

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE



**FAKULTA CHEMICKEJ A POTRAVINÁRSKEJ TECHNOLOGIE
ÚSTAV INFORMATIZÁCIE, AUTOMATIZÁCIE A MATEMATIKY
ODDELENIE INFORMATIZÁCIE A RIADENIA PROCESOV**



**KONVEXNÁ OPTIMALIZÁCIA A PREDIKTÍVNE RIADENIE S
MODELOM
BAKALÁRSKA PRÁCA**

Vypracoval: Branislav Mitterpach

Vedúci práce: Ing. Michal Kvasnica, PhD.

Bratislava 2009

Zadanie

Čestné prehlásenie:

Čestne prehlasujem že bakalársku prácu som vypracoval samostatne, na základe zdrojov uvedených v literatúre, pod vedením vedúceho bakalárskej práce a s pomocou vedomostí nadobudnutých počas štúdia.

.....

Branislav Mitterpach

PodĎakovanie

Chcem sa touto cestou nesmierne poďakovať Ing. Michalovi Kvasnicovi, PhD. ktorý sa ukázal ako vysokokvalifikovaný odborník a skvelý pedagóg, za poskytovanie rád a podnetov ktoré mi dodával počas celého vypracovávania bakalárskej práce, za trpezlivosť a zato že odolal pokušeniu fyzicky ma inzultovať pri mojich častých poznámkach.

Abstrakt

Zameraním bakalárskej práce je rozobrať tematiku konvexnej optimalizácie, jednoduchosť jej použitia a riešenia v bežných činnostiach, význam obmedzení pre tvorbu a riešenie konvexného optimalizačného problému, práca pojednáva o jednoduchosti lineárnych ohraničení ktoré ilustruje príkladmi kódu v Matlabe. Práca ďalej rozoberá problematiku, vlastnosti a základné pojmy s prediktívneho riadenia, naráža pri tom na tému noriem a poukazuje na význam konvexnej optimalizácie ako neoddeliteľnej súčasti efektívneho prediktívneho riadenia.

Kľúčové slová : Konvexná optimalizácia, prediktívne riadenie, obmedzenia

Abstract

Aim of this bachelor work analyze topic of convex optimization , simplicity of use and solutions in normal activities, importance of constraints for creating and solution convex optimization problem, work deals about simplicity linear constraints which illustrating with examples of source code in Matlab. Work further analyze questions, properties and generally terms from predictive model control, along with theme of norms showing importance of convex optimization as integral part of effective predictive model control.

Keywords : Convex optimization, model predictive control, constraints

Obsah

Úvod.....	8
1. Konvexná Optimalizácia.....	9
1.1. Optimalizácia	9
1.1.1. Rozdelenie optimalizačných metód[3]	9
1.1.1.1. Lokálna optimalizácia	10
1.1.1.2. Globálna optimalizácia.....	10
1.1.2. Rozdelenie optimalizačných problémov[3]	11
1.2. Konvexná množina.....	12
1.2.1. Spojitý priestor	12
1.2.2. Konvexná oblasť.....	13
1.3. Konvexný optimalizačný problém	14
1.4. Lineárny konvexný optimalizačný problém	15
1.4.1. Praktické aspekty	15
1.4.2. Formulácia problému.....	16
1.4.3. Vznik lineárnej konvexnej množiny	17
1.4.3.1. Polorovina a polpriestor	17
1.4.3.2. Druhy vznikov	19
1.5. Transformácie lineárneho konvexného optimalizačného problému	22
1.5.1. Transformácia štandardných foriem.....	22
1.5.2. Transformácia s pomocnými premennými	25
1.5.3. Podpriestorová transformácia.....	28
1.6. Nutričný problém	31
2. Prediktívne riadenie.....	35
2.1. Význam prediktívneho riadenia	35
2.1.1. Vývoj prediktívneho riadenia.....	35
2.1.2. Základné princípy	35
2.1.3. Analógie prediktívneho riadenia	36
2.2. Vlastnosti prediktívneho riadenia.....	37
2.2.1. Horizont predikcie a model systému	37
2.2.2. Obmedzenia.....	39
2.3. Konvexná optimalizácia a prediktívne riadenie.....	41
Záver.....	45
Použitá Literatúra.....	46

Úvod

Od úsvitu ľudstva sa musel každý človek každú sekundu svojho života rozhodovať čo bude robiť a hlavne ako to bude robiť. Čím sám podvedome optimalizoval svoju činnosť vzhľadom na čas a dostupné zdroje. Optimalizácia prešla časom vývojom k dnešnej forme v ktorej má nespočetné množstvo vetiev výskumu. Jednou s týchto vetiev je aj konvexná optimalizácia ktorá sa zaoberá hľadaním optima v dopredu definovanej oblasti ktorá musí spĺňať podmienky konvexnej množiny. Konvexná optimalizácia prešla za posledné roky veľkým vývojom a rozvojom metód, pre svoju jednoduchosť sa dá uplatniť v podstate každom bežnom alebo technickom probléme, najväčšie uplatnenie našla práve v priemyselnej oblasti kde sa stala neoddeliteľnou súčasťou mnohých procesov a algoritmov riadenia ako napríklad prediktívneho riadenia. Prediktívne riadenia sa prudko rozvinulo najmä v posledných desaťročiach kedy sa začalo využívať pre svoju efektívnosť a priaznivé účinky na kvalitu produkcie. Vzájomné prepojenie týchto dvoch oblastí sa ukazuje už dnes ako veľmi zaujímavé s vysokým predpokladom budúceho rastu lukratívnosti tohto odvetvia.

Práca sa snaží čo najjednoduchšie čitateľovi načrtnúť a vysvetliť základné pojmy s konvexnej optimalizácie s vysvetlením pojmov konvexnej oblasti, druhov ohraničení základných formulácií problému, poukazuje na jednoduchosť a eleganciu využitia tejto metódy pre riešenie problémov. V časti prediktívneho riadenia sa čitateľ jednoduchou formou zoznámí s hlavnými myšlienkami a princípmi prediktívneho riadenia, jeho rozdielom oproti bežným formám riadenia, vplyvom obmedzení a horizontu predikcie na riadenie a hlavne na významný súvis konvexnej optimalizácie a prediktívneho riadenia.

1. Konvexná Optimalizácia

1.1. Optimalizácia

Od počiatkov ľudskej civilizácie musel človek pri vykonávaní akejkoľvek činnosti uvažovať ako ju vykonať čo najefektívnejšie s najvhodnejším použitím dostupných prostriedkov a s dosiahnutím maximálnej úspešnosti cieľa činnosti. Vykonávanie takejto činnosti sa nazýva optimalizácia, je zmesou matematických poznatkov a experimentálnych poznatkov [3]. Vykonáva sa prakticky pri všetkých oblastiach života a jej použitie a význam neustále narastá, pričom je jej predmetom najmä šetrenie základných nerastných, pracovných, ekonomických a časových zdrojov teda šetrenie času alebo zrýchlenie činnosti. Pod všeobecným pojmom optimalizácie si môžeme predstaviť hľadanie najvhodnejšieho riešenia problému. Toto riešenie problému môže znamenať stanovenie najvhodnejšieho cieľa alebo najvhodnejšej cesty vzhľadom k určenému cieľu teda hľadanie najvhodnejšieho riešenia vzhľadom na stanovené obmedzenia. Pokiaľ takéto riešenie existuje nazveme ho nazvať aj optimálne, samotné slovo optimum predstavuje najlepšie a najvhodnejšie riešenie, jeho ďalšie stupňovanie bez zmien v obmedzení nie je možné.

V matematickom ponímaní optimalizáciu možno zjednodušene vyjadriť ako hľadanie a určenie najvhodnejšej hodnoty funkcie vzhľadom na zvolené obmedzenie v rôznom priestore pri použití rôznych metód v závislosti od množstva premenných a typu funkcie. Pre zavedenie termínu konvexná optimalizácia je potrebné rozobrať niekoľko základných metód a typov problémov v optimalizácii.

1.1.1. Rozdelenie optimalizačných metód[3]

Stanovenie metódy optimalizácie sa stáva kľúčovou potrebou. Určenie optimalizačnej metódy je závislé na optimalizačnom probléme a jeho vhodné zvolenie sa určuje na základe presnosti, rýchlosti výpočtu, náročnosti na výpočet a celkovej funkčnej vhodnosti pre zadaný problém.

- Analytické metódy
 - Gradientové metódy
- Iteračné metódy
 - Jednoduché
 - Adaptačné
- Penalizačné metódy
- Komparatívne metódy

Analytické metódy sú výhodné svojou presnosťou pri jedno rozmerových prípadoch a jednoduchých tvaroch funkcií, zo stúpajúcou dimenziou problému a zložitou účelovej funkcie stúpa náročnosť a čas výpočtu, práve pri takýchto náročnejších úlohách nachádzajú uplatnenie iteračné metódy ktoré čiastočne zahŕňajú aj komparatívne metódy, tvoria najrozsiahlejšiu triedu optimalizačných metód, tieto metódy tvoria prevažne priamo programovateľné algoritmy ktoré skúmajú funkciu pomocou jednoduchých postupov v rôznych smeroch s definovanou presnosťou, následne vyhodnotia celý priebeh a porovnajú získané dáta, komparatívne metódy väčšinou skúmajú rôzne oblasti účelovej funkcie a získané výsledky porovnávajú. Penalizačné metódy sa používajú pri úlohách ktoré majú zadané viacero obmedzení ktoré sa môžu viac menej porušovať, respektíve každé obmedzenie má inú dôležitosť a priblíženie k niektorému alebo jeho prekročenie nemusí mať taký vplyv na výsledok ako prekročenie iného, teda táto metóda pomocou váhových koeficientov rôznych obmedzení určuje dôležitosť obmedzení penalizáciou optimalizácie, výsledné optimum má najmenšiu penalizáciu.

Každá metóda sa volí presne k danému problému, na jej správnej voľbe závisí či metóda nájde riešenie. Voľba metódy tiež priamo odráža druh hľadaného extrém.

1.1.1.1. Lokálna optimalizácia

Pri lokálnej optimalizácii sa hľadá extrém alebo optimálna hodnota funkcie vzhľadom na definovanú množinu prvkov alebo vymedzené územie. Takáto optimalizácia teda nájde najvhodnejšiu hodnotu len s malej oblasti funkcie. Pre riešenie lokálnej optimalizácie bolo vyvinutých veľmi veľa metód a algoritmov ktoré sú prijateľné pre svoju rýchlosť a jednoduchosť, dajú sa aplikovať na širokom spektre činností pri ktorých je oblasť hľadania optima známa a nie je veľmi veľká. Majú aj početné nevýhody ku ktorým samozrejme patrí často krát nemožnosť nájdania globálneho optima. [1] Každá metóda lokálnej optimalizácie vyžaduje určenie akéhosi štartovacieho bodu od ktorého sa bude optimalizácia uberať, jeho určenie má veľký vplyv na výslednú presnosť alebo nájdenie hľadaného optima, správne určenie tohto bodu býva tiež problematické.

1.1.1.2. Globálna optimalizácia

Globálna optimalizácia hľadá riešenie ktoré je globálne a teda nájdené optimum je platné pre celý rozsah funkcie. [1] Podstatnou výhodou je presné zmapovanie oblasti a jednoznačné určenie prípustných riešení a nevhodných riešení, takže aj keď lokálna optimalizácia nájde na určenej oblasti výhodné riešenie, pri globálnom pohľade to môže byť jedno s najhorších. Veľká nevýhoda týchto metód spočíva v obrovskom čase potrebnom na výpočet (hodiny, dni, roky, veky vesmíru), pri väčšine algoritmov sa počet krokov exponenciálne zväčšuje až dospeje do hodnoty pri bežných

podmienkach neriešiteľnú. Globálne metódy sa používajú pri menších systémoch a jednoduchších funkciách, kde je čas potrebný na výpočet v prípustných hraniciach.

1.1.2. Rozdelenie optimalizačných problémov[3]

Počas vývoja optimalizácie rôznych úloh bolo nevyhnutné pre stanovenie metódy hľadania optimálneho riešenia poznať aj problém ktorý je potrebný riešiť preto sa začali optimalizačné problémy klasifikovať podľa tvaru účelovej funkcie, typu obmedzení, počtu premenných a ich typu a následne kategorizovať pre rôzne typy metód riešenia.

Rozdelenie podľa typu premenných

- Spojitá
 - Spojité reálne čísla
 - Spojitá oblasť
- Celočíselná
 - Diskrétna
 - Množiny prvkov
 - Uzavreté intervaly

Rozdelenie podľa množstva premenných:

- Jedno rozmerová
- Mnoho rozmerová

Rozdelenie podľa obmedzení:

- Úlohy bez obmedzení
 - Nelineárne rovnice
 - Nelineárne úlohy najmenších štvorcov
 - Globálna optimalizácia
 - Nediferenciálna optimalizácia
- Úlohy s obmedzeniami
 - Lineárne programovanie
 - Nelineárne obmedzenia
 - Väzbové obmedzenia
 - Sieťové programovanie
 - Stochastické programovanie

1.2. Konvexná množina

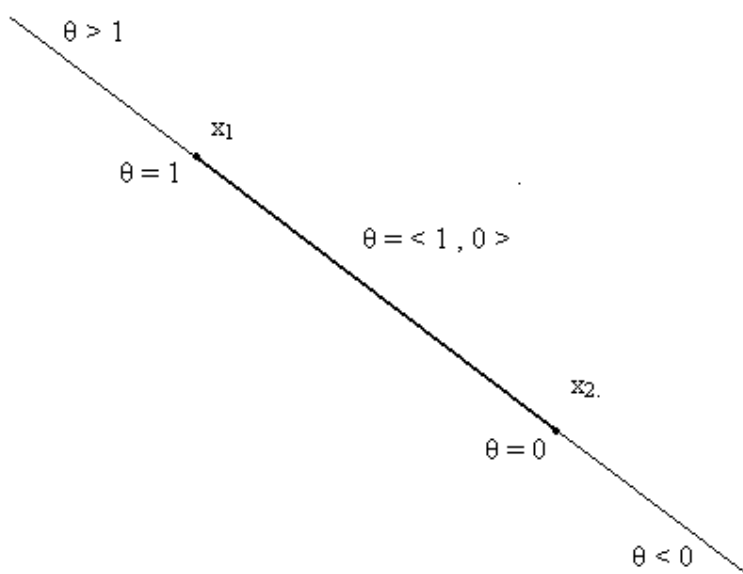
1.2.1. Spojitý priestor

Konvexná optimalizácia na rozdiel od iných oblastí optimalizácie pracuje výhradne v konvexných množinách alebo tiež konvexných útvaroch. Konvexná množina je typická tým že ak si vezmeme ľubovoľné dva body v množine a preložíme nimi priamku, všetky body ktoré ležia na priamke medzi týmito dvoma hraničnými bodmi musia patriť do konvexnej množiny.

Majme teda dva body na priamke x_1 a x_2 , nech platí $x_1 \neq x_2$ a $x_1 < x_2$, aby mohli všetky body ležať medzi dvoma bodmi na priamke musí byť daný priestor spojitý musí platiť [1]

$$y = x_2 + \theta(x_1 - x_2) \quad (1)$$

x_1 a x_2 sú hraničné body a y je nový bod medzi nimi, θ je parameter ktorý sa mení spojitou od 1 po 0 pričom ak $\theta = 1$ tak $y = x_1$ a ak $\theta = 0$ tak $y = x_2$, pre akúkoľvek inú hodnotu parametra θ dostaneme bod ležiaci na úsečke medzi dvoma bodmi. Slovné to možno vyjadriť že ak k prvej menšej súradnici bodu pripočítame nejaký násobok, nie však väčší ako jedna, vzdialenosti medzi obidvoma bodmi, dostaneme bod ktorý leží na priamke prechádzajúcej týmito bodmi a zároveň leží medzi dvoma danými bodmi. Leží teda na úsečke medzi bodmi x_1 a x_2 .

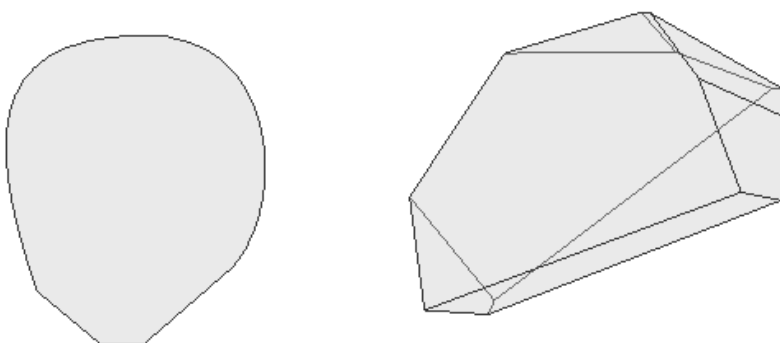


Obr.1 [1]

Jednoduchým menením parametra θ v hraniciach $\theta \in (0, 1)$ môžeme elegantne vyjadriť celú spojitú oblasť medzi dvoma hraničnými bodmi. Uvedený príklad samozrejme platí pre jednorozmerný svet priamky kde je poloha bodu daná len jednou súradnicou, definícia spojitého priestoru však rovnako funguje aj vo vyšších dimenziách kde sa v rovnici (1) nahradia body vektormi v rozmere príslušnej dimenzie.

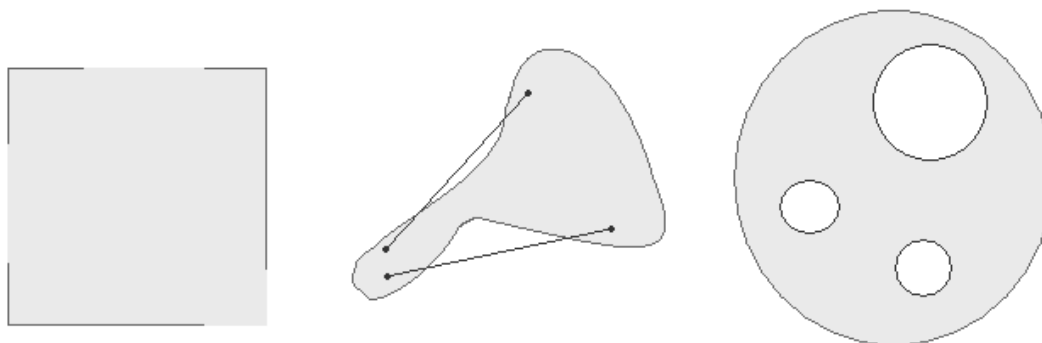
1.2.2. Konvexná oblasť

Konvexná oblasť vzniká ohraňčením n -rozmerného priestoru. Predstavme si n -rozmernú funkciu, jej možným riešením je pochopiteľne celý n -rozmerný priestor, keď k funkcii pridáme rozumný počet ohraňčení vznikne ohraňčený priestor ktorý reprezentuje možné riešenia funkcie, ak by sme hľadali optimálne riešenie funkcie, hľadali by sme ho už len v tomto útvere. Aby bola oblasť konvexná musí platiť že ak spojíme ľubovoľné dva body nachádzajúce sa v tomto priestore, tak všetky body nachádzajúce sa na tejto spojnici musia tiež patriť do danej oblasti, teda lineárna oblasť medzi dvoma bodmi nachádzajúcimi sa v tejto množine musí byť spojitá a musí sa nachádzať v konvexnej oblasti.



Obr.2

S ohľadom na použitý druh ohraňčenia môžu konvexné útvary v rôznych dimenziách nadobúdať rôzne tvary, základná podmienka však musí byť vždy splnená, na obrázku 2 môžeme vľavo vidieť jednoduchý konvexný útvar v 2D, na obrázku vpravo v 3D.



Obr. 3[1]

Pri použitých ohraňčeniach nemusí vždy vzniknúť konvexný útvar, častejšie vznikajú rôzne nekonvexné útvary. Na obrázku 3 vľavo môžeme vidieť útvar ktorý by mohol pripomínať konvexný útvar ale nie je úplne ohraňčený takže je nekonvexný, v strede je útvar na ktorom jasne vidieť že

spojnica medzi nie všetkými bodmi sa nachádza v útware, útvar vpravo nie je vnútorne spojitý platí preň rovnaký poznatok ako pre stredný útvar.

Konvexnou množinou nazývame všetky možné body alebo tiež riešenia ktoré sa nachádzajú v konvexnom útware. Pokiaľ hľadáme v danej množine optimálne riešenie, to predstavuje jeden bod s konvexnej množiny prípadne viacero bodov ktoré sú si rovné. S definície konvexnej množiny a konvexnej oblasti vyplýva jeden s hlavných dôvodov jej používania, ak nájdeme v konvexnom útware lokálne optimum tak sme automaticky našli aj globálne optimum, takže pojem optimum vždy znamená globálne, čo nám dáva voľnú ruku pri používaní metód na hľadanie lokálneho optima ktoré sú vo viacerých ohľadoch rýchlejšie a jednoduchšie ako metódy na hľadanie globálneho optima.

1.3. Konvexný optimalizačný problém

Pojmíme teda klasickú formulácie matematickej optimalizácie [1] (pre zovšeobecnenie sa budú jednoduché pojmy vo vzorcoch uvádzať v konvenčnom spôsobe a v angličtine)

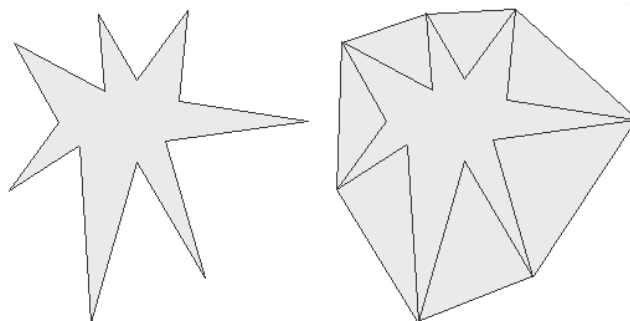
$$\begin{aligned} &\text{minimize} && f_0(x) \\ &\text{subject to} && f_i(x) \leq b_i, i=1,\dots,m \\ &&& h_i(x) = 0, i = 1, \dots, p \end{aligned} \tag{2}$$

Vektor x je vektor hodnôt optimalizovanej premennej, funkcia f_0 je účelová funkcia optimalizovanej premennej, funkcia $f_i(x)$ reprezentuje funkciu obmedzenia v tvare nerovnosti vektora x , b_i sú hranice obmedzení, funkcia $h_i(x)$ reprezentuje obmedzenia v tvare rovnosti, pokiaľ $m = p = 0$ účelová funkcia nemá obmedzenia a problém je neohraničený. Tento zápis sa dá chápať ako minimalizovať funkciu premennej vzhľadom na obmedzenia. Samozrejme že funkciu je možné aj maximalizovať a intervaly ohraničení zapísať iným spôsobom, uvedený spôsob zápisu len rešpektuje zaužívanú konvenciu zapisovania, maximalizovať znamená minimalizovať zápornú funkciu preto sa dá každé hľadanie požadovaného extrému zapísať ako minimalizácia, funkcie premennej v obmedzení sa budú písať na ľavú stranu so zachovaním smeru nerovnosti, teda funkcia optimalizovanej premennej sa vždy zapíše ako rovná prípadne menšia ako hranica ohraničenia, túto konvenciu tiež možno ľahko vyjadriť pomocou matematických úprav s akejkoľvek formulácie obmedzení. Takéto zadanie má riešenie práve vtedy ak všetky funkcie obmedzení majú aspoň jeden spoločný bod, inak je problém neriešiteľný (ang. Infeasible). Riešenie tohto problému sa odvíja od funkcie účelovej premennej ale najmä od funkcie obmedzení práve ich tvar určuje zložitosť problému a voľbu vhodnej metódy na riešenie.

1.4. Lineárny konvexný optimalizačný problém

1.4.1. Praktické aspekty

Ako bolo naznačené v stati 1.2.2 konvexná optimalizácia pracuje s konvexnými útvarmi, s praktického aspektu však množiny riešení v reálnych problémoch nie sú vždy konvexné, tu však konvexná optimalizácia nachádza uplatnenie tiež. Predstavme si silne nekonvexný útvar ktorý vznikne ohraničením množiny riešení praktického problému, vid' obrázok 4 vľavo.

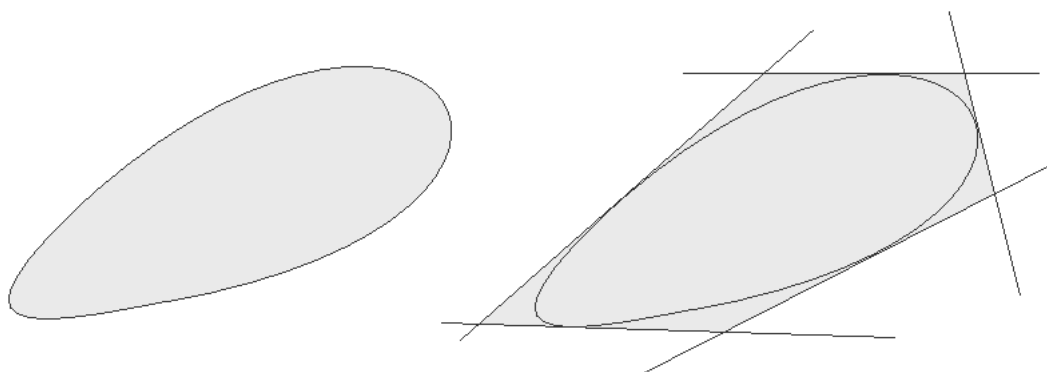


Obr.4

Hľadanie požadovaného optima by bolo v takomto telese veľmi zložitú, aj keď samozrejme v 2D to pripadá veľmi jednoducho graficky, pri vyšších dimenziách to nie je také jednoduché. Krása konvexnej optimalizácie tkvie v tom že aj s tých najsilnejších nekonvexných množín riešenia možno urobiť pridaním jednoduchých lineárnych alebo nelineárnych ohraničení konvexnú množinu ktorá je už viac menej ľahko riešiteľná, pokiaľ nájdeme vo vzniknutom konvexnom útvaru optimum túto hodnotu priblížime k najbližšej hodnote so skutočnej množiny riešení a našli sme optimum aj v nekonvexnej oblasti vid' obrázok 4 vpravo. Pri tomto spôsobe hľadania optima dochádza k istej strate presnosti, v praxi sa však práve takéto riešenie problémov ukázalo ako veľmi rýchle a dostatočne presné na efektívne používanie.

Prečo boli na vzniknutie konvexného útvaru použité práve lineárne ohraničenia ? Na ohraničenie nekonvexného útvaru môžu byť použité aj iné nelineárne ohraničenia najčastejšie kruhové útvary s definovaným zakrivením. Lineárne ohraničenia sú najjednoduchším typom ohraničení, najľahšie sa zapisujú, najľahšie sa na nich pohybuje, veľmi jednoducho sa hľadajú ich prieniky a rovnako sa na nich aj jednoducho hľadajú extrémny alebo optimá. Nelineárne ohraničenia sa používajú tiež, ale vyžadujú si špeciálnu triedu riešení ktorá je presne určená pre daný typ problému. Lineárne ohraničenia dokážu so stratou určitej presnosti vytvoriť s akéhokoľvek zložitého nekonvexného útvaru jednoduchý konvexný útvar alebo po častiach konvexný útvar. Lineárne ohraničenia majú aj mnoho iných výhod, predstavme si konvexné teleso ktoré nemá lineárne ohraničenia, nájsť v takomto telese optimum je možné s použitím konkrétnej metódy pre druh nelineárneho ohraničenia alebo všeobecnej metódy (napr. gradientová metóda), tieto metódy

nemusia byť náročné na zložitosť a niekedy ani na čas výpočtu, predstavme si však že takýto konvexný útvar s nelineárnymi ohraničeniami je nahradený približným konvexným útvarom s lineárnymi ohraničeniami.



Obr.5

Na obrázku 5 je hrubo vidíme hrubý názorný príklad ako by takéto priblíženie vyzeralo, vľavo vidíme konvexný útvar s nelineárnymi ohraničeniami, vpravo vidíme tento útvar nahradený aproximovaným modelom s lineárnymi ohraničeniami.

Prečo by sa niečo takéto vykonávalo ? Aj tá najjednoduchšia metóda na hľadanie extrém v konvexnom útvere bez lineárneho ohraničenia je svojou zložitou časovo náročnejšia než akákoľvek metóda na hľadanie extrém v konvexnom útvere s lineárnymi ohraničeniami čo sa ukázalo ako kľúčové pri riešení reálnych problémov s tisíckami premenných a ohraničení kde veľkosť strojového času potrebného na výpočet bola neprijateľná. Lineárne aproximácie pre opis útvaru na obrázku 5 vpravo sú len ilustračné, dané konvexné teleso by sme mohli opísať stovkami lineárnych ohraničení a stále by výpočet extrému trval zlomok sekundy strojového času, počet ohraničení a voľba metódy závisia od konkrétnej situácie a požadovanej presnosti. Preto sú lineárne ohraničenia také populárne.

1.4.2. Formulácia problému

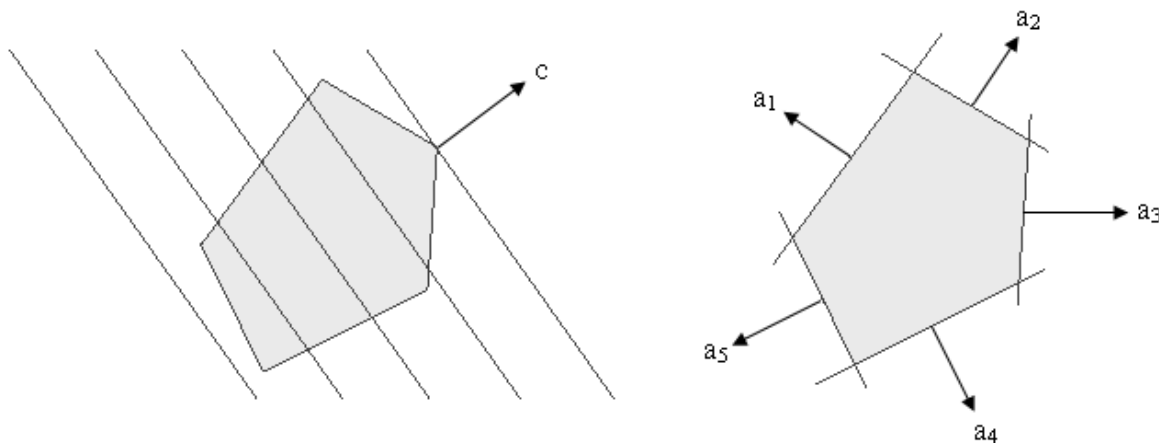
Keď sú všetky obmedzenia v tvare nerovnosti a rovnosti lineárne rovnako ako účelová funkcia môžeme problém nazvať lineárny konvexný optimalizačný problém a môžeme ho vzhľadom na zaužívanú konvenciu zapísať nasledovne [1] :

$$\begin{aligned} &\text{minimize} && c^T x + d \\ &\text{subject to} && Gx \leq h \\ &&& Ax = b \end{aligned} \tag{3}$$

Pričom x je vektor hodnôt optimalizovanej premennej, c^T je transponovaný vektor konštánt lineárnej účelovej funkcie, d konštanta je posunutie účelovej funkcie o rozmere. G je matica hodnôt obmedzení v tvare nerovnosti o rozmere $m \times n$, A je matica obmedzení v tvare rovnosti o rozmere

$p \times n$, h je vektor hraníc obmedzenia pre v tvare nerovnosti. Symbolické označovanie písmenami nie je fixné pretože matice G a A sa môžu zamieňať v závislosti o transformácie problému alebo značenia v zadaní.

Pokiaľ sa v takéto ohraničenia pretnú, prienikom všetkých lineárnych ohraničení bude ohraničený útvar ktorý vznikol na obrázku 6.vpravo sú v ňom naznačené aj smerové vektory polpriestorov obmedzení, ďalšou výhodou lineárnych ohraničení je vzniknutý útvar je vždy konvexný. Takto vzniknutý konvexný útvar ohraničený lineárnymi ohraničeniami sa nazýva polytop, v niektorých druhoch literatúry aj polyhedrón, ďalej



Obr.6 [1]

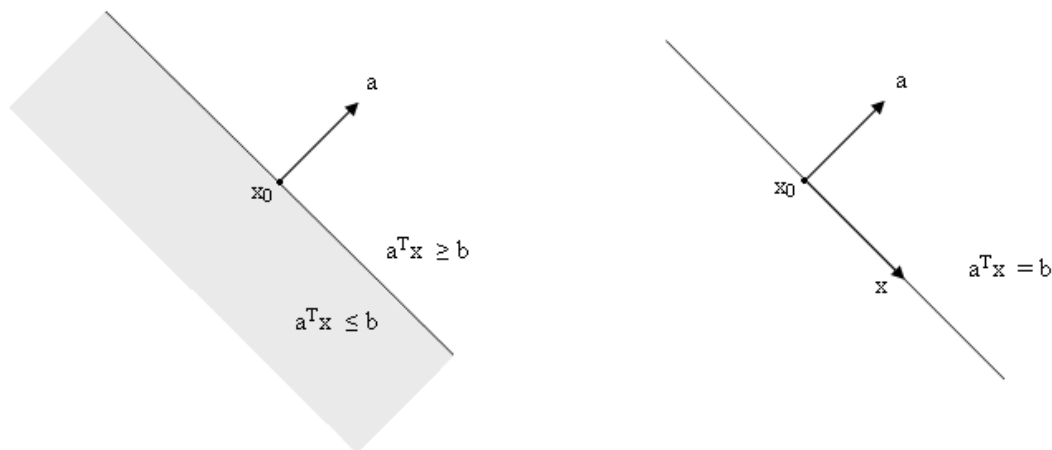
bude používané práve pomenovanie polytop. Hľadať extrém v takomto útware znamená hľadať také riešenie s polytopu pri ktorom má účelová funkcia najmenšiu hodnotu, teda pre dvojrozmerný prípad možno graficky toto hľadanie predstaviť ako posúvanie priamky danej účelovou funkciou v smere vektora c účelovej funkcie tak ako je to zobrazené na obrázku 6 vľavo, narazenie na hranicu znamená nájdenie najmenšej možnej hodnoty, nakoľko sme v konvexnom útware takáto hodnota je najmenšia možná s celej množiny a nájdený bod je hľadané minimum. Dôležitý poznatok vyplývajúci s pohybu v konvexnej množiny je že extrém alebo optimum vždy leží na okraji polytopu alebo iného konvexného útvaru, buď v jeho vrchole alebo na jeho strane. Niektoré metódy hľadania extrému používajú práve pohyb po okraji konvexnej množiny. Riešenie lineárneho konvexného problému sa zvykne nazývať aj lineárne programovanie.

1.4.3. Vznik lineárnej konvexnej množiny

1.4.3.1. Polorovina a polpriestor

Pri klasifikácii obmedzení v lineárnom konvexnom probléme prichádzame na dva základné typy obmedzení a to obmedzenie v tvare nerovnosti a obmedzenie v tvare rovnosti. Predstavme si n-rozmerný priestor a jedno lineárne obmedzenie v tvare nerovnosti, takéto obmedzenie nám celý

priestor možných riešení účelovej funkcie rozdelí na dve časti, jednu kde sa riešenie smie nachádzať a kde sa riešenie nachádzať nesmie. Teda celý priestor rozdelí na dva polpriestory s definovanou hranicou. Nech a je ľubovoľný vektor, nech x vektor premenných a b je hranica posunutia. potom možno polorovinu definovať geometricky ako $a^T x \leq b$ pre obmedzenie v tvare nerovnosti a $a^T x = b$ pre obmedzenie v tvare rovnosti. Na obrázku 7 môžeme vidieť vektorové



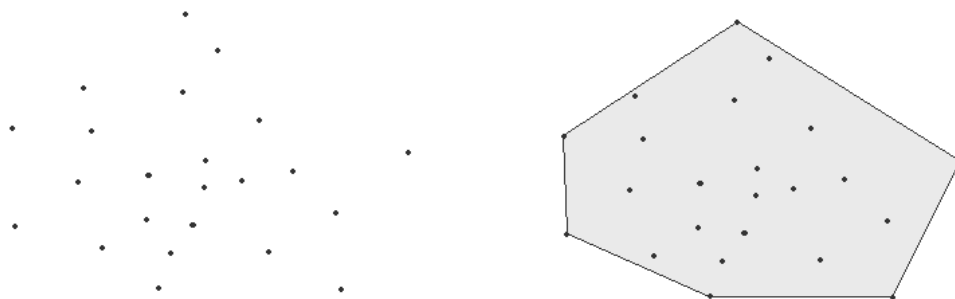
Obr.7 [1]

znázornenie, vektor a je kolmý na danú polorovinu pričom naľavo vidíme obmedzenie v tvare nerovnosti ako hranicou tohto obmedzenia vytyčuje polpriestor v n -rozmernom priestore, x_0 je začiatkový bod sústavy pre dané posunutie, ak by sme chceli zistiť či je daný bod v povolenom polpriestore stačilo by ho vektorovo vynásobiť s vektorom a a následne už len porovnať s posunutím. Zatiaľ čo obmedzenie v tvare rovnosti definuje v danom n -rozmernom priestore polorovinu na ktorej sa musí bod nutne nachádzať, s obrázku 6 vpravo vyplýva že vektor daného bodu musí byť ortogonálny na vektor a aby mohol výsledný vzorec platiť.

Z uvedených skutočností vyplýva dôležitý poznatok, obmedzenie v tvare nerovnosti nám s n -rozmerného priestoru určí časť v ktorej sa môže n -rozmerné riešenie problému nachádzať. Obmedzenie v tvare rovnosti určí s n -rozmerného priestoru takú oblasť tohto priestoru v ktorej sa musí n -rozmerné riešenie problému nachádzať, aby rovnosť platila musí mať n -rozmerné riešenie jednu súradnicu s vektora konštantnú, takže môžeme povedať že obmedzenie v tvare rovnosti nám zmenší dimenziu problému o jeden stupeň, uvedená analógia sa dá použiť aj pre viacero obmedzení v tvare rovnosti, pokiaľ sa tieto nerovnejú tak každé ďalšie obmedzenie zníži dimenziu problému o jeden stupeň. V praxi sa však pri zavádzaní v tvare rovnosti namiesto s požadovaným zjednodušením výpočtu zmenšením dimenzie častejšie stretávame s neriešiteľnosťou problémov, preto sa namiesto obmedzení v tvare rovnosti používajú častejšie obmedzenia v tvare nerovnosti ktoré sú blízke obmedzeniam v tvare rovnosti.

1.4.3.2. Druhy vznikov

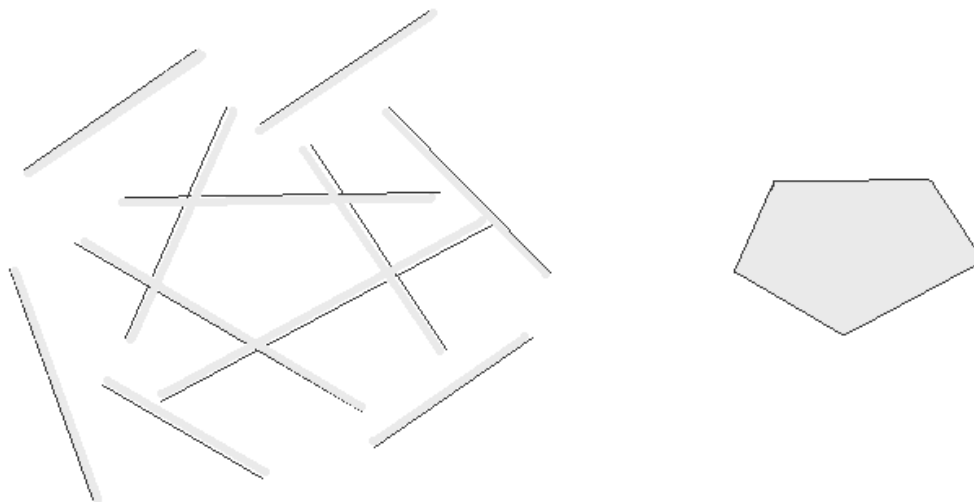
Lineárna konvexná množina môže vzniknúť dvomi spôsobmi. Prvý z nich sa používa keď nepoznáme ohraničenia ale len množinu riešení v n -rozmernom priestore, v takomto prípade sa najprv jednoduchým porovnávaním vzdialeností bodov riešení určia okrajové body množiny, ich spojením vzniká ohraničený lineárny konvexný útvar tak ako to znázorňuje obrázok 8.



Obr.8

Vľavo môžeme vidieť množinu riešení predstavujúcu body v priestore, vpravo môžeme vidieť vznik konvexného telesa spojením okrajových bodov.

Druhý spôsob ako vytvoriť konvexné teleso je keď poznáme všetky ohraničenia v tvare nerovnosti, tieto ohraničenia vymedzujú v priestore pol roviny ktorých prienik vymedzuje konvexný lineárny útvar. Pri vymedzovaní tohto priestoru často narážame na problém redundantných ohraničení, teda ohraničení ktoré nie sú potrebné na vymedzenie konvexnej oblasti prípadne sú rovnobežné s inými



Obr.9

ohraničeniami alebo ich iné ohraničenia už vylúčili tak ako môžeme vidieť na obrázku 9, vľavo vidíme všetky stanovené obmedzenia pre problém, vpravo vidíme skutočný tvar konvexného telesa (polytopu) po zjednotení prieniku všetkých obmedzení. Existuje aj druhá situácia keď prienik všetkých ohraničení je prázdna množina, problém vtedy za daných obmedzení nemá riešenie a jeho obmedzenia sú disjunktné.

Vykonať takúto redukciu redundantných ohraničení možno jednoduchým algoritmom ktorý si môžeme ukázať na jednoduchom príklade. Nech je náš príklad dvojrozmerný teda pohybujeme sa v dvojrozmernom svete kde je riešením problému celý R^2 a súradnice bodu tvorí dvojprvkový vektor. Majme zadanú maticu A reprezentujúcu maticu ľavej strany obmedzení v tvare nerovnosti a stĺpcový vektor hraníc obmedzení b. Vybrané dáta boli generované náhodne ale pre názornosť riešenia samozrejme boli vybrané tak aby mali riešenie, teda aby prienik všetkých ohraničení existoval. Hodnoty dát :

$$A = \begin{bmatrix} -0.0877 & 0.9961 \\ -0.2667 & 0.9638 \\ -0.3624 & -0.9320 \\ -0.7850 & 0.6194 \\ -0.1850 & -0.9827 \\ -0.2551 & 0.9669 \\ -0.0654 & -0.9979 \\ 0.9627 & 0.2705 \end{bmatrix} \quad B = \begin{bmatrix} 0.7065 \\ 0.4866 \\ 1.5392 \\ 0.9787 \\ 1.3395 \\ 0.4911 \\ 1.3809 \\ 2.1194 \end{bmatrix}$$

Geometricky možno tieto obmedzenia prirovnať k obrázku 8 vľavo. Spôsob určenia výsledného polytopu sa vykoná určením jeho okrajových bodov. Každé obmedzenie predstavuje polorovinu ktorá vymedzuje polpriestor, ak sa tieto dve poloroviny pretnú udajú vrcholový bod polytopu ak sú tieto ohraničenia okrajové ohraničenia polytopu tak musí platiť že bod prieniku je v platnej oblasti každého ohraničenia, teda keby sme bod dosadili do klasickej rovnice $Ax \leq b$ tá musí byť splnená. zistením všetkých prienikov a následné vyhodnotenie tohto bodu ako okrajového bodu polytopu nám udáva výsledný algoritmus na určenie polytopu. Ten by sa v Matlabe realizoval nasledovne:

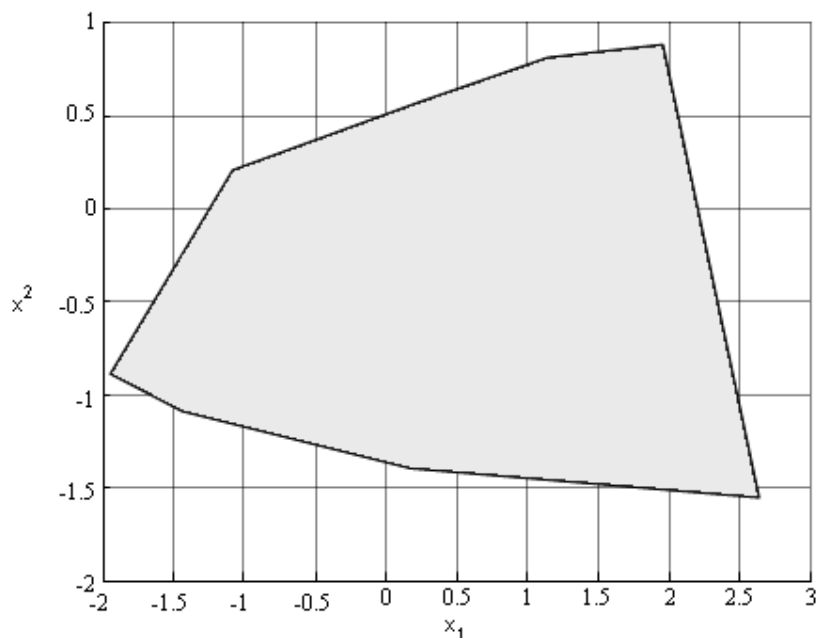
```
bod = [];  
S=0;
```

```
for S=1:(size(A, 1)-1)  
    for C = (S+1):(size(A, 1))  
        a1=A(S,1:2);  
        a2=A(C,1:2);  
        a=cat(1,a1,a2);  
  
        b1=B(S,1);  
        b2=B(C,1);  
        b=cat(1,b1,b2);  
  
        x=inv(a)*b;  
  
        if all(A*x<=B)  
            bod=[bod x];  
        end  
    end  
end
```

Uvedený algoritmus postupuje nasledovne, pomocou dvoch cyklov odskúša všetky kombinácie dvoch polorovín, v každom preskúšaní najprv vyberie s hlavnej matice obmedzení dva obmedzenia ktoré príkazom `cat` spojí do jednej matice, spojením obmedzenia a hranice obmedzenia vzniknú dve matice, pre dvojrozmerný je výsledný bod riešením dvoch rovníc o dvoch neznámych, zistený bod sa následne porovná s každým obmedzením príkazom `all` ktorý vráti 1 v prípade pravdy a 0 v prípade nepravdy, ak je tento bod skutočne platný pre všetky obmedzenia zaradi sa do výslednej matice bod reprezentujúcej okrajové body polytopu. Pre tento príklad má matica bod nasledovné hodnoty :

$$\text{bod} = \begin{bmatrix} 1.1449 & 1.9539 & 0.2372 & -1.9511 & -1.4374 & 0.1700 & 2.6388 \\ 0.8100 & 0.8812 & 0.5705 & -0.8927 & -1.0924 & -1.3950 & -1.5569 \end{bmatrix}$$

Vykreslenie tohto polytopu je znázornené na obrázku 9. (napr. príkazom `polytope(A,B)` s knižnice programového balíka Yalmip)



Obr.10

Graficky je možné overiť že všetky body ktoré program určil sú skutočne okrajové body polytopu. Táto metóda ukazuje najjednoduchšiu metódu nájdenia okrajov polytopu s množiny ohraničení, pri vyšších dimenziách strojový čas na výpočet narastá pretože prieniky obmedzení nie sú len body ale aj priamky, roviny a viacrozmerné nedefinované útvary, stúpa počet kombinácií ohraničení a ich vrcholov, pre vyššie dimenzie sa používajú iné nekomplexné metódy ktoré sú schopné ohraničenie vylúčiť vzhľadom na iné ohraničenie pričom sa čas výpočtu prijateľne skrúti.

1.5. Transformácie lineárneho konvexného optimalizačného problému

Pri riešení optimalizačného problému je väčšinou ťažšie problém vytvoriť než ho vyriešiť, na riešenie optimalizačných problémov sa vytvorilo veľké množstvo metód, ktoré sú dnes prakticky všetky už integrované ako softwarové programy ktoré vyžadujú špecifické zadanie problému. Vytvorenie problému sa preto pre praktické situácie stáva náročnejším ako samotné vyriešenie problému. Každý riešiteľ vyžaduje presné zadanie problému v predpísanom tvare, tento tvar je potrebné vytvoriť alebo už získať s iného tvaru optimalizačného problému. Prednosti používania lineárneho optimalizačného problému alebo lineárneho programovania boli rozobraté v časti 1.4.1., ďalej sa budeme zaoberať práve transformáciami lineárneho problému. Aby sme dokázali plnohodnotne pracovať s nástrojmi riešenia problémov musíme vedieť transformovať zadaný problém do tvaru ktorý vyžaduje riešiteľ(solver) s ktorým ideme pracovať.

1.5.1. Transformácia štandardných foriem

Všeobecný problém zadania lineárneho konvexného problému má tvar [1] :

$$\begin{aligned} &\text{minimize} && c^T x + d \\ &\text{subject to} && Gx \leq h \\ &&& Ax = b \end{aligned} \tag{4}$$

Často je výhodné túto základnú formu previesť na štandardné formy ktoré majú tvar len vo forme nerovností

$$\begin{aligned} &\text{minimize} && c^T x \\ &\text{subject to} && Ax \leq b \end{aligned} \tag{5}$$

alebo v tvare nerovností s jedným obmedzením v tvare nerovnosti väčšinou pre kladnosť optimalizovanej premennej

$$\begin{aligned} &\text{minimize} && c^T x \\ &\text{subject to} && Ax = b \\ &&& x \geq 0 \end{aligned} \tag{6}$$

Rovnica (6) by sama o sebe mohla reprezentovať všeobecný zápis lineárneho konvexného problému, naučenia transformácie týchto dvoch foriem medzi sebou patrík základným formám manipulácie zo zadaním problému.

Majme zadaný problém ktorý je zadaný v tvare rovností a podmienky kladnosti riešenia (vektor optimalizovanej premennej musí byť väčší ako nula). Potom zadanie tohto problému vyzerá nasledovne :

$$\begin{aligned} &\text{minimize} && c^T x \\ &\text{subject to} && Ax = b \\ &&& x \geq 0 \end{aligned} \tag{7}$$

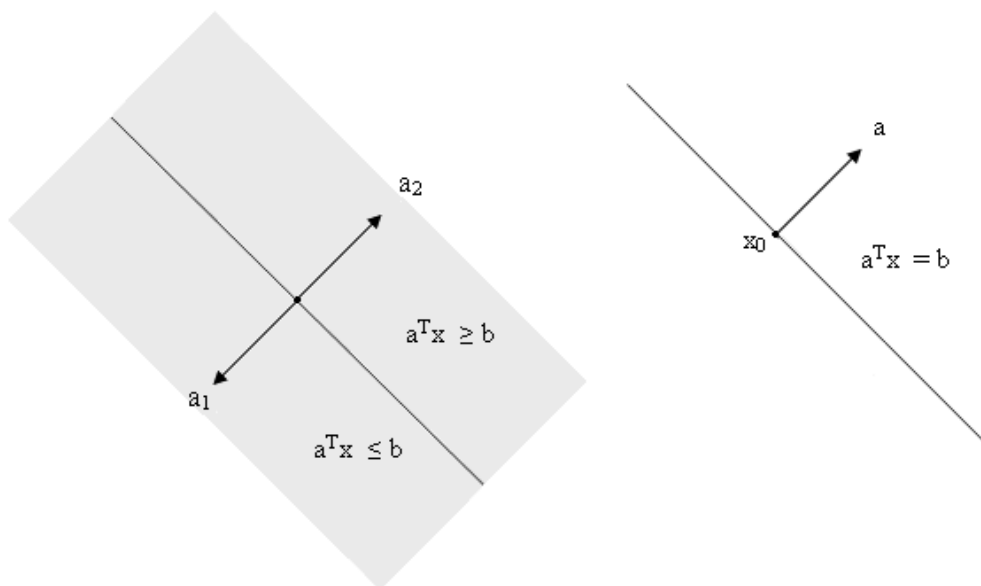
(Pri zápise sme zanedbali konštantu d , budeme predpokladať že pre zadaný prípad je nulová) Pre správne zavedenia a používanie konvencie sa zadanie trochu upraví do správnej podoby:

$$\begin{aligned} &\text{minimize} && c^T x \\ &\text{subject to} && Ax = b \\ &&& -x \leq 0 \end{aligned} \tag{8}$$

Riešiteľ v ktorom pracujeme však akceptuje zadanie len v tvare nerovností :

$$\begin{aligned} &\text{minimize} && c^T x \\ &\text{subject to} && Ax \leq b \end{aligned} \tag{9}$$

Vstupné zadanie je možné transformovať a upraviť po akceptácii základných faktov. Obmedzenie v tvare rovnosti nám v n -rozmernom priestore vytína polorovinu na ktorej sa musí nachádzať riešenie, takúto polorovinu môžeme v danom priestore nahradiť prienikom dvoch polpriestorov na



Obr.11 [1]

rovnakej hranici ktorú reprezentuje obmedzenie v tvare rovnosti, tak ako to zobrazuje obrázok 11 obmedzenie v tvare rovnosti môžeme preto nahradiť nasledovnými obmedzeniami:

$$\begin{aligned} Ax = b &\leftrightarrow Ax \leq b \\ &Ax \geq b \end{aligned} \tag{10}$$

Získali sme teda nasledovnú formuláciu problému [1]:

$$\begin{aligned} &\text{minimize} && c^T x \\ &\text{subject to} && Ax \leq b \\ &&& -Ax \leq -b \\ &&& -x \leq 0 \end{aligned} \tag{11}$$

Získali sme iba obmedzenia v tvare nerovností a môžeme ich zapísať ako jeden vektor

$$\tilde{A} = \begin{bmatrix} A \\ -A \\ -I \end{bmatrix} \quad \tilde{B} = \begin{bmatrix} b \\ -b \\ 0 \end{bmatrix} \quad (12)$$

I je jednotková matica o veľkosti vektora optimalizovaných premenných, veľkosťou je jej ekvivalentná nulová matica ktorá zohľadňuje obmedzenie kladnosti. Výsledná transformácia problému vyzerá:

$$\begin{aligned} &\text{minimize} \quad c^T x \\ &\text{subject to} \quad \tilde{A}x \leq \tilde{B} \end{aligned} \quad (13)$$

V Matlabe by sa dala táto transformácia realizovať nasledovne, v uvedenom príklade sa problém aj rieši a výsledky sa navzájom porovnávajú. Všetky problémy sú riešené pomocou softwarového balíka Yalmip ktorý je zameraný práve na riešenie optimalizačných problémov v konvexných množinách.

```
A=rand(2,8);
```

```
B=rand(2,1);
```

```
C=rand(8,1);
```

```
nx = size(A, 2);
```

```
x=sdpvar(nx, 1)
```

```
% Povodne formulacia problemu:
```

```
% min C'*x
```

```
% s.t. A*x = B
```

```
% x >= 0
```

```
obj = C'*x;
```

```
F = [ A*x == B; x >= 0 ];
```

```
solvesdp(F, obj)
```

```
obj_opt = double(obj);
```

```
xopt = double(x);
```

```
all(abs(A*xopt - B) < 1e-10)
```

```
% Reformulacia pomocou obojstrannych nerovnosti
```

```
% min C'*x
```

```
% s.t. A*x <= B
```

```
% A*x >= B
```

```
% x >= 0
```

```
Ac = [ -eye(nx); A; -A ];
```

```
Bc = [ zeros(nx, 1); B; -B ];
```

```
Fc = [Ac*x <= Bc];
```

```
objc = C'*x;
```

```
solvesdp([Ac*x <= Bc], C'*x)
```

```
obj_opt_r = double(objc)
```

```
xopt_trans = double(x);
```

```
% Vysledne porovnanie vypocitanych hodnot
```

```
xopt_trans
```

```
xopt
```

V úvode kódu ako vstupné parametre zadávame náhodné premenné, takéto zadanie je nebezpečné pretože náhodné ohraničenia nemusia mať vždy riešenie a výsledný problém je neriešiteľný (infeasible) ak daný kód použijeme viackrát uvidíme že výsledné hodnoty x_{opt_trans} a x_{opt} , ktoré obsahujú výsledok optimalizácie pred a po transformácii, uvidíme že sa pri akýchkoľvek riešiteľných hodnotách vstupných vektorov rovnajú.

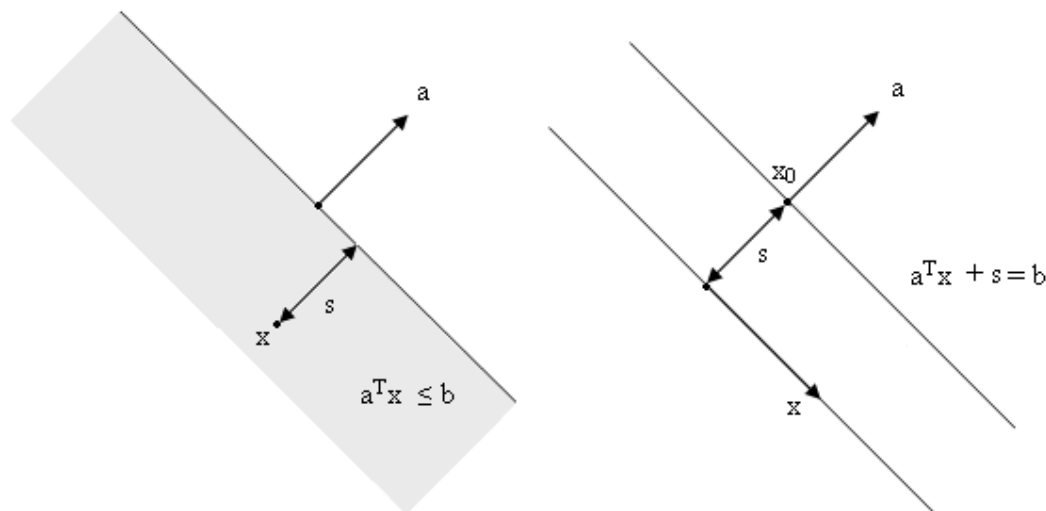
1.5.2. Transformácia s pomocnými premennými

Predchádzajúca transformácia nám ukázala ako jednoducho vyjadriť obmedzenie v tvare nerovnosti ako obmedzenie v tvare rovnosti, opačná transformácia už nie je taká elegantná a vyžaduje si pre korektnú transformáciu zavedenie nových optimalizovaných pomocných premenných (slack variables).

Majme zadaný lineárny optimalizačný problém reprezentujúci základný zjednodušený tvar (zanedbáme obmedzenie $Gx = h$ kvôli skráteniu postupu)

$$\begin{aligned} &\text{minimize} && c^T x \\ &\text{subject to} && Ax \leq b \end{aligned} \tag{14}$$

Obmedzenie v tvare nerovnosti je polpriestor vymedzený hranicou obmedzenia, pričom hľadané riešenie sa nachádza niekde v povolenom polpriestore, nech sa takéto riešenie nachádza kdekoľvek vždy je rovné danej hranici ak k nemu pripočítame vzdialenosť od hranice, graficky by sme to



Obr.12

mohli zobraziť tak ako na obrázku 12 táto vzdialenosť je nezáporná a nazvime ju s , doplníme teda zistenú skutočnosť do rovnice (14)

$$\begin{aligned} &\text{minimize} && c^T x \\ &\text{subject to} && Ax + s \leq b \\ &&& s \geq 0 \end{aligned} \tag{15}$$

Ďalej uvažujme, každé číslo vieme zapísať ako rozdiel dvoch nezáporných čísel

$$x = x^+ - x^- \quad (16)$$

$$x^+ \geq 0$$

$$x^- \geq 0$$

pokiaľ rovnice (16) doplníme do rovníc (15) získame nasledovné rovnice [1]:

$$\text{minimize} \quad c^T x^+ - c^T x^- + s \quad (17)$$

$$\text{subject to} \quad Ax^+ - Ax^- + s = b$$

$$s \geq 0$$

$$x^+ \geq 0$$

$$x^- \geq 0$$

Pre získanie základného tvaru ešte sformulujem matice konštánt do jednotlivých matíc:

$$\tilde{C}^T = [c^T, -c^T, I] \quad (18)$$

$$\tilde{A} = [A, -A, I]$$

$$z = \begin{bmatrix} x^+ \\ x^- \\ s \end{bmatrix}$$

Základný zápis štandardnej formy potom vyzerá :

$$\text{minimize} \quad \tilde{C}^T z + s \quad (19)$$

$$\text{subject to} \quad \tilde{A}z = b$$

$$z \geq 0$$

Zanedbanie obmedzenia $Gx = h$ nemalo žiadny význam, finálna transformácia má obmedzenia v tvare rovnosti, v prípade že by sme obmedzenie použili a pracovali s ním počas transformácie nič by to na systéme transformácie nemenilo, zmenili by sa len veľkosti finálnych tilde matíc ktoré by mali príslušný rozmer zväčšený o parametre obmedzenia. V Matlabe si môžeme funkčnosť transformácie jednoducho overiť tým že najprv vypočítame riešenie pôvodného problému a porovnáme ho s riešením transformovaného problému presne, uvedené skutočnosti vykonáva nasledovný kód v m-file:

```
A=randn(20,8);
B=rand(20,1);
C=rand(8,1);
nx = size(A, 2);
nc = size(A, 1);
```

```

% Povodny problem:
% min c'*x
% s.t. A*x <= B
x = sdpvar(nx, 1);
F = [ A*x <= B ];
obj = C'*x;
solvesdp(F, obj)
obj_opt = double(obj);
x_opt = double(x);

% transformovany problem:
% min ct'*z
% s.t. At*z = Bt
%      z >= 0
%      z = [xp; xm; s]
%
% xopt = xp_opt - xm_opt

xp=sdpvar(nx, 1);
xm=sdpvar(nx, 1);
s=sdpvar(nc, 1);

Atilde = [A -A eye(nc, nc)];
Btilde = B;
Ctilde = [C; -C; zeros(nc, 1)];

z = [xp; xm; s];
Ftilde = [ Atilde*z == Btilde ] + [z >= 0 ];
obj_tilde = Ctilde'*z;
solvesdp(Ftilde, obj_tilde);
obj_tilde_opt = double(obj_tilde)
z_opt = double(z);
xp_opt = z_opt(1:nx);
xm_opt = z_opt(nx+1:2*nx);
s_opt = double(s);
x_tilde_opt = xp_opt - xm_opt;

[obj_opt obj_tilde_opt]
[x_opt'; x_tilde_opt']

max(A*x_opt - B) < 1e-10

```

V úvode kódu sa opäť generujú náhodné dáta, pri niektorých pokusoch o spustenie problém nemusí vykazovať riešiteľnosť preto je potrebné spustiť ho viackrát v prípade neriešiteľnosti. ďalej kód definuje symbolické premenné ktoré nadobudnú rozmer neskôr, vytvorí premenné ktoré požaduje riešiteľ a problém rieši, rovnako postupuje pri transformovanom probléme, pri spustení vidíme že výsledok optimalizácie obidvoch metód je rovnaký, uvedená transformácia skutočne funguje.

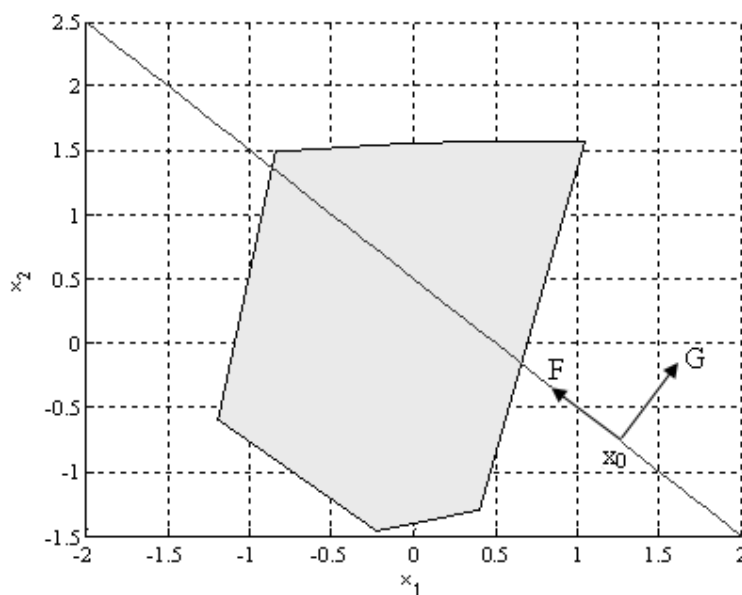
1.5.3. Podpriestorová transformácia

Ako bolo spomenuté v časti 1.4.3.1 obmedzenie v tvare rovnosti nám pre každú premennú zníži dimenziu priestoru o jeden stupeň, v prípade rovnosti všetkých premenných má systém buď jedno exaktné určené riešenie alebo žiadne, takéto zadanie sa v praxi z pochopiteľných dôvodov nevyskytuje nakoľko by tým zanikla potreba riešenia problému a jeho optimalizácie. Uplatnením obmedzenia v tvare rovnosti vznikne podpriestor, riešenie tohto priestoru sa nachádza práve v ňom skúsme teda riešiť problém len v podpriestore s vymedzenými obmedzeniami.

Majme zadaný konvexný lineárny optimalizačný problém s obmedzeniami v tvare nerovnosti aj tvare rovnosti, zadaný problém vyzerá štandardne:

$$\begin{aligned} &\text{minimize} && c^T x + d \\ &\text{subject to} && Ax \leq b \\ &&& Gx = h \end{aligned} \tag{20}$$

Graficky môžeme vzniknutý podpriestor znázorniť na obrázku 13, pre dvojrozmerný systém jedno



Obr.13

obmedzenie v tvare rovnosti vytvorí priamku ktorá po prieniku obmedzení utvorí úsečku riešeni. riešenie sa nachádza práve na nej. matica G je v tomto prípade smerový vektor tejto priamky alebo poloroviny, vektor kolmý na G sa musí nachádzať na tejto polorovine, nazvime ho F, potom sa dá akýkoľvek bod, ktorý by mohol byť riešením a nachádza sa na priamke G, vyjadriť ako násobok vektora F a násobku vzdialenosti od začiatkovej hranice, nazvime túto vzdialenosť z a počiatkový bod x_0 potom

$$x = F \cdot z + x_0 \tag{21}$$

Aby platila ortogonálnosť vektorov musí byť ich vektorový súčet rovný nule

$$G \times F = 0 \quad (22)$$

túto rovnosť dosiahnuť jednoducho za predpokladu že F je vektor kolmý na vektor G , v Matlabe sa takýto vektor získa príkazom `null()` s argumentom vektora na ktorý chceme kolmý vektor získať.

vektor x_0 udáva počiatočnú hranicu alebo tiež vektor ku ktorému po pripočítaní násobku vektora F získame optimálnu hodnotu x , môžeme ho získať vyjadrením s obmedzenia v tvare rovnosti.

$$\begin{aligned} Gx_0 &= h \\ x_0 &= G^{-1}h \end{aligned} \quad (23)$$

Tu by sme mohli naraziť na problém utvorenia inverznej matice v prípade neštvorcovej matice, riešenie nám ponúka pseudoinverzia matice pre ktorú platia vzťahy :

$$\begin{aligned} G \cdot G^\dagger \cdot G &= G \\ G^\dagger \cdot G \cdot G^\dagger &= G^\dagger \end{aligned} \quad (24)$$

Je zrejmé že násobok $G^\dagger$ a G dá vždy jednotkovú maticu čo nám dáva využiteľnosť pseudoinverzie v prípade neštvorcovej matice.

$$x_0 = G^\dagger \cdot h \quad (25)$$

Dosadením rovnice 21 do rovnice 20 získame nasledovný tvar zadaného problému:

$$\begin{aligned} \text{minimize} \quad & c^T(F \cdot z + x_0) + d \\ \text{subject to} \quad & A(F \cdot z + x_0) \leq b \end{aligned} \quad (26)$$

Obmedzenie v tvare rovnosti samozrejme zmizlo nakoľko sa pohybujeme už len vo vymedzenom podpriestore rovnosti pre ktoré platia len zvyšné obmedzenia v tvare nerovnosti, obmedzenie v tvare rovnosti je funkčne dosadené v zmysle definície pohybu v podpriestore. Po úprave :

$$\begin{aligned} \text{minimize} \quad & c^T F \cdot z + c^T x_0 + d \\ \text{subject to} \quad & A F \cdot z + A x_0 \leq b \end{aligned} \quad (27)$$

Pre zjednodušenie si môžeme vyjadriť matice štandardného tvaru:

$$\tilde{C}^T = [c^T F, -c^T x_0] \quad (28)$$

$$\tilde{A} = [A F]$$

$$\tilde{B} = [b - A x_0]$$

Štandardný tvar nadobudne zápis

$$\begin{aligned} \text{minimize} \quad & \tilde{C}^T z + s \\ \text{subject to} \quad & \tilde{A} z \leq \tilde{B} \end{aligned} \quad (29)$$

Vznikol nový problém v ktorom sa stáva optimalizovanou premennou vzdialenosť z v podpriestore od definovanej hranice vzhľadom na obmedzenia. Samotné riešenie tohto problému nám nedáva riešenie počiatočného problému, udáva len vzdialenosť z , pre určenie skutočnej hodnoty pôvodnej

optimalizovanej premennej je potrebné vykonať spätnú transformáciu na základe rovnice (21) pri podmienkach optimálneho z.

$$x^{\text{opt}} = F \cdot z^{\text{opt}} + x_0 \quad (30)$$

V Matlabe môžeme uvedený algoritmus overiť presne podľa uvedeného postupu, nasledujúci kód najprv vypočíta hodnotu optima pôvodného problému, následne problém transformuje, vypočíta optimum a výsledky porovná graficky (Obr. 14) program opäť pracuje s náhodnými hodnotami, napriek tomu sa pre každé riešiteľné zadanie výsledky rovnajú. Grafickému riešeniu zodpovedá graf polytopu riešení, s definície redukcie dimenzie musí vyplývať že riešenie sa nachádza na polorovine obmedzenia v tvare rovnosti a optimálna hodnota sa v závislosti od tvaru účelovej funkcie vyskytne na niektorom prieniku obmedzenia v tvare rovnosti a obmedzení v tvare rovnosti, teda niekde na okraji polytopu, uvedený kódy ho zobrazuje ako bod, pokiaľ sa body obidvoch riešení prekrývajú obidva problémy poskytujú rovnaké riešenie.

```
p = polytope(randn(10, 2));
plot(p);
[A, B] = double(p);
Aeq = [1 1]; Beq = 0.5;
x1 = -2:1:2;
x2 = Beq - x1;
hold on
plot(x1, x2);

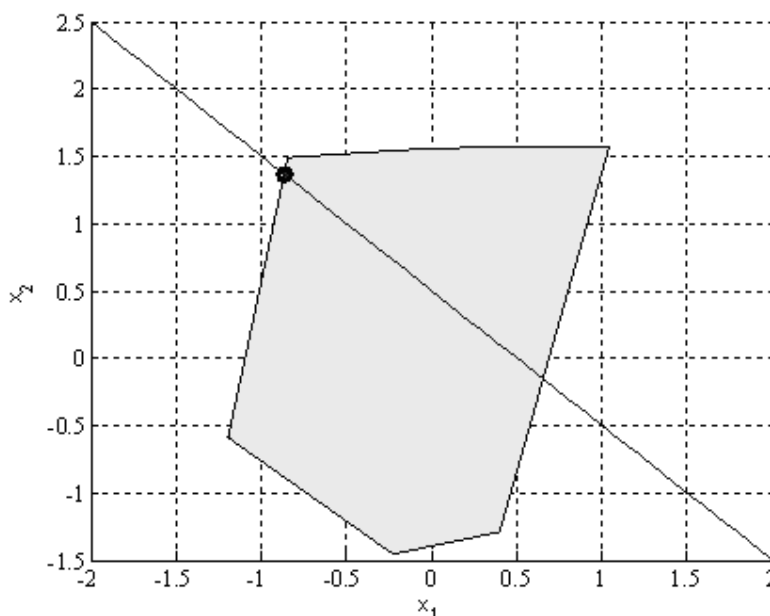
x = sdpvar(2, 1);
C = [ A*x <= B; Aeq*x == Beq ];
obj = x(1);
solvesdp(C, obj);
double(obj), double(x)

x0 = pinv(Aeq)*Beq;
F = null(Aeq);

z = sdpvar(1, 1);
x = F*z + x0;

C = [ A*(F*z + x0) <= B ];
obj = x(1);
solvesdp(C, obj)

double(obj), double(z), xopt = F*double(z)+x0
plot(xopt(1), xopt(2), 'ko', 'LineWidth', 3);
```



Obr.14

1.6. Nutričný problém

Použitelnosť konvexnej optimalizácie si pre názornosť a jednoduchosť použitia v bežnom živote predvedieme na najbežnejšom príklade nutričného problému. Nutričný problém má za úlohu pri obmedzeniach množstva živín v jedle určiť počet jednotiek rôznych potravín potrebných na prípravu takého to jedla za podmienky minimalizácie ceny za potraviny.

Majme konkrétne zadanie [4]. Je potrebné pripraviť jedlo ktoré obsahuje 900 až 1500 kalórií, minimálne 4 gramy železa, nie viac ako 50 gramov tuku, minimálne 26 gramov proteínov a menej než 50 gramov uhl'ohydrátov. Druhy jedál a ich nutričné hodnoty a cenu obsahuje obrázok 15.

Tabuľka nutričných hodnôt potravín						
	Kalórie	Železo	Tuk	Proteíny	Uhl'ohydráty	Cena
Mlieko	295	0,2	16	16	22	0,6
Základné jedlo	1216	0,2	96	81	0	2,35
Kura	394	4,3	9	74	0	1,15
Fazuľky	358	3,2	0,5	83	0	2,25
Ryba	128	3,2	0,8	7	28	0,58
Špenát	118	14,1	1,4	14	19	1,17
Zemiaky	279	2,2	0,5	8	63	0,33

Obr.15[4]

Na základe týchto údajov môžeme skonštruovať maticu obmedzení v tvare nerovností, vektor optimalizovaných premenných x bude v nutričnom probléme reprezentovať počet jednotlivých druhov jedla potrebných na zhotovenie jedla. Pojednajme o zhotovení matíc pre vytvorenie

štandardného tvaru lineárneho optimalizačného konvexného problému. Použijeme tvar :

$$\begin{aligned} &\text{minimize} \quad c^T x \\ &\text{subject to} \quad Ax \leq b \end{aligned} \tag{31}$$

Vektor neznámych hodnôt x tvoria počty jednotlivých jedál, označme si ich rešpektujúc poradie v tabuľke nutričných hodnôt poradí premennými x_1 až x_7 , vznikne stĺpcový vektor x . Vznik matice A je zložitejší musíme pri ňom vhodne použiť zadanie aby po vynásobení vektorom x vznikli v riadkoch obmedzení nerovnosti skutočných hodnôt, zjednodušene aby v riadku pre konkrétnu živinu napríklad kalórie boli len násobky množstva druhu jedla a množstva železa., každú surovinu možno obmedzeniami vyjadriť nasledovne v intervaloch ako napríklad kalórie:

$$\text{Kalórie}^{\min} \leq a_{11}x_1 + a_{22}x_2 + a_{33}x_3 + a_{44}x_4 + a_{55}x_5 + a_{66}x_6 + a_{77}x_7 \leq \text{Kalórie}^{\max}$$

Indexy max a min symbolizujú maximálne a minimálne hranice pre danú živinu, nie obidve hranice sú pre každú veličiny definované, pre nedefinované hranice platí hranica plus a mínus nekonečna, s reálnych dôvodov je možné získať len kladný počet jedál preto zahrnieme túto podmienku do ohraničení tiež a získame maticu A v ktorej každý riadok reprezentuje jedno obmedzenie, pre kalórie sú tam dva riadky berúc do úvahy horné a dolné obmedzenie, na konci je pridaná jednotková matica pre zohľadnenie kladnosti riešenia, všetky riadky boli matematickou operáciou upravené do podoby konvencie ľavej strany menšej alebo rovnnej ako hranica obmedzenia, preto sú niektoré riadky záporné, výsledná matica A :

$$A = 10^3 \begin{bmatrix} -0.2950 & -1.2160 & -0.3940 & -0.3580 & -0.1280 & -0.1180 & -0.2790 \\ 0.2950 & 1.2160 & 0.3940 & 0.3580 & 0.1280 & 0.1180 & 0.2790 \\ -0.0002 & -0.0002 & -0.0043 & -0.0032 & -0.0032 & -0.0141 & -0.0022 \\ 0.0160 & 0.0960 & 0.0090 & 0.0005 & 0.0008 & 0.0014 & 0.0005 \\ -0.0160 & -0.0810 & -0.0740 & -0.0830 & -0.0070 & -0.0140 & -0.0080 \\ 0.0220 & 0.0002 & 0 & 0 & 0.0280 & 0.0190 & 0.0630 \\ 0.0006 & 0.0025 & 0.0017 & 0.0022 & 0.0006 & 0.0012 & 0.0003 \\ -0.0010 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -0.0010 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.0010 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.0010 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.0010 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.0010 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.0010 \end{bmatrix}$$

Po vynásobení tejto matice vektorom x je zabezpečené že suma v každom riadku by bola rovná úplnému množstvu živiny v jedle. Maticu obmedzení b vyrobíme veľmi jednoducho na základe definovaných obmedzení a teda :

$$B = \begin{bmatrix} -900 \\ 1500 \\ -4 \\ 50 \\ -26 \\ 50 \\ \text{Pen_max} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Siedmi riadok reprezentuje obmedzenie pre peniaze, formálne toto obmedzenie neexistuje a je rovné mínus nekonečnu pretože nemáme hornú hranicu peňazí, pre konkrétny výpočet ju môžeme nahradiť náhodnou hodnotou dostatočne vysokou nato aby sa zabezpečilo že sa v jej hranici bude riešenie nachádzať. Maticu(vektor) c určíme na základe požadovanej minimalizácie, je to teda jedna hodnota ktorú je potrebné minimalizovať, a tou sú peniaze, v matici A ju reprezentuje riadok sedem, budeme teda minimalizovať práve tento riadok, vektor c :

$$c = \begin{bmatrix} 0.6000 \\ 2.4500 \\ 1.6700 \\ 2.2500 \\ 0.5800 \\ 1.1700 \\ 0.3300 \end{bmatrix}$$

Vybrali sme riadok, ten sa musí rovnať transponovanému vektoru c, preto je c stĺpcový. Určili sme kompletne zadanie problému. V zmysle poznatku že určenie problému je často ťažšie ako jeho vyriešenie máme ťažšiu časť za sebou, nasleduje vyriešenie ktoré za nás vykoná program. Nasleduje riešenie vytvorené v kóde Matlabu , kód zadá matice problému a problém vyrieši.

% Matica A reprezentuje druhy jedal a jednotlivé množstvá výživných hodnôt
% k nim prisluchajúcich, znamienka reprezentujú konvenciu pozavedení
% nerovnosti

```
x=sdpvar(7,1);
A=[-295 295 -0.2 16 -16 22 0.6 ;
    -1216 1216 -0.2 96 -81 0.2 2.45;
    -394 394 -4.3 9 -74 0 1.67;
```

```
-358 358 -3.2 0.5 -83 0 2.25;  
-128 128 -3.2 0.8 -7 28 0.58;  
-118 118 -14.1 1.4 -14 19 1.17;  
-279 279 -2.2 0.5 -8 63 0.33];
```

```
A = A';  
c = A(end, :)' ; d = 0;  
A = [A; -eye(7)];
```

```
% obmedzenia
```

```
Pen_max = 10;  
B=[-900;  
    1500;  
    -4;  
    50;  
    -26;  
    50  
    Pen_max; zeros(7, 1)];
```

```
F = [ A*x <= B ];  
obj = c'*x + d;
```

```
solvesdp(F, obj)
```

```
xopt = double(x)  
objopt = double(obj)
```

Problém bol úspešne riešiteľný a výsledná matica x_{opt} vyzerá:

$$x_{opt} = \begin{bmatrix} 0 \\ 0.4990 \\ 0.1743 \\ 0 \\ 0 \\ 0.1048 \\ 0.7605 \end{bmatrix}$$

Pri hodnote minimalizovanej premennej (ceny alebo počtu peňazí):

```
objopt = [1.8872]
```

Čo znamená že pri optimálna zloženie jedla sa skladá s násobkov počtu druhov jedál uvedených vo vektore x_{opt} pri sume peňazí 1.8872. Neceločíselné násobky znamenajú že problém riešime v obore reálnych čísel, úpravami v programe by sme mohli dosiahnuť celočíselné násobky avšak predmetom tohto programu bolo len ilustrovať riešenie jednoduchých každodenných problémov pomocou utvorenia problému lineárnej konvexnej optimalizácie a jej následného vyriešenia.

2. Prediktívne riadenie

2.1. Význam prediktívneho riadenia

2.1.1. Vývoj prediktívneho riadenia

Prediktívne riadenie sa v dnešnej dobe radí medzi najvýznamnejšie a najviac používané druhy riadenia. Prudký vývoj tejto metódy riadenia bol silne ovplyvnený najmä priemyslom posledných desaťročí kedy zaznamenal prudký prechod od teoretickej úrovne k praktickým aplikáciám reálnych situácií. V súčasnosti sa algoritmy prediktívneho riadenia uplatňujú v širokých oblastiach priemyselnej činnosti ako sú chémia, metalurgia, energetika, robotika [2] a mnoho iných. V počiatočných fázach implementácie sa používalo prediktívne riadenie len na riadenie pomalých procesov, kvôli spôsobu práce prediktívneho riadenia ktoré má veľké výpočtové nároky. Raketový vývoj informačných technológií však v posledných rokoch otvoril prediktívnemu riadeniu dvere takmer do všetkých oblastí v ktorých sa riadenie uplatňuje. Za jednu z najväčších výhod, z dôvodu ktorej sa prediktívne riadenie vôbec začalo používať, sa považuje schopnosť aktívne pracovať s obmedzeniami, základné typy riadenia ako spätnoväzbové nedokáže funkčne pracovať s obmedzeniami praktických ohraničení, dokáže ich len vnímať ako obmedzenia na vstup alebo výstup procesu. Prediktívne riadenie dokáže počas celého behu funkčne pracovať s obmedzeniami, čo vedie k dokonalejšiemu systému riadenia ako pri spätnoväzbovom. Dôvody prečo sa aj napriek očividnej výhode prediktívneho riadenia v niektorých procesoch stále používa spätnoväzbové je zrejmý, sú to finančné náklady na zriadenie a funkčnú prevádzku ktoré sú aj napriek viditeľnému zlepšeniu kvality prevádzky procesu stále väčšie ako náklady na využívanie a prevádzkovanie spätnoväzbového riadenia.

2.1.2. Základné princípy

Prediktívne riadenie nie je exaktne definovaný postup alebo metóda ktorá sa uplatňuje pri tomto type riadenia, skôr sa jedná o vcelku rozsiahly súbor metód a algoritmov ktoré sa implementujú podľa konkrétnej situácie. Spájajú ich ale rovnaké pojmy a celkový spôsob integrácie do riadenia procesu.

Prvou s nich je matematický model systému [2], pre efektívne zavedenie prediktívneho riadenia je potrebné mať čo najpresnejší model riadeného systému, keď je známy presný model systému dá sa presne predpovedať ako systém zareaguje na daný vstup predpokladaným výstupom, čo nadväzuje na druhý dôležitý pojem ktorým je horizont predikcie. Prediktívne riadenie predpovie stav systému vzhľadom na očakávaný vstup a smer ktorým sa chceme uberať, táto predikcia má určité časové obmedzenie veľkosti predpovede do budúcnosti, toto obmedzenie sa nazýva horizont predikcie. Pre reálne využitie sa používa pohyblivý horizont predikcie, to znamená že s celého horizontu predikcií sa zoberie len prvý člen postupnosti krokov a ten sa aplikuje, po jeho aplikácii

sa na základe uvedenej zmeny určí nový horizont predikcií s ktorého sa opäť použije len prvá hodnota, to sa ďalej opakuje do dosiahnutie požadovanej úrovne riadenia s čím súvisí ďalší dôležitý pojem ktorým je optimalizácia, prediktívne riadenie dokáže funkčne pracovať s obmedzeniami, optimalizácia na určenie horizontu predikcií je neoddeliteľnou zložkou prediktívneho riadenia a stáva sa jeho základnou súčasťou, v každom kroku ovplyvnenia procesu prediktívnym riadením sa optimalizuje výstup vzhľadom na vstup prípadne sa optimalizuje vstup vzhľadom na žiadaný výstup a dynamické vlastnosti systému (matematický model), pre efektívne riadenie je potrebné určiť v každom bode riadenia presný zásah pre najlepšie ovplyvnenie systému vzhľadom na obmedzenia a sledovanú výstupnú veličinu.

2.1.3. Analógie prediktívneho riadenia

Samotný význam prediktívneho riadenia vyplýva už s jeho názvu, takýto prístup k riadeniu má za cieľ predpovedať na základe presného matematického modelu systému a známeho vstupu jeho presný výstup, pričom v každom kroku akčného zásahu pozná budúci stav po tomto zásahu ktorý získa priebežnou optimalizáciou vstupných a výstupných obmedzení vzhľadom na dané obmedzenia. Prediktívne riadenie sa neriadi zákonom riadenia a nemá žiadny pevne stanovený a platný spôsob vykonávania, čím sa líši od klasického spätnoväzbového riadenia[2]. Analógiu medzi týmto rozdielom môžeme nájsť pri mnohých každodenných činnostiach. Predstavme si ako proces zalievania záhrady vstupom je množstvo vody ktoré záhrade dodáme a výstup kvalita pestovanej zelene, klasické spätnoväzbové riadenie by konalo len v zmysle pokiaľ je pôda suchá zalievajme ju pokým rastliny nezačnú kvitnúť, pri takomto postupe by sme okrem vstupu porúch ako dážď nemuseli vykompenzovať ani dopravné oneskorenie ktoré by pri takomto systéme vzniklo a nechceným výsledkom by mohol byť úhyn rastlín v dôsledku priveľkého zalievania. Prediktívne riadenie by tento proces riadilo nasledovne, presne pozná akým spôsobom rastliny rastú a koľko vody potrebujú, ďalej by sme pred polievaním zvážili či bude pršať alebo bude svietiť slnko a koľko vody budeme potrebovať na dosiahnutie najlepšieho rozkvetu záhrady, toto množstvo by sme vždy menili podľa potreby. Je poznať že človek vo svojej prirodzenosti prediktívne riadenie zaviedol pre bežné činnosti už veľmi dávno a jeho technologické zavedenie je len ďalší stupeň. S uvedenej analógie vyplývajú dva dôležité poznatky, prediktívne riadenie významným spôsobom redukuje vplyv dopravného oneskorenia, druhým dôležitým zistením je zásadné zlepšenie kvality výstupnej veličiny a celkové zefektívnenie. Prediktívne riadenie oproti spätnoväzbovému funkčne pracuje s obmedzeniami ako sú počasie, množstvo dodanej vody, vlhkosť pôdy, čas zalievania , spolu s priebežnou optimalizáciou zdrojov ako je dodávanie vody, náklady na dodávanie vody a rôzne iné riadi tento systém oveľa vhodnejšie než klasické spätnoväzbové riadenie. Zo zložitou systému rastú nároky na jeho ovládanie, tu nachádza uplatnenie prediktívne riadenie.

2.2. Vlastnosti prediktívneho riadenia

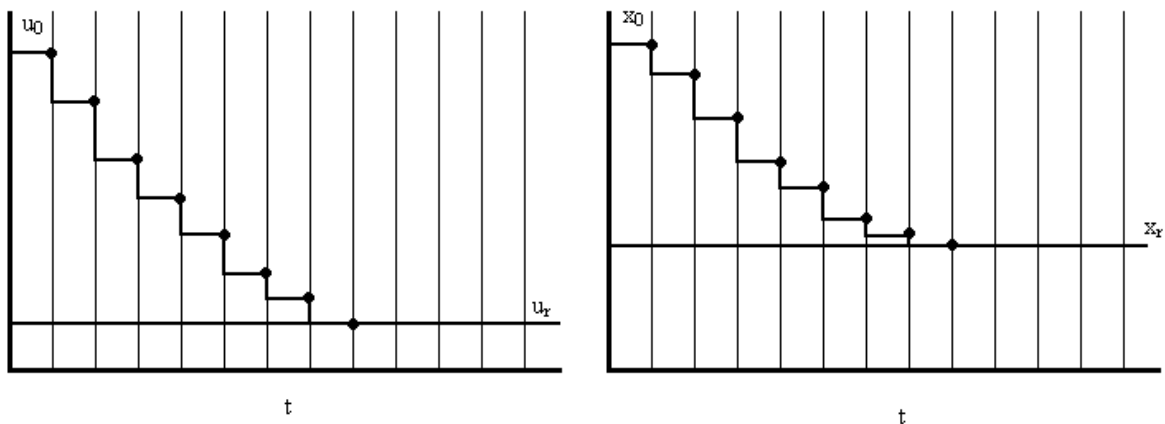
Vlastnosti prediktívneho riadenia v sebe spájajú integráciu jeho základných prvkov a určenie prepojenia týchto prvkov a problémy ktoré sa s takouto integráciou môžu spájať, identifikáciou ich môžeme dostatočne presne odstrániť alebo zohľadniť.

2.2.1. Horizont predikcie a model systému

Horizont predikcie je výsledkom jednej optimalizácie v jednom kroku alebo akčného zásahu v jednom intervale periódy vzorkovania.

Pri prediktívnom riadení sa stretávame s pojmom perióda vzorkovania, ako vieme čas a reálne čísla sú nekonečne deliteľné, reálne čísla sa pre praktické situácie zaokrúhľujú s určitou presnosťou, niečo podobné sa udeje aj s časom v ktorom sa pohybujeme, definujeme si časový úsek na základe známej jednotky, napríklad sekundy, takýto časový úsek je pre nás jednotkový a znamená minimálny krok v ktorom môžeme realizovať akčný zásah, došlo teda k rozdeleniu spojitaj oblasti na rozdelené časti, takáto rozdelená oblasť je po častiach spojená pričom sa perióda vzorkovania volí dostatočne hustá aby sa rozdiely medzi jednotlivými úsekmi mohli zanedbať, zo spojitaj oblasti sa stala po častiach spojitá a môžeme ju nazvať diskretná oblasť. Pre reálne použitie vo výpočtovej technológii musí byť každá používaná hodnota či už času alebo vstupného signálu diskretná.

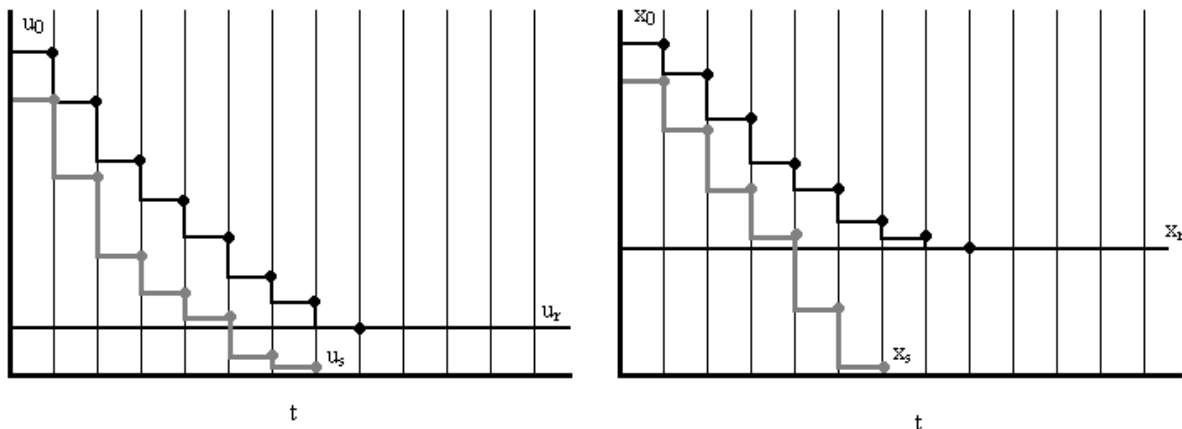
Horizont predikcie je postupnosť akčných zásahov ktorá by sa vykonala pre optimálny postup do cieľa horizontu udalostí. Pre úplne ideálny systém v ktorom poznáme úplne presný model systému a nepôsobia v ňom poruchy by stačilo určiť jeden horizont a ten by bol najlepší a stačil by pre riadenie, takýto spôsob riadenia pomocou akčných zásahov s predikciou môžeme znázorniť na obrázku 16. Horizont predikcií bol určený a pohybom po ňom, respektíve pridávaním vypočítaných



Obr.16

vstupov dosiahneme presne očakávanú a ideálnu odozvu systému, pridávaním vstupov od u_0 po referenčnú hladinu u_r následnou ideálnou reakciou očakávaného stavu systému ho možno dostať od

počiatočnej hladiny x_0 po referenčnú cieľovú hladinu x_r . Takéto ideálne podmienky samozrejme neexistujú, zanedbávajú sa pri nich dve základné podmienky, prvou s nich je vplyv porúch. Pokiaľ sa v strede procesu prudko zmení vstup systém by ďalej postupoval s uvedenými veľkosťami vstupov nerešpektujúc skutočný stav systému čo by viedlo k buď k neočakávanému výstupu alebo havárii. Druhým nezanedbateľnou podmienkou je fakt že je nemožné získať presný model systému, žiadny model nie je dost' presný a každý vykazuje ,aj keď minimálnu, odchýlku od ideálneho správania. Vplyv takéhoto správania na systém môžeme interpretovať na obrázku 17, vidíme že



Obr17

skutočný stav sa začína s postupnosťou času čoraz viac líšiť od predpokladaného stavu, v prípade poruchy na vstupe pre diagram vstupov, či už porucha vstupov alebo nepresnosť modelu obidve sa výrazne podpišu na zmene stavu systému v prípade výskytu obidvoch sa ich účinok znásobuje. Skutočný stav systému sa dramaticky mení a v závislosti od času sa stále viac vzdďľuje od predpovedaného stavu až do stavu skutočného x_s . Tento vplyv nepresnosti modelu a porúch na systém a následná odchýlka sa nazýva efekt otvďrajúcich sa nožníc, názov vychádza s grafickej interpretácie.

Takéto ideálne riadenie nie je v súlade s prediktívnym riadením, ukážka riadenia bola ukázaná na nemennom určenom horizonte udalostí bez ohľadu na neočakávané zmeny v systéme. Interpretovať by sme to mohli ako jazdu autom kde sa pozrieme od štartu do cieľu a celú cestu odriadieme úplne naslepo len na základe znalosti ovládania auta a tvaru vozovky, nerešpektovali by sme pri tom ostatné autá na vozovke a nepresné ovládania auta, takýto štýl by veľmi rýchlo viedol k havárii. Správne prediktívne riadenie pri každom akčnom zásahu najprv zistí skutočný stav systému (čím sa čiastočne mení na spätnoväzbové riadenie) a vypočíta nasledovné akčné zásahy ktoré zasiahnu systém skutočne, takto síce efekt otvďrajúcich sa nožníc nezmizne ale významne sa takto redukuje, prirovnať to možno opďť k riadeniu auta pričom sa neustále pozeráme dopredu a na základe predpokladanej okamžitej trasy určíme dďľšie kroky. Dôležitým aspektom horizontu

predikcie je aj jeho veľkosť. Ak je priveľký síce dobre vníma celú dynamiku systému a je optimálny ale zanedbáva náhodný vplyv poruchových veličín a prípadné zmeny dynamiky systému preto nie je celkom vhodný, môžeme si tento prípad predstaviť ako cestu s cez mesto, naplánuje sa ideálna trasa prechodu ktorá sa vyhodnocuje priebežne, v prípade výskytu zastavení v meste kvôli nákupu alebo pohonných hmôt sa trasa významne zmení, to aj v prípade poruchy auta alebo zhoršení jeho funkčnosti, tak sa s dlhého predikčného horizontu ktorý bol optimálny stane po častiach nie práve optimálny postup. Ak je predikčný horizont príkrátky nedokáže prakticky nijak zohľadniť zmenu dynamiky systému alebo akúkoľvek prudkú zmenu spôsobenú poruchou, predstaviť si to môžeme tak že sa rútime autom po ceste rýchlosťou dvestošesťdesiatosem kilometrov za hodinu ale pozeráme sa len sedem metrov dopredu, v prípade výskytu priepasti stihneme len veľmi ťažko auto ubrzdiť. Voľba veľkosti predikčného horizontu sa stáva dôležitou zložkou prediktívneho riadenia a konkrétna voľba závisí len od konkrétneho systému väčšinou sa volí odhadom.

Pojednajme viac o určovaní budúceho stavu systému, ten sa určí na základe presného modelu a známeho vstupu do neho, pomocou jednoduchých matíc stavového opisu systému, tieto matice vznikajú na základe matematicko-fyzikálneho vyjadrenia systému.

$$x_1 = Ax_0 + Bu_0 \quad (32)$$

Zjednodušene možno povedať že ďalší stav systému x_1 sa určí ako stav počiatočný x_0 plus príspevok vstupnej veličiny u_0 ktorý sa do systému dostal. Matice A a B reprezentujú fyzikálnu väzbu medzi systémom a vstupom vyjadrujúcu správanie sa v reálnom svete. Takto môžeme získať hodnoty stavov pričom prediktívne riadenie určuje akčným zásahom veľkosť vstupov

2.2.2. Obmedzenia

Prednostná výhoda prediktívneho riadenia spočíva v rešpektovaní všetkých obmedzení a funkčného narábania s nimi. Obmedzenia ako také sú neoddeliteľnou súčasťou každej prevádzky a môžeme ich rozdeliť do viacerých skupín[2]:

- Fyzikálne obmedzenia

Sú povinnou súčasťou systému, obmedzujú hlavne vstupy a výstupy zo systému ktoré sa môžu meniť len v daných intervaloch určenou rýchlosťou (čerpadlo nemôže pracovať výkonnejšie, ventil nemožno otvoriť viac, termodynamické dogmy)

- Technologické obmedzenia

Vytvárajú ich riadené výstupné a vstupné veličiny, sú zamerané na dodržiavanie podmienok pre dosiahnutie kvality (udržiavanie teplotných intervalov, správny prietok)

- Bezpečnostné obmedzenia

Majú za úlohu preventívne odstraňovať havárie, všetky pomocné ale aj hlavné veličiny nesmú s bezpečnostných dôvodov prekročiť kritické limity (tlak v reaktore, napätie v transformátore)

- Stabilizujúce obmedzenia

Špecifické opatrenia ktoré garantujú stabilitu riadenia, ich úlohou je hlavne zaručiť zlučiteľnosť obmedzení

Zlučiteľnosť obmedzení je základnou podmienkou stability riadenia, pokiaľ sú všetky obmedzenia zlučiteľné vymedzia oblasť prípustných riešení v ktorej je aj stabilný výstup zo systému. Všetky druhy menovaných obmedzení možno vyjadriť či už lineárne alebo nie v tvare rovníc alebo nerovníc. Obmedzenia a vznik ohraničenej oblasti riešení bol opísaný aj v časti 1.4.3 a viac ho už rozvádzať nebudeme. Dôležitejšou časťou je implementácia takýchto obmedzení do reálneho systému. Majme rovnicu (32) ktorá reprezentuje stavový opis systému

$$x_1 = Ax_0 + Bu_0 \quad (33)$$

predstavme konečný predikčný horizont s konečnou hranicou x_r systém obmedzení v tvare rovníc je potom rovný počtu rovníc ktorý vytvorí predikčný horizont o určitom počte krokov.

$$x_1 = Ax_0 + Bu_0 \quad (34)$$

$$x_2 = Ax_1 + Bu_1$$

$$x_3 = Ax_2 + Bu_2$$

A tak ďalej až po

$$x_r = Ax_{r-1} + Bu_{r-1} \quad (35)$$

tým by sme vymedzili všetky fyzikálne obmedzenia pre daný systém, nasledujú technologické obmedzenia ktoré obmedzujú vstupy a výstupy, všeobecne ich môžeme zapísať

$$u^{\min} \leq u_n \leq u^{\max} \quad (36)$$

$$x^{\min} \leq x_n \leq x^{\max} \quad (37)$$

pre ľubovoľný vstup a výstup platia horné a dolné obmedzenia veľkosti, obmedzenia v tvare horných a dolných hraníc sa dajú napísať aj konkrétne pre každý vstup a výstup, pri známej veľkosti predikčného horizontu nám takto vznikne určený počet obmedzení v tvare nerovnosti. Bezpečnostné opatrenia vznikajú podľa pravidiel prevádzky a zahrňujú v sebe pomocné ale aj ostatné veličiny. V našom prípade ich vynecháme. Stabilizujúce obmedzenia majú za následok stabilitu riešenia, väčšinou sa používajú v prípade nezlučiteľnosti obmedzení, za zmienku stojí to že v takomto prípade skôr obmedzenia odstraňujú prípadne posúvajú hranice platných obmedzení, namiesto pridávania nových obmedzení, pokiaľ nevznikne oblasť stability ohraničená obmedzeniami. S uvedeného vyplýva že obmedzenia v tvare rovnosti väčšinou zastupujú fyzikálne obmedzenia zatiaľ čo obmedzenia v tvare nerovnosti technologicko-bezpečnostné obmedzenia.

2.3. Konvexná optimalizácia a prediktívne riadenie

Neoddeliteľnou časťou prediktívneho riadenia je aj optimalizácia, určenie veľkosti vstupov na základe dynamiky systému a obmedzení ktoré poskytuje. Ak porovnáme zlučiteľnosť obmedzení prediktívneho riadenia s konvexnou oblasťou zistíme zhodu a to takú že obmedzenia ohraničujúce stabilitu procesu vytvárajú oblasť ktorá môže byť konvexná, v prípade že nie je môžeme s nej konvexnú oblasť utvoriť tak ako bolo uvedené v časti 1.4.1. Vznikol predpoklad ktorý sa zmenil na pravdu, akúkoľvek situáciu prediktívneho zadania pri známych obmedzeniach môžeme preformulovať na problém konvexnej optimalizácie v jej zmysle riešiť. Obmedzeniami máme vyjadrenú konvexnú oblasť, matice obmedzení a ich hraníc A, B, G, H sú taktiež jasne definované obmedzeniami, vektory optimalizovaných premenných tvoria všetky stavy systému a vstupné stavy x_1 až x_r a u_1 až u_r . Jediná vec ktorá chýba k úplnej premene problému prediktívneho riadenia na konvexný optimalizačný problém je účelová funkcia, teda čo budeme minimalizovať – optimalizovať ? Akú hodnotu ? Akú funkciu ? Čo je pre nás dôležitým ukazovateľom ? Samozrejme sú to hodnoty vektorovo vstupných a výstupných veličín, ale akou vlastnosťou ich spútať do kopy a vyjadriť vplyv na celý horizont predikcie ? Vyjadríme ho pomocou vzdialenosti jednotlivých stavov systému od konečného stavu systému, budeme sa snažiť aby priebeh od začiatočného do konečného stavu bol čo najjednoduchší s najmenším počtom výkyvov a vyzeral čo najplynulejšie, práve pri takomto priebehu sú vstupy najmenšie, zároveň sa minimalizuje opotrebovanie systému , nakoľko zmeny v systéme sú veľmi plynulé, všetko opäť raz smeruje k ekonomickému optimu prevádzky. V určovaní tejto vzdialenosti nám pomôžu normy.

Normy sú vo všeobecnosti vyjadrením vzdialenosti, v závislosti od druhu použitej normy môžeme stanoviť rôznu vzdialenosť systému vzhľadom na referenciu ktorou obvykle býva nulová hranica, norma sa označuje dvojitou rovnou zátvorkou napríklad $\|x_0\|_1$ je jedna norma vektoru x_0 číselne sa dajú normy vyjadriť podľa rovnice[5] :

$$\|x\|_p = \left(\sum_{i=1}^n |x_i|^p \right)^{\frac{1}{p}} \quad (38)$$

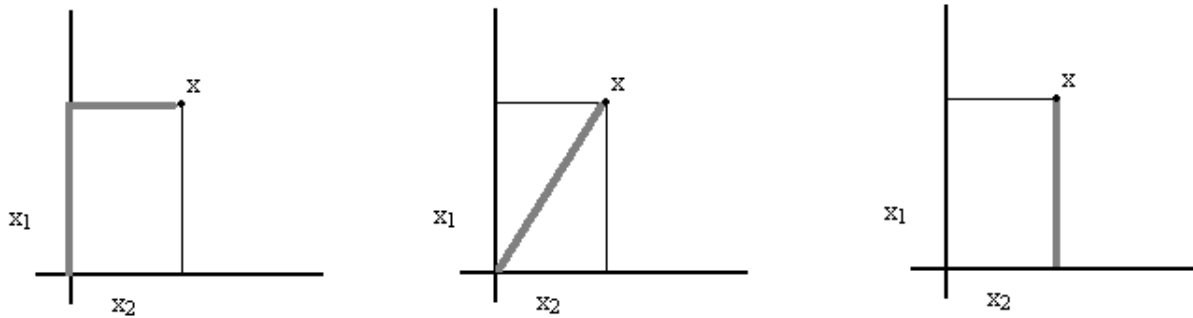
Napríklad pre dva normu platí[5]

$$\|x\|_2 = \sqrt{(|x_1|^2 + |x_2|^2 + \dots + |x_n|^2)} \quad (39)$$

Pre jedna normu po zanedbaní prvej odmocniny a umocnenia na jednu možno zjednodušene zapísať

$$\|x\|_1 = (|x_1| + |x_2| + \dots + |x_n|) \quad (40)$$

Každá s týchto noriem vyjadruje vzdialenosť jednotlivých zložiek vektora, alebo tiež súradníc, od počiatku súradnicovej osi, na obrázku 18 môžeme vidieť grafickú interpretáciu týchto noriem.



Obr.18

Obrázok naľavo zobrazuje systém výberu vzdialenosti v jedna norme, vidíme že sú najjednoduchšie vybrané len dĺžky jednotlivých zložiek vektora x . V strede je dva norma ktorá priamo vyplýva s Pytagorovej vety, jej vzdialenosť je vždy tá najkratšia možná vzdialenosť bodu a počiatku sústavy, na pravej strane môžeme vidieť nekonečno normu, zvláštnosťou tejto normy je to že jej vzdialenosť je vždy len tá najväčšia súradnica s celého vektora hodnôt. S poznaním témy noriem môžeme smelo určiť účelovú funkciu ako vzdialenosť jednotlivých stavov a vstupov od konečného stavu, najprv najzákladnejšou rovnicou pre predikčný horizont rovný jednej:

$$\|x_0 - x_r\|_1 + \|u_0 - u_r\|_1 + \|x_1 - x_r\|_1 \quad (41)$$

Vidíme že táto rovnica ukazuje vzdialenosť ktorá je prekonaná pri predikčnom horizonte jedna, v prípade že by sme chceli uvedenú vzdialenosť minimalizovať stačí minimalizovať rovnicu (41) vzhľadom na uvedené obmedzenia. Uvedme si najprv všeobecnú rovnicu pre nekonečný horizont:

$$\|(x_n - x_r)\|_{1,\infty} + \left(\sum_{k=0}^{n-1} \|(x_k - x_r)\|_{1,\infty} + \sum_{k=0}^{n-1} \|(u_k - u_r)\|_{1,\infty} \right) \quad (42)$$

Najrozšírenejšie používanie zaznamenala jedna a nekonečno norma, dva norma je tiež veľmi rozšírená ale pre prediktívne riadenie a následnú optimalizáciu sa najviac pres svoju jednoduchosť hodia práve jedna a nekonečno norma, my si vyberiem práve jedna normu a ďalej budeme používať už len tú. Zavedme ešte váhové matice P, Q a R , tieto bez rozmerné matice koeficientov pridávajú váhu jednotlivým premenným podľa uváženia a najčastejšie podľa konkrétnej situácie, určujú tým priority pri optimalizovaní dá sa povedať že sú mierkou rozdeľovania dôležitosti medzi optimalizáciou vstupov, výstupov, prejdenej vzdialenosti a vzdialenosti ktorou musí ešte proces prejsť. Dosadíme ich do rovnice (42) a dostaneme finálny tvar všeobecnej rovnice:

$$\|P(x_n - x_r)\|_{1,\infty} + \left(\sum_{k=0}^{n-1} \|Q(x_k - x_r)\|_{1,\infty} + \sum_{k=0}^{n-1} \|R(u_k - u_r)\|_{1,\infty} \right) \quad (43)$$

Rovnica (43) popisuje všeobecný tvar vyjadrujúci vzdialenosť v horizonte predikcií a je to zároveň účelová funkcia pre konvexný optimalizačný problém, lenže norma nie je korektná funkcia vhodná pre tvar účelovej funkcie, našťastie sa riešenie ponúka samo na základe poznatku jedna normy ktorá sa rovná sume absolútnych hodnôt súradníc n-rozmerného vektora. Majme jedna normu n-rozmerného vektora z , potom :

$$\|z\|_1 = |z_1| + |z_2| + \dots + |z_n| \quad (44)$$

Funkcia absolútnej hodnoty tiež nie je práve korektná pre účelovú funkciu, musíme ju preto definovať pomocou jednoduchšej transformácie s použitím pomocných premenných (slack variables):

$$\begin{array}{ll} \min |z| & \min \varepsilon \\ \min \varepsilon & \Leftrightarrow \text{s. t. } z \leq \varepsilon \\ \text{s. t. } |z| \leq \varepsilon & -z \leq \varepsilon \end{array} \quad (45)$$

Rovnice (45) fungujú v zmysle poznania funkcie absolútnej hodnoty, pokiaľ sa k bodu z ktorý sa nachádza vo vrchole funkcie absolútnej hodnoty, pokiaľ sa k tejto hodnote blížíme s dvoch strán, musia sa tieto hodnoty stretnúť v spoločnom bode z ktorý je prienikom týchto dvoch obmedzení v tvare nerovnosti. Na uvedenom prípade dobre vidieť ako vplýva počet pomocných premenných, vezmime v úvahu rovnicu (44) pri každej súradnici vektora z musíme pre zadefinovanie vektora C určiť rovnaký počet pomocných premenných:

$$\begin{array}{ll} \min \varepsilon_1 + \varepsilon_2 + \dots + \varepsilon_n & \\ \text{s. t. } z_1 \leq \varepsilon_1 & \\ -z_1 \leq \varepsilon_1 & \\ z_2 \leq \varepsilon_2 & \\ -z_2 \leq \varepsilon_2 & \\ \dots & \\ z_n \leq \varepsilon_n & \\ -z_n \leq \varepsilon_n & \end{array} \quad (46)$$

Takže pre každú súradnicu vektora v norme vznikne dvakrát toľko obmedzení v tvare nerovnosti, predstavme si že máme horizont predikcie o hodnote 100, čomu zodpovedá cca 201 noriem vektorov, takže optimalizačný program musí v priebehu jednej periódy vzorkovania vypočítať komplexný problém a určiť optimálne riešenie, a to sme ešte zanedbali obmedzenia v tvare rovnosti a nerovnosti dané dynamikou a technickou prevádzkou systému. Zadefinovaním úplne všetkých obmedzení v tvare rovnosti a nerovnosti, definovaním účelovej funkcie pre jedna normu pomocou pomocných premenných máme všetko potrebné aby sme mohli vyjadriť celý opis prediktívneho riadenia ako lineárny konvexný optimalizačný problém, konkrétne vyjadrenie by sa dalo vyjadriť na

základe vyjadrovania finálnych vektorov s čiastkovými vektorov v časti 1.5 , po takýchto úpravách by sme dostali problém v jednej zo štandardných foriem ktoré sú ľahko riešiteľné, napríklad :

$$\begin{array}{ll} \text{minimize} & c^T x + d \\ \text{subject to} & Gx \leq h \\ & Ax = b \end{array} \quad (47)$$

S uvedeného vyplýva že akýkoľvek problém prediktívneho riadenia sa dá vyjadriť ako konvexný problém a ten následne riešiť.

Záver

Úlohou tejto bakalárskej práce bolo predstaviť čitateľovi konvexnú optimalizáciu ako vetvu optimalizácie s poukázaním na zvláštnosti konvexnej optimalizácie hlavne pohyb v spojitaj konvexnej množine ktorá je vytvorená funkciami obmedzení s ktorých boli vyberaná pre jednoduchosť, rýchlosť a názornosť hlavne lineárne obmedzenia. Praktické aspekty konvexnej optimalizácie vedú k predstavám zjednodušeného hľadania optima v konvexnej ohraničenej oblasti, pričom takýto problém môžeme vytvoriť takmer s každého problému technickej alebo bežnej praxe, následne ho môžeme pomocou jednoduchých transformácií premeniť na formu v ktorej ho vieme riešiť alebo si toto riešenie zjednodušiť. Ďalšou ale rovnakou úlohou bolo jednoducho a názorne vysvetliť problematiku prediktívneho riadenia, vysvetliť prečo je výhodnejšie oproti spätnoväzbovému riadeniu, rozobrať jeho hlavné myšlienky ako pracovanie na pohyblivom horizonte predikcií spojeného s neustálou optimalizáciou vstupov a výstupov systému, opodstatnenosť ohraničení systému s názornosťou používania noriem pre vyjadrenie optimalizovaných premenných a vysvetliť hlavný vzťah medzi prediktívnym riadením a konvexnou optimalizáciou.

Konvexná optimalizácia spolu s prediktívnym riadením dnes patria k dvom najrozvíjanejším oborom v obore matematiky a obore riadenia procesov, môže nám priniesť razantné zlepšenie kvality technologických procesov čím prispeje k zníženiu nákladov a vplyvu na životné prostredie, tieto dva významné obory spolu súvisia a ich súčasný potenciál poukazuje na vysokú využiteľnosť v budúcnosti kedy sa počíta s ešte väčším rozmachom týchto metód.

Použitá Literatúra

[1] Convex Optimization - Stephen Boyd, Lieven Vandeberghe

<http://www.stanford.edu/~boyd/cvxbook/bv_cvxbook.pdf>

[2] Praktické aspekty prediktívneho riadenia – Adrián Karas, Boris Rohál' - il'kiv, Cyril Belav

[3] Prednášky k predmetu Optimalizácia – Ján Dvoran

[4] Zadanie nutričného problému bolo prevzaté zo stránky, príklad B.42

<<http://cwx.prenhall.com/bookbind/pubbooks/heizer2/chapter24/>>

[5] Euklidovská norma

<http://cs.wikipedia.org/wiki/Eukleidovsk%C3%A1_norma>