mathrm{tr}\left[\mathbf{G}\left(\mathbf{P}_{wa} + \mathbf{P}_{aw}\right)\right]\mathbf{h}_x^T\right)\hat{\mathbf{x}} + \omega_1(t+1) \\
&= \hat{\mathbf{x}}^T\boldsymbol{\Omega}_1(t+1)\hat{\mathbf{x}} + \boldsymbol{\Omega}_2^T(t+1)\hat{\mathbf{x}} + \omega_1(t+1)
\end{aligned}$$

kde pomocná posloupnost $\omega_1(t+1)$ je rovna

$$\begin{aligned}
\omega_1(t+1) &= -2\,\mathbf{P}_{x_1w}\mathbf{G}\hat{\mathbf{a}} + \mathbf{P}_{x_1x_1}\left(\hat{\mathbf{a}}^T\mathbf{G}\hat{\mathbf{a}} + \mathrm{tr}\left[\mathbf{G}\mathbf{P}_{aa}\right]\right) + \\
&\quad 2\,\mathbf{P}_{x_1a}\mathbf{G}\mathbf{P}_{ax_1} + \mathrm{tr}\left[\mathbf{G}\mathbf{P}_{ww}\right].
\end{aligned}$$

a význam matice $\boldsymbol{\Omega}_1(t+1)$ a vektoru $\boldsymbol{\Omega}_2(t+1)$ je patrný z předchozího vztahu. Podobně upravíme další člen, který již ale ovlivňuje optimální řízení. Platí

$$\mathcal{E}\left\{\mathbf{B}^T(t)\mathbf{G}(t+1)\mathbf{B}(t)\right\} = \mathrm{tr}\left[\mathbf{G}(t+1)\,\mathcal{E}\left\{\mathbf{B}\mathbf{B}^T\right\}\right] \tag{10.47}$$

Při tom

$$\mathbf{B}\mathbf{B}^T = (\mathbf{b} - b_0\mathbf{a})\,(\mathbf{b} - b_0\mathbf{a})^T = \mathbf{b}\mathbf{b}^T - b_0\mathbf{a}\mathbf{b}^T - b_0\mathbf{b}\mathbf{a}^T + b_0^2\mathbf{a}\mathbf{a}^T$$

a proto

$$\begin{aligned}\mathcal{E}\left\{\mathbf{B}\mathbf{B}^T\right\} &= \hat{\mathbf{B}}\hat{\mathbf{B}}^T + \mathbf{P}_{bb} - \hat{\mathbf{a}}\mathbf{P}_{b_0b} - \mathbf{P}_{bb_0}\hat{\mathbf{a}}^T - \mathbf{P}_{ab_0}\hat{\mathbf{b}}^T - \hat{\mathbf{b}}\mathbf{P}_{b_0a} - \\ &\quad \hat{b}_0\left(\mathbf{P}_{ab} + \mathbf{P}_{ba}\right) + 2\,\hat{b}_0\hat{\mathbf{a}}\mathbf{P}_{b_0a} + 2\,\hat{b}_0\mathbf{P}_{ab_0}\hat{\mathbf{a}}^T + \hat{b}_0^2\mathbf{P}_{aa} + \\ &\quad \hat{\mathbf{a}}\hat{\mathbf{a}}^T\mathbf{P}_{b_0b_0} + \mathbf{P}_{b_0b_0}\mathbf{P}_{aa} + 2\,\mathbf{P}_{ab_0}\mathbf{P}_{b_0a}\end{aligned}$$

Hledaná střední hodnota je tedy

$$\begin{aligned}\mathcal{E}\left\{\mathbf{B}^T\mathbf{G}(t+1)\mathbf{B}\right\} &= \hat{\mathbf{B}}^T\mathbf{G}(t+1)\hat{\mathbf{B}} + \\ &\operatorname{tr}\Big[\mathbf{G}\Big(\mathbf{P}_{bb} - \hat{\mathbf{a}}\mathbf{P}_{b_0b} - \mathbf{P}_{bb_0}\hat{\mathbf{a}}^T - \mathbf{P}_{ab_0}\hat{\mathbf{b}}^T - \hat{\mathbf{b}}\mathbf{P}_{b_0a} - \hat{b}_0\left(\mathbf{P}_{ab} + \mathbf{P}_{ba}\right) + \\ &2\,\hat{b}_0\hat{\mathbf{a}}\mathbf{P}_{b_0a} + 2\,\hat{b}_0\mathbf{P}_{ab_0}\hat{\mathbf{a}}^T + \hat{b}_0^2\mathbf{P}_{aa} + \hat{\mathbf{a}}\hat{\mathbf{a}}^T\mathbf{P}_{b_0b_0} + \mathbf{P}_{b_0b_0}\mathbf{P}_{aa} + 2\,\mathbf{P}_{ab_0}\mathbf{P}_{b_0a}\Big)\Big]\end{aligned}$$

Po úpravě

$$\begin{aligned}\mathcal{E}\left\{\mathbf{B}^T\mathbf{G}(t+1)\mathbf{B}\right\} &= \hat{\mathbf{B}}^T\mathbf{G}(t+1)\hat{\mathbf{B}} + \\ &\mathbf{P}_{b_0b_0}\left(\hat{\mathbf{a}}^T\mathbf{G}\hat{\mathbf{a}} + \operatorname{tr}\left[\mathbf{G}\mathbf{P}_{aa}\right]\right) + 4\,\hat{b}_0\mathbf{P}_{b_0a}\mathbf{G}\hat{\mathbf{a}} + 2\,\mathbf{P}_{b_0a}\mathbf{G}\mathbf{P}_{ab_0} + \operatorname{tr}\left[\mathbf{G}\mathbf{P}_{bb}\right] + \\ &-2\,\mathbf{P}_{b_0b}\mathbf{G}\hat{\mathbf{a}} - 2\mathbf{P}_{b_0a}\mathbf{G}\hat{\mathbf{b}} - \hat{b}_0\,\operatorname{tr}\left[\mathbf{G}\left(\mathbf{P}_{ab} + \mathbf{P}_{ba}\right)\right] \\ &= \hat{\mathbf{B}}^T\mathbf{G}(t+1)\hat{\mathbf{B}} + \omega_2(t+1)\end{aligned}$$

kde pomocná posloupnost $\omega_2(t+1)$ je zřejmá z předchozího vztahu. Podobně vypočteme poslední výraz

$$\begin{aligned}\mathcal{E}\left\{\mathbf{x}^T(t)\mathbf{A}^T(t)\mathbf{G}(t+1)\mathbf{B}(t)\right\} &= \operatorname{tr}\left[\mathbf{G}(t+1)\;\mathcal{E}\left\{\mathbf{B}(t)\mathbf{x}^T(t)\mathbf{A}^T(t)\right\}\right] \\ \mathcal{E}\left\{\frac{1}{2}\boldsymbol{\gamma}^T(t+1)\mathbf{B}(t)\right\} &= \frac{1}{2}\boldsymbol{\gamma}^T(t+1)\left(\hat{\mathbf{B}} - \mathbf{P}_{ab_0}\right) \qquad (10.48)\end{aligned}$$

Opět upravujeme (s vynecháním všech argumentů)

$$\begin{aligned}\mathcal{E}\left\{\mathbf{B}\mathbf{x}^T\mathbf{A}^T\right\} &= \mathcal{E}\left\{(\mathbf{b} - b_0\mathbf{a})\left(\mathbf{w}^T - x_1\mathbf{a}^T\right)\right\} = \\ &= \mathcal{E}\left\{-x_1\mathbf{b}\mathbf{a}^T + \mathbf{b}\mathbf{w}^T - b_0\mathbf{a}\mathbf{w}^T + b_0x_1\mathbf{a}\mathbf{a}^T\right\} = \\ &= -\left(\hat{x}_1\hat{\mathbf{b}}\hat{\mathbf{a}}^T + \hat{x}_1\mathbf{P}_{ba} + \hat{\mathbf{b}}\mathbf{P}_{x_1a} + \mathbf{P}_{bx_1}\hat{\mathbf{a}}^T\right) + \left(\hat{\mathbf{b}}\hat{\mathbf{w}}^T + \mathbf{P}_{bw}\right) + \\ &\quad -\left(\hat{b}_0\hat{\mathbf{a}}\hat{\mathbf{w}}^T + \hat{b}_0\mathbf{P}_{aw} + \hat{\mathbf{a}}\mathbf{P}_{b_0w} + \mathbf{P}_{ab_0}\hat{\mathbf{w}}^T\right) + \\ &\quad \left(\hat{b}_0\hat{x}_1\hat{\mathbf{a}}\hat{\mathbf{a}}^T + \hat{b}_0\hat{\mathbf{a}}\mathbf{P}_{x_1a} + \hat{b}_0\mathbf{P}_{ax_1}\hat{\mathbf{a}}^T + \hat{x}_1\hat{\mathbf{a}}\mathbf{P}_{b_0a} + \hat{x}_1\mathbf{P}_{ab_0}\hat{\mathbf{a}}^T + \right. \\ &\quad \left.\hat{\mathbf{a}}\hat{\mathbf{a}}^T\mathbf{P}_{b_0x_1} + \hat{x}_1\hat{b}_0\mathbf{P}_{aa} + \mathbf{P}_{aa}\mathbf{P}_{b_0x_1} + \mathbf{P}_{ax_1}\mathbf{P}_{b_0a} + \mathbf{P}_{ab_0}\mathbf{P}_{x_1a}\right)\end{aligned}$$

kde střední hodnoty jednotlivých výrazů jsou pro snadnější orientaci v závorkách. Po dosazení a úpravě dostaneme

$$\begin{aligned}\mathcal{E}\left\{\mathbf{x}^T\mathbf{A}^T\mathbf{G}(t+1)\mathbf{B}\right\} &= \hat{\mathbf{x}}^T\hat{\mathbf{A}}^T\mathbf{G}\hat{\mathbf{B}} + \\ &-\hat{x_1}\,\operatorname{tr}\left[\mathbf{G}\mathbf{P}_{ba}\right] - \mathbf{P}_{x_1a}\mathbf{G}\hat{\mathbf{b}} - \hat{\mathbf{a}}^T\mathbf{G}\mathbf{P}_{bx_1} + \operatorname{tr}\left[\mathbf{G}\mathbf{P}_{bw}\right] - \hat{b_0}\,\operatorname{tr}\left[\mathbf{G}\mathbf{P}_{aw}\right] - \\ &\mathbf{P}_{b_0w}\mathbf{G}\hat{\mathbf{a}} - \hat{\mathbf{w}}^T\mathbf{G}\mathbf{P}_{ab_0} + 2\,\hat{b_0}\mathbf{P}_{x_1a}\mathbf{G}\hat{\mathbf{a}} + 2\,\hat{x_1}\mathbf{P}_{b_0a}\mathbf{G}\hat{\mathbf{a}} + \\ &\mathbf{P}_{b_0x_1}\hat{\mathbf{a}}^T\mathbf{G}\hat{\mathbf{a}} + \operatorname{tr}\left[\mathbf{G}\mathbf{P}_{aa}\right]\left(\hat{x_1}\hat{b}_0 + \mathbf{P}_{b_0x_1}\right) + 2\,\mathbf{P}_{b_0a}\mathbf{G}\mathbf{P}_{ax_1}\end{aligned}$$

Předchozí střední hodnotu vyjádříme ve tvaru lineární formy v $\hat{\mathbf{x}}$. Po úpravě dostaneme

$$\mathcal{E}\left\{\mathbf{x}^T\mathbf{A}^T\mathbf{G}(t+1)\mathbf{B}+\frac{1}{2}\boldsymbol{\gamma}^T\mathbf{B}\right\}=\hat{\mathbf{x}}^T\ \boldsymbol{\Omega}_3(t+1)\ +\ \omega_3(t+1) \tag{10.49}$$

kde

$$\begin{aligned}
\boldsymbol{\Omega}_3(t+1) &= \widehat{\mathbf{A}}^T\mathbf{G}\left(\widehat{\mathbf{B}}-\mathbf{P}_{ab_0}\right)+\mathbf{h}_x\left(-\mathrm{tr}\left[\mathbf{G}\mathbf{P}_{ba}\right]+\mathbf{P}_{b_0a}\mathbf{G}\hat{\mathbf{a}}+\hat{b}_0\mathrm{tr}\left[\mathbf{G}\mathbf{P}_{aa}\right]\right)\\
\omega_3(t+1) &= -\mathbf{P}_{x_1a}\mathbf{G}\hat{\mathbf{b}}-\hat{\mathbf{a}}^T\mathbf{G}\mathbf{P}_{bx_1}+\mathrm{tr}\left[\mathbf{G}\mathbf{P}_{bw}\right]-\hat{b_0}\ \mathrm{tr}\left[\mathbf{G}\mathbf{P}_{aw}\right]\\
&\quad -\mathbf{P}_{b_0w}\mathbf{G}\hat{\mathbf{a}}+2\ \hat{b_0}\mathbf{P}_{x_1a}\mathbf{G}\hat{\mathbf{a}}+\mathbf{P}_{b_0x_1}\hat{\mathbf{a}}^T\mathbf{G}\hat{\mathbf{a}}+\\
&\quad \mathrm{tr}\left[\mathbf{G}\mathbf{P}_{aa}\right]\mathbf{P}_{b_0x_1}+2\ \mathbf{P}_{b_0a}\mathbf{G}\mathbf{P}_{ax_1}+\frac{1}{2}\boldsymbol{\gamma}^T(t+1)\left(\widehat{\mathbf{B}}-\mathbf{P}_{ab_0}\right)
\end{aligned}$$

Po dosazení do (10.32) dostaneme výsledný vztah pro optimální řízení

$$\begin{aligned}
u^*(t) &= -\left[r+\widehat{\mathbf{B}}^T\mathbf{G}(t+1)\widehat{\mathbf{B}}+\omega_2(t+1)\right]^{-1}\times\\
&\quad\times\left(\boldsymbol{\Omega}_3^T(t+1)\ \hat{\mathbf{x}}(t|t-1)+\omega_3(t+1)\right)
\end{aligned}$$

Po dosazení do (10.33) dostaneme

$$\begin{aligned}
V_N(t) &= \hat{\mathbf{x}}^T(t|t-1)\mathbf{G}(t)\hat{\mathbf{x}}(t|t-1)+\boldsymbol{\gamma}^T(t)\hat{\mathbf{x}}(t|t-1)+g(t)\\
&\quad \sigma_e^2\left[\widehat{\mathbf{C}}\mathbf{G}(t+1)\widehat{\mathbf{C}}+\mathrm{tr}\left(\mathbf{G}(t+1)\mathbf{P}_{aa}\right)\right]+\boldsymbol{\gamma}^T(t+1)\left(\widehat{\mathbf{A}}\hat{\mathbf{x}}-\mathbf{P}_{ax_1}\right)+\\
&\quad g(t+1)+\hat{\mathbf{x}}^T\mathbf{Q}\mathbf{x}+\hat{\mathbf{x}}^T\boldsymbol{\Omega}_1(t+1)\hat{\mathbf{x}}+\boldsymbol{\Omega}_2^T(t+1)\hat{\mathbf{x}}+\omega_1(t+1)-\\
&\quad \left(\hat{\mathbf{x}}^T\ \boldsymbol{\Omega}_3(t+1)\ +\ \omega_3(t+1)\right)\left[r+\widehat{\mathbf{B}}^T\mathbf{G}(t+1)\widehat{\mathbf{B}}+\omega_2(t+1)\right]^{-1}\\
&\quad \left(\boldsymbol{\Omega}_3^T(t+1)\ \hat{\mathbf{x}}\ +\ \omega_3(t+1)\right)
\end{aligned}$$

Odtud

$$\begin{aligned}
\mathbf{G}(t) &= \mathbf{Q}+\widehat{\mathbf{A}}^T\mathbf{G}(t+1)\widehat{\mathbf{A}}+\mathrm{tr}(\mathbf{G}(t+1)\mathbf{P}_{aa})\mathbf{h}\mathbf{h}^T-\\
&\quad \boldsymbol{\Omega}_3(t+1)\ \left[r+\widehat{\mathbf{B}}^T\mathbf{G}(t+1)\widehat{\mathbf{B}}+\omega_2(t+1)\right]^{-1}\boldsymbol{\Omega}_3^T(t+1)\\
\boldsymbol{\gamma}(t) &= \left(\mathbf{h}\hat{\mathbf{a}}^T-\widehat{\mathbf{A}}^T\right)\mathbf{G}(t+1)\mathbf{P}_{ax_1}-\mathrm{tr}\left[\mathbf{G}(t+1)\left(\mathbf{P}_{wa}+\mathbf{P}_{aw}\right)\right]\mathbf{h}+\\
&\quad \widehat{\mathbf{A}}^T\boldsymbol{\gamma}(t+1)+2\ \boldsymbol{\Omega}_3(t+1)\ \left[r+\widehat{\mathbf{B}}^T\mathbf{G}(t+1)\widehat{\mathbf{B}}+\omega_2(t+1)\right]^{-1}\ \omega_3(t+1)\\
g(t) &= g(t+1)-2\mathbf{P}_{x_1w}\mathbf{G}(t+1)\hat{\mathbf{a}}+\\
&\quad \mathbf{P}_{x_1x_1}\left(\hat{\mathbf{a}}^T\mathbf{G}(t+1)\hat{\mathbf{a}}+\mathrm{tr}\left(\mathbf{G}(t+1)\mathbf{P}_{aa}\right)\right)+\\
&\quad 2\mathbf{P}_{x_1a}\mathbf{G}(t+1)\mathbf{P}_{ax_1}+\mathrm{tr}\left(\mathbf{G}(t+1)\mathbf{P}_{ww}\right)+\mathrm{tr}\left(\mathbf{Q}\mathbf{P}_{xx}\right)+\\
&\quad \sigma_e^2\left[\widehat{\mathbf{C}}\mathbf{G}(t+1)\widehat{\mathbf{C}}+\mathrm{tr}\left(\mathbf{G}(t+1)\mathbf{P}_{aa}\right)\right]-\boldsymbol{\gamma}^T(t+1)\mathbf{P}_{ax_1}+\\
&\quad \omega_3(t+1)\left[r+\widehat{\mathbf{B}}^T\mathbf{G}(t+1)\widehat{\mathbf{B}}+\omega_2(t+1)\right]^{-1}\omega_3(t+1)
\end{aligned}$$

kde okrajové podmínky jsou

$$\mathbf{G}(N)=\mathbf{S},\quad \boldsymbol{\gamma}(N)=0,\quad g(N)=\mathrm{tr}\left[\mathbf{S}\mathbf{P}(N|N-1)\right].$$

V předchozích výrazech jsme odvodili **opatrné strategie stochasticky optimálního řízení** ARMAX modelu, které operují pouze na měřitelných vstupních a výstupních datech. Tyto strategie využívají všechny neurčitosti odhadů. Jsou použitelné pouze pro konečnou dobu řízení, protože vlivem neurčitostí hodnota kritéria roste nade všechny meze s rostoucí dobou řízení.

Pokud v předchozích výrazech pro optimální řízení ignorujeme neurčitosti odhadu stavů a parametrů (zanedbáme všechny kovarianční matice stavů a parametrů), dostaneme **důvěřivé strategie stochasticky optimálního řízení** - strategie optimální podle tzv. **určitostního principu**. Ani tyto strategie nelze použít bez úpravy pro nekonečnou dobu řízení.

10.2 Stochasticky optimální řízení ARX modelu

Na rozdíl od obecnějšího ARMAX modelu má model ARX jednodušší vliv šumu na vývoj výstupu. Zvolíme neminimální realizaci ARX modelu, v níž je stav systému roven měřitelným posunutým hodnotám vstupu a výstupu. Proto v tomto modelu je třeba odhadovat pouze jeho parametry. Tím se liší následující odvození od předchozího případu.

10.2.1 ARX model

Vztahy mezi vstupem a výstupem jednorozměrového ARX modelu (autoregresního modelu s externím vstupem) jsou popsány diferenční rovnicí

$$y(t) = \sum_{i=1}^{n} a_i(t) y(t-i) + \sum_{j=0}^{n} b_j(t) u(t-j) + e(t) \tag{10.50}$$

kde $y(t)$ je výstup systému, $u(t)$ je jeho vstup, a_i, b_j jsou parametry systému a $e(t) = N(0, \sigma_e^2)$ je šum měření výstupu. Zavedeme si vektor parametrů $\boldsymbol{\theta}(t)$ a vektor posunutých vstupů a výstupů systému $\mathbf{z}(t)$ podle následujícího předpisu

$$\begin{aligned} \boldsymbol{\theta}(t) &= \left[\begin{array}{ccccccccc} b_0(t) & a_1(t) & b_1(t) & a_2(t) & b_2(t) & \dots & a_n(t) & b_n(t) \end{array} \right]^T \\ \mathbf{z}(t) &= \left[\begin{array}{ccccccc} u(t) & y(t-1) & u(t-1) & \dots & y(t-n) & u(t-n) \end{array} \right]^T \end{aligned}$$

Pomocí takto zavedených vektorů můžeme psát diferenční rovnici systému v následujícím jednoduchém tvaru

$$y(t) = \mathbf{z}^T(t) \boldsymbol{\theta}(t) + e(t) \tag{10.51}$$

10.2.2 Odhadování parametrů ARX modelu

Rovnice (10.51) je výstupní rovnicí, ve které vektor neznámých parametrů $\boldsymbol{\theta}(t)$ můžeme formálně považovat za stavový vektor systému a známý vektor dat $\mathbf{z}^T(t)$ můžeme považovat za výstupní matici systému.

Protože neznáme parametry systému (nyní vlastně jeho stavy), můžeme jejich vývoj modelovat formální stavovou rovnicí

$$\boldsymbol{\theta}(t+1) = \boldsymbol{\theta}(t) \tag{10.52}$$

Předpokládáme tedy sice neznámé, ale konstantní parametry systému a ještě navíc předpokládáme, že známe rozptyl σ_e^2 šumu $e(t)$.

Stavy (vlastně parametry) takového systému můžeme odhadovat Kalmanovým filtrem. Zavedeme obvyklé značení odhadu parametrů $\widehat{\boldsymbol{\theta}}(t,\tau)$, což je odhad (střední hodnota odhadu) parametrů v čase t podmíněný daty až do času τ. Stejným způsobem označíme i kovarianční matici odhadu jako $\mathbf{P}_\theta(t,\tau)$.

Kalmanův filtr se skládá z predikčního a filtračního kroku. Protože model vývoje stavu (parametrů) je zde tak jednoduchý, je predikční krok pouze formální $\widehat{\boldsymbol{\theta}}(t+1,t) = \widehat{\boldsymbol{\theta}}(t,t)$ a podobně pro kovarianci, není třeba v tomto případě u odhadů psát dva indexy. Platí tedy, že $\widehat{\boldsymbol{\theta}}(t)$ je odhad parametru (v čase t nebo $t+1$) podmíněný daty až do času t.

Pro odhad parametrů ARX modelu platí tedy jednoduché vztahy pro podmiňování

$$\begin{aligned}
\widehat{\boldsymbol{\theta}}(t) &= \widehat{\boldsymbol{\theta}}(t-1) + \mathbf{P}_{\theta y}(t-1)\mathbf{P}_y^{-1}(t-1)\left[y(t) - \hat{y}\right] \\
\hat{y}(t) &= \mathbf{z}^T(t)\widehat{\boldsymbol{\theta}}(t-1) \\
\mathbf{P}_\theta(t) &= \mathbf{P}_\theta(t-1) - \mathbf{P}_{\theta y}(t-1)\mathbf{P}_y^{-1}(t-1)\mathbf{P}_{y\theta}(t-1)
\end{aligned} \tag{10.53}$$

Vzájemná kovariance mezi stavy a výstupem je $\mathbf{P}_{\theta y}(t-1) = \mathbf{P}_\theta(t-1)\mathbf{z}(t)$ a rozptyl výstupu získáme ze stavové rovnice, pak $\mathbf{P}_y(t-1) = \mathbf{z}^T(t)\mathbf{P}_\theta(t-1)\mathbf{z}(t) + \sigma_e^2$.

Zavedeme si ještě normalizované kovariance $\mathbf{R}_\theta(t) = \dfrac{\mathbf{P}_\theta(t)}{\sigma_e^2}$. Potom po úpravě dostaneme výsledné vztahy pro střední hodnoty a normalizované kovariance odhadu parametrů ARX modelu

$$\begin{aligned}
\widehat{\boldsymbol{\theta}}(t) &= \widehat{\boldsymbol{\theta}}(t-1) + \frac{\mathbf{R}_\theta(t-1)\mathbf{z}(t)}{1 + \mathbf{z}^T(t)\mathbf{R}_\theta(t-1)\mathbf{z}(t)}\left[y(t) - \mathbf{z}^T(t)\widehat{\boldsymbol{\theta}}(t-1)\right] \\
\mathbf{R}_\theta(t) &= \mathbf{R}_\theta(t-1) - \frac{\mathbf{R}_\theta(t-1)\mathbf{z}(t)\mathbf{z}^T(t)\mathbf{R}_\theta(t-1)}{1 + \mathbf{z}^T(t)\mathbf{R}_\theta(t-1)\mathbf{z}(t)}
\end{aligned} \tag{10.54}$$

Odhad parametrů provádíme tedy sekvenčně na základě změřených dat (vstupů a výstupů systému). Výpočet startujeme z apriorních odhadů $\widehat{\boldsymbol{\theta}}(0) = \widehat{\boldsymbol{\theta}}_0$ a $\mathbf{R}_\theta(0)$, které vyjadřují naše počáteční nebo apriorní znalosti (znalosti, které nejsou založeny na datech).

10.2.3 Stavové rovnice ARX modelu

Pro účely řízení ARX modelu zavedeme stavové rovnice ARX modelu v poněkud jiném tvaru než byly stavové rovnice ARX modelu pro účely odhadování parametrů.

Výstupní rovnici (10.51) budeme psát ve tvaru

$$y(t) = \mathbf{C}(t)\mathbf{x}(t) + d(t)u(t) + e(t) \tag{10.55}$$

kde vektor stavů je roven starým hodnotám vstupu a výstupu ARX modelu

$$\mathbf{x}(t) = \begin{bmatrix} y(t-1) & u(t-1) & y(t-2) & u(t-2) & \dots & y(t-n) & u(t-n) \end{bmatrix}^T$$

Pro takto zavedený vektor stavů jsou výstupní matice $\mathbf{C}$ a d rovny

$$\begin{aligned}
\mathbf{C} &= \begin{bmatrix} a_1(t) & b_1(t) & a_2(t) & b_2(t) & \dots & a_n(t) & b_n(t) \end{bmatrix} \\
d &= b_0
\end{aligned}$$

Výstupní matice $\mathbf{C}$ a skalár d souvisejí s dříve zavedeným vektorem parametrů ARX modelu $\boldsymbol{\theta} = \begin{bmatrix} d & \mathbf{C} \end{bmatrix}^T$. Protože stavy jsou posunuté vstupy a výstupy systému, můžeme jejich vývoj vyjádřit stavovou rovnicí

$$\mathbf{x}(t+1) = \overline{\mathbf{A}}\mathbf{x}(t) + \overline{\mathbf{B}}u(t) + \mathbf{E}y(t) \tag{10.56}$$

kde matice $\overline{\mathbf{A}}$, $\overline{\mathbf{B}}$ a $\mathbf{E}$ jsou rovny

$$\overline{\mathbf{A}} = \begin{bmatrix} 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 & 0 \\ \dots & & \dots & & & \\ 0 & 0 & \dots & 1 & 0 & 0 \end{bmatrix}, \quad \overline{\mathbf{B}} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ \dots \\ 0 \end{bmatrix}, \quad \mathbf{E} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ \dots \\ 0 \end{bmatrix}$$

Protože stavová rovnice (10.56) obsahuje výstupní proměnnou $y(t)$, dosadíme (10.55) do stavové rovnice (10.56) a dostaneme stavovou rovnici v obvyklém tvaru

$$\mathbf{x}(t+1) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) + \mathbf{E}e(t) \tag{10.57}$$

kde matice $\mathbf{A}$ a $\mathbf{B}$ jsou rovny

$$\mathbf{A} = \overline{\mathbf{A}} + \mathbf{E}\mathbf{C} = \begin{bmatrix} a_1 & b_1 & \dots & b_{n-1} & a_n & b_n \\ 0 & 0 & \dots & 0 & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 & 0 \\ \dots & & \dots & & & \\ 0 & 0 & \dots & 1 & 0 & 0 \end{bmatrix}, \quad \mathbf{B} = \overline{\mathbf{B}} + \mathbf{E}d = \begin{bmatrix} b_0 \\ 1 \\ 0 \\ 0 \\ \dots \\ 0 \end{bmatrix}$$

Do prvních řádků matic $\mathbf{A}$ a $\mathbf{B}$ se okopírovaly parametry systému. Parametry systému jsou tedy pouze v prvních řádcích matic $\mathbf{A}$ a $\mathbf{B}$ a parametry systému tvoří také výstupní vektor $\mathbf{C}$ a skalár d.

10.2.4 Opatrné strategie ARX modelu

Nejprve uvedeme bez podrobnějšího odvození výsledky optimálního řízení stochastického systému za předpokladu, že neurčitost je pouze v parametrech systému. Stochastický systém budeme uvažovat ve tvaru

$$\begin{aligned} \mathbf{x}(t+1) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) + \mathbf{E}e(t) \\ y(t) &= \mathbf{C}\mathbf{x}(t) + du(t) + e(t) \end{aligned} \tag{10.58}$$

Předpokládáme, že šum $e(t)$ je stacionární bílá posloupnost s normálním rozdělením s nulovou střední hodnotou a rozptylem σ_e^2, čili $e(t) \sim N(\mathbf{0}, \sigma_e^2)$. Můžeme předpokládat, že počáteční podmínka $\mathbf{x}(0)$ je také náhodná proměnná, nezávislá na šumu $e(t)$, se střední hodnotou $\boldsymbol{\mu}_{x0}$ a kovariancí $\mathbf{P}_{x0}$, pak tedy $\mathbf{x}(0) \sim N(\boldsymbol{\mu}_{x0}, \mathbf{P}_{x0})$. Předpokládáme úplnou znalost stavu systému.

Budeme opět hledat optimální řízení, které minimalizuje kvadratické kritérium kvality řízení ve tvaru

$$J = \mathcal{E}\left\{ y(N)s_y y(N) + \sum_{t=0}^{N-1} y(t)q_y y(t) + r_u u^2(t) \right\} \tag{10.59}$$

V kritériu kvality řízení nyní vážíme výstup systému (a ne jeho stav).

Poznámka: V kritériu kvality řízení se obvykle v koncovém čase $t = N$ váží stav a ne pouze výstup systému. Tím se pro $N \to \infty$ zaručí stabilita celého zpětnovazebního obvodu. Protože stavy neminimální realizace ARX modelu jsou pouze posunuté vstupy a výstupy, tak vážením koncového stavu v kritériu bychom v tomto případě nic nezískali. □

Pro řešení této úlohy dynamickým programováním si zavedeme optimální funkci $V(\mathbf{x}(t), t)$, která je opět rovna optimální hodnotě kritéria kvality řízení při obecném počátečním stavu $\mathbf{x}(t)$ a počátečním čase t. Předpokládáme že má tvar

$$V(\mathbf{x}(t), t) = \mathbf{x}^T(t)\mathbf{G}(t)\mathbf{x}(t) + g(t) \tag{10.60}$$

Po dosazení do Bellmanovy rovnice a minimalizaci dostaneme následující výsledky:

- Nejprve zavedeme nové váhové matice, které vzniknou při dosazení do kritéria za výstup $y(k)$ ze stavové rovnice.

$$\begin{aligned} \mathbf{S} &= \mathbf{C}^T s_y \mathbf{C}, & \mathbf{Q} &= \mathbf{C}^T q_y \mathbf{C}, \\ s_u &= s_y d^2, & \mathbf{Q}_u &= \mathbf{C}^T q_y d, \\ r &= r_u + q_y d^2 \end{aligned}$$

- Optimální řízení v čase $t = N$ je rovno

$$u^*(N) = -\mathcal{E}\left\{s_u\right\}^{-1} \mathcal{E}\left\{d s_y \mathbf{C}\right\} \mathbf{x}(N) \tag{10.61}$$

  Optimální funkce v čase $t = N$ je rovna

$$V(\mathbf{x}(N), N) = s_y \sigma_e^2$$

- Optimální řízení v čase $t \in [0, N-1]$ je rovno

$$\begin{aligned} u^*(t) &= -\left[\mathcal{E}\left\{r\right\} + \mathcal{E}\left\{\mathbf{B}^T\mathbf{G}(t+1)\mathbf{B}\right\}\right]^{-1} \times \\ &\quad \times \left[\mathcal{E}\left\{\mathbf{Q}_u^T\right\} + \mathcal{E}\left\{\mathbf{B}^T\mathbf{G}(t+1)\mathbf{A}\right\}\right] \mathbf{x}(t) \end{aligned} \tag{10.62}$$

  Dosazením optimálního řízení do optimální funkce dostaneme

$$\begin{aligned} V(\mathbf{x}(t), t) &= \mathbf{x}^T(t)\mathbf{G}(t)\mathbf{x}(t) + g(t) = q_y\sigma_e^2 + \mathcal{E}\left\{\mathbf{E}^T\mathbf{G}(t+1)\mathbf{E}\right\}\sigma_e^2 + \\ & g(t+1) + \mathbf{x}^T(t)\left[\mathcal{E}\left\{\mathbf{Q}\right\} + \mathcal{E}\left\{\mathbf{A}^T\mathbf{G}(t+1)\mathbf{A}\right\}\right]\mathbf{x}(t) - \\ & \mathbf{x}^T(t)\left[\mathcal{E}\left\{\mathbf{Q}_u\right\} + \mathcal{E}\left\{\mathbf{A}^T\mathbf{G}(t+1)\mathbf{B}\right\}\right]\left[\mathcal{E}\left\{\mathbf{r}\right\} + \mathcal{E}\left\{\mathbf{B}^T\mathbf{G}(t+1)\mathbf{B}\right\}\right]^{-1} \times \\ & \times \left[\mathcal{E}\left\{\mathbf{Q}_u^T\right\} + \mathcal{E}\left\{\mathbf{B}^T\mathbf{G}(t+1)\mathbf{A}\right\}\right]\mathbf{x}(t) \end{aligned}$$

Porovnáním dostaneme rekurentní vztahy pro výpočet maticové posloupnosti $\mathbf{G}(t)$ a posloupnosti $g(t)$

$$\begin{aligned}\mathbf{G}(t) &= \mathcal{E}\left\{\mathbf{Q}\right\} + \mathcal{E}\left\{\mathbf{A}^T\mathbf{G}(t+1)\mathbf{A}\right\} - \\ &\quad \left[\mathcal{E}\left\{\mathbf{Q}_u\right\} + \mathcal{E}\left\{\mathbf{A}^T\mathbf{G}(t+1)\mathbf{B}\right\}\right]\left[\mathcal{E}\left\{\mathbf{r}\right\} + \mathcal{E}\left\{\mathbf{B}^T\mathbf{G}(t+1)\mathbf{B}\right\}\right]^{-1} \times \\ &\quad \times \left[\mathcal{E}\left\{\mathbf{Q}_u^T\right\} + \mathcal{E}\left\{\mathbf{B}^T\mathbf{G}(t+1)\mathbf{A}\right\}\right] \\ g(t) &= q_y\sigma_e^2 + \mathcal{E}\left\{\mathbf{E}^T\mathbf{G}(t+1)\mathbf{E}\right\}\sigma_e^2 + g(t+1) \qquad (10.63)\end{aligned}$$

Koncové podmínky rekurze jsou

$$\mathbf{G}(N) = \mathbf{0}, \qquad g(N) = s_y\sigma_e^2 \qquad (10.64)$$

Poznámka: Pokud vážíme v kritériu (10.59) v koncovém čase $t = N$ místo výstupu koncový stav, pak kritérium (10.59) na řízení $\mathbf{u}(N)$ vůbec nezávisí. Vztahy (10.63) pro matici $\mathbf{G}(t)$ a posloupnost $g(t)$ zůstávají v platnosti, pouze koncové podmínky jsou v tomto případě $\mathbf{G}(N) = \mathbf{S}$ a $g(N) = 0$.

Odvodili jsme rekurzivní vztahy pro výpočet optimálního řízení při respektování neurčitosti v parametrech systému. Tyto neurčitosti se projeví při výpočtu středních hodnot v předchozích výrazech.

Stav $\mathbf{x}(t)$ ve stavových rovnicích ARX modelu je roven zpožděným hodnotám vstupu a výstupu modelu a proto je stav měřitelný a není třeba jej odhadovat. To je ale možné pouze u neminimální realizace ARX modelu popsané v předchozím odstavci. Vektor parametrů $\boldsymbol{\theta}(t)$ je podle námi zavedených matic roven

$$\boldsymbol{\theta}(t) = \begin{bmatrix} d \\ \mathbf{C}^T \end{bmatrix} = \begin{bmatrix} b_0 \\ \mathbf{C}^T \end{bmatrix}$$

Pokud parametry systému neznáme, můžeme je odhadovat podle předchozího postupu a potom známe pouze jejich střední hodnotu a kovarianční matici. Odhad vektoru parametrů je náhodná proměnná s normálním rozdělením (pokud šum $e(t)$ je normální), čili

$$\widehat{\boldsymbol{\theta}}(t) = \begin{bmatrix} \hat{b}_0(t) \\ \widehat{\mathbf{C}}^T(t) \end{bmatrix}, \qquad \mathbf{P}_\theta = \begin{bmatrix} \sigma_{b_0}^2 & \mathbf{P}_{Cb_0} \\ \mathbf{P}_{b_0C} & \mathbf{P}_C \end{bmatrix} \qquad (10.65)$$

Abychom mohli použít odvozené výsledky pro opatrné strategie, je třeba vypočítat střední hodnoty výrazů, které se vyskytují ve vztazích pro opatrné strategie optimálního řízení. Podle předchozího rozdělení náhodného vektoru $\theta(t)$ dostaneme

$$\begin{aligned}\mathbf{Q} &= \mathbf{C}^T q_y \mathbf{C} \quad \text{a proto} \quad \mathcal{E}\left\{\mathbf{Q}\right\} = q_y\mathcal{E}\left\{\mathbf{C}^T\mathbf{C}\right\} = q_y\left(\widehat{\mathbf{C}}^T\widehat{\mathbf{C}} + \mathbf{P}_C\right) \\ \mathbf{Q}_u &= \mathbf{C}^T q_y d \quad \text{a proto} \quad \mathcal{E}\left\{\mathbf{Q}_u\right\} = q_y\mathcal{E}\left\{\mathbf{C}^T d\right\} = q_y\left(\widehat{\mathbf{C}}^T\hat{b_0} + \mathbf{P}_{b_0C}\right) \\ r &= r_u + q_y d^2 \quad \text{a proto} \quad \mathcal{E}\left\{r\right\} = r_u + q_y\left(\hat{b_0}^2 + \sigma_{b_0}^2\right) \\ \mathbf{S} &= \mathbf{C}^T s_y \mathbf{C} \quad \text{a proto} \quad \mathcal{E}\left\{\mathbf{S}\right\} = s_y\mathcal{E}\left\{\mathbf{C}^T\mathbf{C}\right\} = s_y\left(\widehat{\mathbf{C}}^T\widehat{\mathbf{C}} + \mathbf{P}_C\right) \\ s_u &= s_y d^2 \quad \text{a proto} \quad \mathcal{E}\left\{s_u\right\} = s_y\mathcal{E}\left\{d^2\right\} = s_y\left(\hat{b_0}^2 + \sigma_{b_0}^2\right)\end{aligned}$$

Nyní vypočteme střední hodnoty součinů matic $\mathbf{A}^T\mathbf{G}(t+1)\mathbf{A}$, $\mathbf{A}^T\mathbf{G}(t+1)\mathbf{B}$, $\mathbf{B}^T\mathbf{G}(t+1)\mathbf{B}$ a $\mathbf{B}^T\mathbf{G}(t+1)\mathbf{A}$. Uvědomme si, že neznámé parametry jsou pouze v prvním řádku matice $\mathbf{A}$, (kde je vlastně řádkový vektor $\mathbf{C}$) a v prvním prvku vektoru řízení $\mathbf{B}$, (kde je prvek b_0). Proto platí

$$\begin{aligned}\mathcal{E}\left\{\mathbf{B}^T\mathbf{G}(t+1)\mathbf{B}\right\} &= \mathcal{E}\left\{\left(b_0G_{11}+G_{21}\right)b_0+b_0G_{12}+G_{22}\right\}\\ &= \mathcal{E}\left\{\mathbf{B}^T\right\}\mathbf{G}(t+1)\mathcal{E}\left\{\mathbf{B}\right\}+G_{11}\sigma_{b_0}^2\\ &= G_{11}\left(\hat{b}_0^2+\sigma_{b_0}^2\right)+2\hat{b}_0G_{12}+G_{22}\end{aligned} \tag{10.66}$$

kde G_{11}, G_{12} a G_{22} jsou odpovídající prvky matice $\mathbf{G}(t+1)$. Podobně

$$\mathcal{E}\left\{\mathbf{A}^T\mathbf{G}(t+1)\mathbf{B}\right\}=\mathcal{E}\left\{\mathbf{A}^T\right\}\mathbf{G}(t+1)\mathcal{E}\left\{\mathbf{B}\right\}+G_{11}\mathbf{P}_{b_0C} \tag{10.67}$$

Matice $\mathcal{E}\left\{\mathbf{B}^T\mathbf{G}(t+1)\mathbf{A}\right\}$ je pouze transpozice předchozího výrazu. S ohledem na to, že neurčitosti jsou pouze v prvním řádku matice $\mathbf{A}$, snadno spočteme i $\mathcal{E}\left\{\mathbf{A}^T\mathbf{G}(t+1)\mathbf{A}\right\}$. Po jednoduchých úpravách dostaneme

$$\mathcal{E}\left\{\mathbf{A}^T\mathbf{G}(t+1)\mathbf{A}\right\}=\mathcal{E}\left\{\mathbf{A}^T\right\}\mathbf{G}(t+1)\mathcal{E}\left\{\mathbf{A}\right\}+G_{11}\mathbf{P}_C \tag{10.68}$$

Na závěr shrneme **výsledky kvadraticky optimálního řízení ARX modelu při opatrných i důvěřivých strategiích**.

- Optimální řízení je podle (10.61) a (10.62) rovno

$$\begin{aligned}\mathbf{u}^*(N) &= -\mathcal{E}\left\{s_u\right\}^{-1}\mathcal{E}\left\{ds_y\mathbf{C}\right\}\mathbf{x}(N)=-\frac{1}{\hat{b}_0^2+\sigma_{b_0}^2}\left(\hat{\mathbf{C}}\hat{b_0}+\mathbf{P}_{Cb_0}\right)\mathbf{x}(N)\\ \mathbf{u}^*(t) &= -\left[\mathcal{E}\left\{r\right\}+\mathcal{E}\left\{\mathbf{B}^T\mathbf{G}(t+1)\mathbf{B}\right\}\right]^{-1}\times\\ &\quad\times\left[\mathcal{E}\left\{\mathbf{Q}_u^T\right\}+\mathcal{E}\left\{\mathbf{B}^T\mathbf{G}(t+1)\mathbf{A}\right\}\right]\mathbf{x}(t)=\\ &= -\frac{q_y\left(\hat{\mathbf{C}}\hat{b_0}+\mathbf{P}_{Cb_0}\right)+\mathcal{E}\left\{\mathbf{B}^T\right\}\mathbf{G}(t+1)\mathcal{E}\left\{\mathbf{A}\right\}+\mathbf{P}_{Cb_0}G_{11}(t+1)}{r_u+q_y\left(\hat{b_0}^2+\sigma_{b_0}^2\right)+G_{11}\left(\hat{b}_0^2+\sigma_{b_0}^2\right)+2\hat{b}_0G_{12}+G_{22}}\mathbf{x}(t)\end{aligned}$$

  kde $t=0,1,\ldots,N-1$.

- Matice $\mathbf{G}(t)$ je určena rekurentním předpisem (10.63a). Po dosazení středních hodnot dostaneme

$$\begin{aligned}\mathbf{G}(t) &= q_y\left(\hat{\mathbf{C}}^T\hat{\mathbf{C}}+\mathbf{P}_C\right)+\mathcal{E}\left\{\mathbf{A}^T\right\}\mathbf{G}(t+1)\mathcal{E}\left\{\mathbf{A}\right\}+G_{11}\mathbf{P}_C+\\ &\quad+\frac{\left[q_y\left(\hat{\mathbf{C}}^T\hat{b_0}+\mathbf{P}_{b_0C}\right)+\mathcal{E}\left\{\mathbf{A}^T\right\}\mathbf{G}(t+1)\mathcal{E}\left\{\mathbf{B}\right\}+\mathbf{P}_{b_0C}G_{11}(t+1)\right]}{r_u+q_y\left(\hat{b_0}^2+\sigma_{b_0}^2\right)+G_{11}\left(\hat{b}_0^2+\sigma_{b_0}^2\right)+2\hat{b}_0G_{12}+G_{22}}\times\\ &\quad\times\left[Q_y\left(\hat{\mathbf{C}}^T\hat{b_0}+\mathbf{P}_{b_0C}\right)+\mathcal{E}\left\{\mathbf{A}^T\right\}\mathbf{G}(t+1)\mathcal{E}\left\{\mathbf{B}\right\}+\mathbf{P}_{b_0C}G_{11}(t+1)\right]^T\end{aligned}$$

  Koncová podmínka je rovna $\mathbf{G}(N)=\mathbf{0}$.

- Posloupnost $g(t)$ je určena rekurentním předpisem (10.63b), který po dosazení je tvaru
$$g(t) = g(t+1) + \sigma_e^2 \left(1 + G_{11}(t+1)\right) \tag{10.69}$$
s koncovou podmínkou $g(N) = s_y \sigma_e^2$.

- Optimální hodnota kritéria kvality řízení je
$$J^* = J_N^*(\mathbf{x}(0), 0) = \mathbf{x}^T(0)\mathbf{G}(0)\mathbf{x}(0) + g(0) \tag{10.70}$$

- Důvěřivé strategie dostaneme při nulových rozptylech i kovariančních maticích parametrů.

10.3 Příklad

V jazyce MATLAB byl sestaven program pro výpočet optimálního řízení stochastického systému pro opatrné i důvěřivé strategie.

Zde uvedeme řešení jednoho jednoduchého problému, ze kterého je zřejmé, jak nejistota ve stavech a parametrech ovlivňuje optimální strategii řízení.

ARMAX model uvažujeme ve tvaru (10.1) a kvadratické kritérium ve tvaru (10.26). ARMAX model získáme tak, že nejprve uvažujeme spojitý nestabilní systém s přenosem

$$S(s) = \frac{1}{(10s-1)(10s+1)^2},$$

který diskretizujeme s periodou vzorkování $T_s = 5$. Diskrétní model má potom přenos

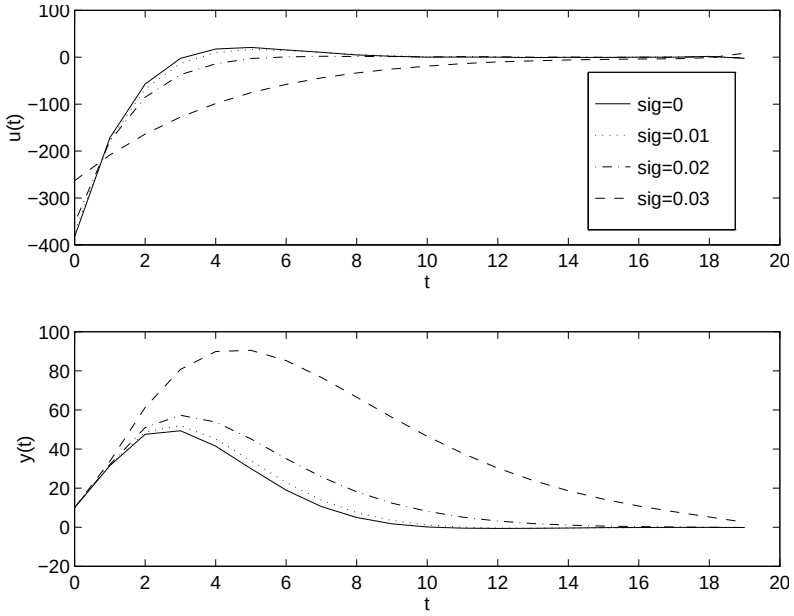
$$S(d) = \frac{b(d)}{a(d)} = \frac{-0.0187d - 0.0672d^2 - 0.0146d^3}{1 - 2.8618d + 2.3679d^2 - 0.6065d^3}$$

Parametry diskretizovaného modelu byly považovány za střední hodnoty parametrů $\widehat{\boldsymbol{\theta}}$, které byly určeny identifikací ze změřených dat. Neurčitost parametrů a stavů systému je reprezentována jejich kovarianční maticí $\mathbf{P}$ - viz (10.42), která byla pro jednoduchost zvolena jako diagonální matice s prvky σ_b^2, σ_a^2 a σ_x^2, což jsou po řadě rozptyly všech parametrů b_i a a_i v čitateli a jmenovateli přenosu a rozptyly stavů $x_i(t)$.

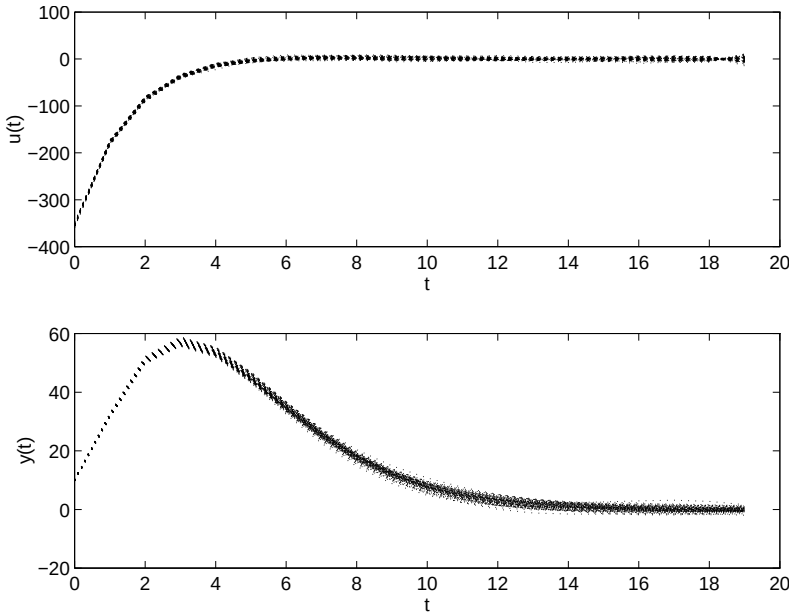
Počet kroků řízení je $N = 20$, počáteční stav byl zvolen $\mathbf{x}_0^T = [10,\ 10,\ 10]$. Stavový model diskrétního systému je ve tvaru (10.3). Váhové matice v kritériu (10.26) byly zvoleny $\mathbf{Q} = \mathbf{I}_n$ (váha ve stavech), $\mathbf{S} = 1000\ \mathbf{I}_n$ (váha koncového stavu) a $r = 1$ (váha řízení).

Z mnoha simulačních běhů ukážeme zde pouze vliv neurčitosti v parametrech systému na optimální řízení a optimální přechodový jev. V obr. 10.2 je pro různé hodnoty rozptylu parametrů $\sigma_a = \sigma_b = \sigma$ zaznamenán průběh optimální řídicí veličiny $u(t)$ a průběh výstupu systému $y(t)$. Z obr. 10.2 je zřejmé, že s rostoucí nejistotou v parametrech je řízení opatrnější (akční zásahy jsou menší) a přechodový jev je proto pomalejší. Při tomto experimentu byl šum $e(t)$ ARMAX modelu modelován jako bílý normálně rozdělený šum s nulovou střední hodnotou a směrodatnou odchylkou $\sigma_e = 0.01$.

Optimální trajektorie výstupu závisejí na realizaci šumu $e(t)$. Proto v obr. 10.2 jsou pouze realizace řízení a výstupu systému. V obr. 10.3 je zaznamenán průběh jednoho



Obrázek 10.2: Průběh optimálního řízení $u(t)$ a optimálního výstupu $y(t)$ při různých nejistotách v parametrech systému (v obr. sig $= \sigma$)

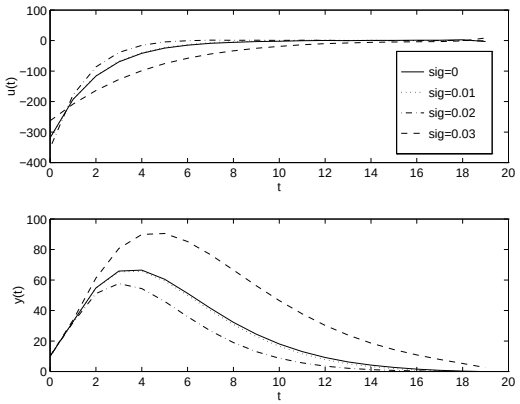


Obrázek 10.3: 100 realizací optimálního řízení a optimálního výstupu řízeného systému

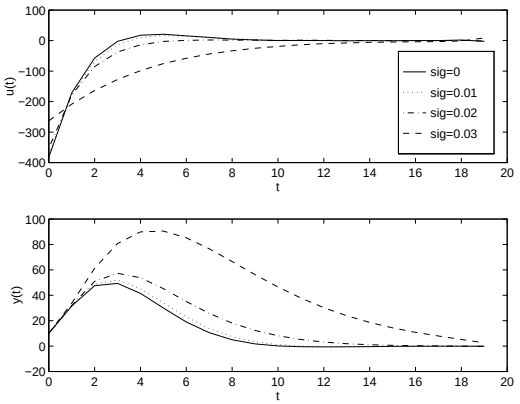
sta simulací optimálního řízení a optimálního výstupu při různých realizacích šumu $e(t)$, $t = 1, 2, \ldots, 20$. Při tom počáteční stav byla také náhodná veličina $\mathbf{x}(0) \sim \mathcal{N}(\widehat{\mathbf{x}}(0), \sigma_{x_0})$. Při tom nejistota v parametrech byla $\sigma_a = \sigma_b = 0.1$, nejistota ve stavech $\sigma_x = 0.1$. Šum modelu byl $\sigma_e = 0.1$ a směrodatná odchylka počátečního stavu byla $\sigma_{x_0} = 0.5$. Váhové matice v kritériu byly stejné jako v předchozí simulaci.

V obr. 10.4 je ukázán vliv váhových matic v kritériu na optimální řízení a optimální výstup systému při různých nejistotách v parametrech systému.

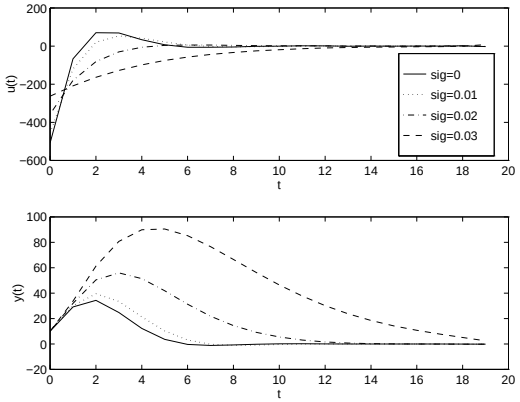
Výsledky simulací názorně ukazují ten zřejmý fakt, že pokud se rozhodujeme v podmín-



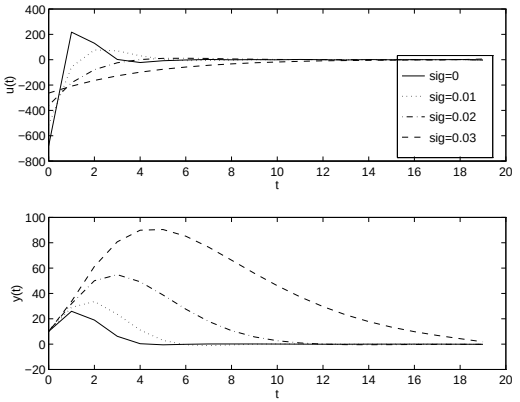
$Q = 0.01 \cdot I_n,\ S = 1000 \cdot I_n,\ r = 1;$



$Q = I_n,\ S = 1000 \cdot I_n,\ r = 1;$



$Q = 10 \cdot I_n,\ S = 1000 \cdot I_n,\ r = 1;$



$Q = 100 \cdot I_n,\ S = 1000 \cdot I_n,\ r = 1;$

Obrázek 10.4: Optimální řízení a optimální výstup systému v závislosti na váhových maticích v kritériu při různých nejistotách v parametrech systému

kách nejistoty, pak naše rozhodování je opatrnější, musíme se prostě zajistit proti nejhoršímu možnému případu. To odpovídá reálnému chování lidí. Výsledky takového chování nemusejí být vždy nejlepší. Často jsou úspěšnější ti, kteří více riskují. Nejistota nemusí vždy vést k opatrnosti. Dokonce výsledky simulací v některých případech tento neobvyklý jev potvrzují.

Literatura

[1] Ansari N., Hou E. : Computational Intelligence for Optimization. Kluwer, Boston, 1997.

[2] Aström K., J. : Introduction to Stochastic Control Theory. Academic Press, New York, 1970.

[3] Athans M., Falb P. L. : Optimal Control. An Introduction to the Theory and Its Application. McGraw-Hill, New York, 1966.

[4] Bellman R. : Dynamic Programming. Princeton University Press, Princeton, 1957.

[5] Bertsekas D. P. : Dynamic Programming and Optimal Control. Vol I and II, Athena Scientific, Belmont, MA, 1995.

[6] Bertsekas D. P. : Nonlinear Programming. Athena Scientific, Belmont, MA, 1995.

[7] Björck Å. : Numerical Methods for Least Squares Problems, Siam, Philadelphia, 1996.

[8] Boltjanskij V. G. : Matematičeskije metody optimalnovo upravlenija. Nauka, Moskva, 1969.

[9] Brunovský P. : Matematická teória optimálného riadenia. Alfa, Bratislava; SNTL, Praha, 1980.

[10] Bryson A. E., Yu Chi Ho : Applied Optimal Control. Massachusetts, Waltham, 1969.

[11] Canon M. D., Cullum C. D., Polak E. : Theory of Optimal Control and Mathematical Programmimg. McGraw-Hill, New York, 1970.

[12] Clarke F. H. : Nonsmooth Analysis and Optimization. Wiley Interscience, N. Y., 1983.

[13] Dantzing G. B. : Linear Programming and Extensions. Pronceton University Press, Princeton, 1963.

[14] Feldbaum A. A. : Osnovy teorii optimalnych avtomatičeskich sistem. Moskva, 1963.

[15] Gabasov R., Kirillova F. M. : Osnovy dinamičeskovo programovania. Izd. BGU, Minsk, 1975.

[16] Hamala M. : Nelineárné programovanie. Alfa, Bratislava, 1970.

[17] Hillier F. S., Lieberman G. J. : Introduction to Operations Research. McGraw-Hill, Inc., New York, 1995.

[18] Himmelblau D. M. : Applied Nonlinear Programming. McGraw-Hill, New York, 1972.

[19] Isaacs R. : Differencial Games. John Wiley, New York, 1965.

[20] Kwakernaak H., Sivan R. : Linear Optimal Control Systems. John Wiley, New York, 1972.

[21] Lavrjentjev M. A., Ljusternik L. A. : Kurs variačního počtu. Přírodovědecké nakladatelství, Praha, 1952.

[22] Larson R. E. : State Increment Dynamic Programming. Elsevier, New York, 1967.

[23] Luenberger D. G. : Optimization by Vector Space Methods. John Wiley, New York, 1969.

[24] Luenberger D. G. : Introduction to Linear and Nonlinear Programming. Reading, Massachusetts, Addison-Wesley, 1973.

[25] Luenberger D. G. : Linear and Nonlinear Programming. Reading, Massachusetts, Addison-Wesley, 1984.

[26] Maňas M. : Teorie her a optimální rozhodování. SNTL, Praha, 1974.

[27] Maňas M. : Optimalizační metody. SNTL, Praha, 1979.

[28] Polak E. : Computational Methods in Optimization. A Unified Approach. Academic Press, Nwe York, 1971.

[29] Pontrjagin L. S., Boltjanskij V., Gamkrelidze R, Misčenko E. : Matěmatičeskaja těorija optimalnych processov. Fizmatgiz, Moskva 1961. (český překlad SNTL, Praha, 1964)

[30] Propoj A. I. : Elementy teorii optimalnych diskretnych processov. Nauk, Moskva, 1973.

[31] Rockafellar R. T. : Convex Analysis. Princeton Univ. Press, Prinseton, N. Y., 1970.

[32] Vidyasagar M. : A Theory of Learning and Generalization: With Applications to Neural Networks and Control Systems. Springer-Verlag, London, 1997.