

RNDr. Margaréta Halická, CSc.
Ústav aplikovanej matematiky MFF UK

Optimálne riadenie I

Učebné texty

Bratislava, 1999

Predhovor

Nižšie uvedený text má slúžiť ako zatiaľ neúplná študijná pomôcka k predmetu Optimálne riadenie I pre študentov magisterského štúdia EFM ako aj k predmetu ... pre študentov bakalárskeho štúdia. Obsahom je diskrétna teória optimálneho riadenia. Podkladom pre text bola hlavne prvá časť nižšie uvedenej knihy profesora Brunovského. Obsah aj spôsob výkladu je však podstatne upravený a doplnený o nové časti. Teória je ilustrovaná viacerými príkladmi prevažne ekonomického charakteru. Výklad je pritom dostatočne podrobný, tak aby bol vhodný aj pre študentov bakalárskeho štúdia.

Podotýkam, že niektoré časti tohoto textu tvoria zatiaľ nedostatočne zeditované časti mojich prednášok z r. 1997 zaznamenané mojou diplomantkou E. Juklovou. V budúcnosti ma čaká úprava týchto častí, oprava prípadných chýb (aj na základe pripomienok študentov) a doplnenie textu o ďalšie časti. Na tento text nadväzuje pripravovaný učebný text Optimálne riadenie II venovaný spojitaj teórii optimálneho riadenia.

Ako doplnok obidvoch študijných textov doporučujem nasledovnú literatúru:

- P. Brunovský: Matematická teória optimálneho riadenia, ALFA, Bratislava, 1980
- P. Brunovský: Teória optimálneho riadenia (skriptá) UK, Bratislava, 1977
- M. Vlach: Optimální řízení regulovaných systémů, STTL, Praha, 1975
- P. Mandl: Základy optimalizace vícestupňových soustav, Kybernetika 8, 1972
- G. Feichtinger, R.F. Hartl : Optimale Kontrolle Ökonomische Prozesse, W.Gruyter, Berlin-New York 1986
- A.C. Chiang : Elements of Dynamic Optimization, McGraw-Hill, Inc. New York, 1996
- M.I. Kamien, N.L. Schwartz : Dynamic Optimization. The Calculus of Variations and Optimal Control in Economics and Management, Elsevier, 1995
- D. Leonard, N. Long : Optimal control theory and static optimization in economics, Cambridge University Press, 1992

autorka

1. Úvod do teórie optimálneho riadenia

Teória optimálneho riadenia (TOR) má v slovenčine niekoľko ekvivalentných pomenovaní. Niekedy sa označuje ako teória optimálnych procesov, dynamická optimalizácia alebo dokonca dynamické programovanie. V angličtine sa táto teória nazýva Optimal Control, v nemčine Optimale Steuerung, v češtine Optimální regulace, v ruštine Optimal'noje upravlenije.

Táto teória je zhruba 40 rokov stará a podnet na jej vznik dali rozvoj automatizovaného riadenia a kozmickej techniky v bývalom Sovietskom zväze. Základom teórie a to najmä jej spojitnej časti sa stal tzv. Pontrjaginov princíp maxima (Pontrjagin, ZSSR), formulujúci nutné podmienky optimality pre určité optimalizačné úlohy. Zároveň na druhom konci sveta, v USA, ako podnet na riešenie niektorých ekonomických problémov, boli Bellmanom formulované určité princípy optimality umožňujúce efektívne riešiť niektoré úlohy tzv. metódou dynamického programovania. Ako sa čoskoro ukázalo, takýmito úlohami sú aj niektoré diskkrétne úlohy optimálneho riadenia a metóda dynamického programovania má súvis s Pontrjaginovým princípom maxima.

Poznamenajme, že metódy optimálneho riadenia našli uplatnenie v mnohých odboroch, ako sú: chemické inžinierstvo, ekológia, elektrotechnika, medicína. Významné uplatnenie najmä v poslednom desaťročí nachádza v makroekonomickej teórii rastu, kde mnohé dynamické modely sú formulované ako úlohy optimálneho riadenia a príslušná teória pomáha pri kvalitatívnej analýze takýchto modelov.

Nebudeme sa teraz snažiť vymedziť predmet teórie optimálneho riadenia. Predstavu o tom, čím sa táto teória zaoberá získame v nasledujúcich úvodných častiach, kde sa abstrakciou z niekoľkých konkrétnych príkladov formuluje všeobecná schéma úlohy OR v dvoch verziách diskkrétnej a kontinuálnej. V ďalších častiach tohoto textu sa potom budeme venovať výlučne diskrétnym úlohám.

1.1. Príklady

Príklad 1. Najrýchlejšie premiestnenie

Na rovných koľajničkách (na ktorých sme zaviedli dĺžkovú súradnicu y) majme v bode $y = 0$ v pokoji vozík, ktorý chceme z tohto bodu za najkratší čas dopraviť do bodu $y = y_1 > 0$ tak, aby v tomto bode zastal. Ťažná sila môže nadobúdať hodnoty v intervale $[-\alpha, \beta]$, $\alpha, \beta > 0$ (záporná ťažná sila znamená brzdenie); odpor, ktorý prostredie kladie pohybu vozíka, zanedbávame.

Ak teraz $u(t)$ označuje ťažnú silu, ktorou v čase t pôsobíme na vozík, z Newtonovho zákona vyplýva, že dráha $y(t)$, ktorú vozík prešiel za čas t , bude riešením diferenciálnej rovnice

$$m\ddot{y} = u(t), \quad y(0) = 0, \quad \dot{y}(0) = 0, \quad (1.1)$$

kde m označuje hmotnosť vozíka. Našou úlohou je nájsť program $u(t)$ pôsobenia na vozík, ktorý vyhovuje ohraničeniam $-\alpha \leq u(t) \leq \beta$ (nazveme ho *riadením*); pritom

zodpovedajúce riešenie $y(t)$ rovnice (1), so začiatočnou podmienkou $y(0) = 0$, $\dot{y}(0) = 0$ (ktoré budeme nazývať *odozvou* riadenia u) má spĺňať rovnice

$$y(\hat{T}) = y_1, \quad \dot{y}(\hat{T}) = 0 \quad (1.2)$$

a $\hat{T}$ má byť minimálne $T \geq 0$ také, pre ktoré možno prípustným riadením dosiahnuť splnenie rovníc (1.2).

Intuitívne je jasné, aký má mať $u(t)$ tvar: najprv treba vozík maximálnou silou ťahať ($u(t) = \beta$) až dovtedy, kým ho ešte stačíme do bodu y_1 zbrzdiť, potom ho treba maximálnou silou brzdiť ($u(t) = -\alpha$). Neskôr uvidíme, že tento výsledok možno exaktne dostať z princípu maxima.

Príklad 2. Energeticky optimálna jazda vlaku

je vlastne modifikáciou príkladu 1. Namiesto vozíka si predstavme elektrickú lokomotívu, ktorú treba z miesta $y = 0$ dopraviť do miesta $y = a$ za daný čas T tak, aby sa zo siete odobralo minimálne množstvo elektrickej energie. V tomto prípade budeme brať do úvahy aj odpor prostredia, ale predpokladáme, že tento nezávisí od ubehnutej dráhy, ale iba od rýchlosti (zodpovedá to pohybu lokomotívy po ideálnej rovnej trati); predpokladáme, že spotrebovaná energia je úmerná vykonanej práci

Rovnica pohybu bude mať teraz tvar

$$m\ddot{y} = u - q(\dot{y}), \quad u \in [-\alpha, \beta] \quad (1.3)$$

kde q je odpor prostredia; našim cieľom je nájsť riadenie $u(t)$, tak aby jeho odzva $y(t)$ so začiatočnými podmienkami $y(0) = 0$, $\dot{y}(0) = 0$ spĺňala $y(T) = a$, $\dot{y}(T) = 0$ a aby práca vykonaná elektrickou energiou

$$J(u) = \int_0^a u^+ dy = \int_0^T u^+ \dot{y} dt$$

kde

$$u^+ = \begin{cases} u, & \text{ak } u \geq 0 \\ 0, & \text{ak } u < 0 \end{cases}$$

(pri brzdení energiu zo siete neodoberáme!) bola minimálna.

Môžeme pridať ešte ďalšie ohraničenie, napr. že lokomotíva nesmie prekročiť danú rýchlosť, t.j. musí platiť $y(t) \leq \gamma$.

Príklad 3. Optimálna spotreba

Predpokladajme, že jednotlivec vlastní kapitál, ktorého veľkosť v každom časovom okamžiku $t \in [0, T]$ je $K(t)$. Tento kapitál prináša vlastníkovi dôchodok o veľkosti $iK(t)$, kde i je trhová úroková miera. Vlastník sa môže v každom okamžiku $t \in [0, T]$ rozhodnúť, aké množstvo kapitálu spotrebuje. Spotreba $C(t)$ v čase t , prináša úžitok veľkosti $U(C(t))$, kde U je daná úžitková funkcia. Pretože vlastník je netrpezlivý a preferuje skoršiu spotrebu pred neskoršou, úžitok je v každom časovom okamžiku t diskontovaný na veličinu $e^{-t\delta}U(C(t))$, kde δ je mierkou jeho netrpezlivosti. Predpokladáme, že na začiatku procesu máme k dispozícii množstvo kapitálu $K_0 \geq 0$. Žiadame, aby hodnota držaného

kapitálu na konci procesu nadobúdala predpísanú hodnotu K_1 . Úlohou je maximalizovať diskontovaný úžitok zo spotreby za celé uvažované obdobie.

Správanie sa procesu spotreby možno popísať diferenciálnou rovnicou

$$\dot{K}(t) = iK(t) - C(t), \quad t \in [0, T]$$

s začiatočnou podmienkou

$$K(0) = K_0$$

Cieľom je nájsť také riadenie $C(t)$, aby príslušná odozva spĺňala cieľovú (koncovú) podmienku

$$K(T) = K_T$$

aby účelová funkcia

$$J = \int_0^T e^{-t\delta} U(C(t)) dt$$

nadobúdala maximálnu hodnotu.

Túto úlohu možno prirodzeným spôsobom formulovať aj ako diskrétnu úlohu

Príklad 4. Optimálna spotreba (diskrétny variant)

Predpokladajme, že jednotlivec vlastní kapitál, ktorého veľkosť na začiatku i -teho obdobia, je x_i . Tento kapitál prináša vlastníčkovi na konci i -teho obdobia dôchodok o veľkosti rx_i , kde r je trhovú úrokovú mieru. Vlastník sa môže na konci každého obdobia rozhodnúť, aké množstvo kapitálu spotrebuje. Spotreba u_i prináša (diskontovaný) úžitok veľkosti $(\frac{1}{1+\delta})^i \ln u_i$, kde $\delta > 0$ je mierkou netrpezlivosti pri spotrebovávaní. Predpokladáme, že na začiatku procesu máme k dispozícii množstvo kapitálu $a > 0$. Žiadame, aby hodnota držaného kapitálu na konci procesu trvajúceho k období nadobúdala predpísanú hodnotu $b \geq 0$. Úlohou je maximalizovať diskontovaný úžitok zo spotreby za k období.

V zadaní úlohy vystupujú nasledovné údaje:

x_i – kapitál na začiatku i -teho obdobia

rx_i – úrok (dôchodok) za i -te obdobie z x_i

r – úroková miera

u_i – spotreba na konci i -teho obdobia

$\ln u$ – úžitková funkcia

$(\frac{1}{1+\delta})^i \ln u_i$ – diskontovaný úžitok zo spotreby u_i v i -tom období

$\delta > 0$ miera netrpezlivosti $\frac{1}{1+\delta} = \alpha < 1$

$a > 0$ – začiatočný kapitál

b – žiadaná hodnota kapitálu na konci uvažovaného procesu

Premennými veličinami sú

$i, i \in \{0, 1, \dots, k\}$, ktorá má charakter času,

x_i možno brať za stavovú premennú, vyjadruje stav procesu (bankového konta)

u_i je vstupom do procesu, bude to riadiaca premenná, jej voľbou ovplyvňujeme, riadime správanie procesu

Správanie sa procesu pri prechode z jedného obdobia do druhého opisuje nasledovná diferenčná rovnica

$$x_{i+1} = x_i + rx_i - u_i, \quad i = 0, \dots, k-1$$

s začiatočnou podmienkou

$$x_0 = a.$$

Cieľom je nájsť také hodnoty riadenia u_i , $i = 0, \dots, k-1$, aby príslušné riešenie diferenčnej rovnice a začiatočnej podmienky (t.j. odozva) spĺňalo koncovú podmienku

$$x_k = b$$

a aby hodnota účelovej funkcie

$$J = \sum_{i=0}^{k-1} \left(\frac{1}{1+\delta} \right)^i \ln u_i$$

bola maximálna.

Príklad 5. Optimálne rozdeľovanie zdrojov

Predpokladajme, že na začiatku i -teho obdobia ($i=0, \dots, k-1$) máme k dispozícii x_i finančných prostriedkov, ktoré treba rozdeliť medzi dva možné spôsoby použitia. Toto rozdelenie možno charakterizovať číslom $u_i \in [0, 1]$ udávajúcim pomer prostriedkov pridelených na použitie prvým spôsobom k celkovému množstvu prostriedkov. Z daného množstva x_i je teda na prvý spôsob použitia pridelených $u_i x_i$ a na druhý $(1-u_i)x_i$. Z množstva $u_i x_i$ použitého prvým spôsobom zostane na konci i -teho obdobia $0.7u_i x_i$ prostriedkov, ale prinesú zisk vo výške $2u_i x_i$. Z množstva $(1-u_i)x_i$ použitého druhým spôsobom zostane na konci i -teho $0.4(1-u_i)x_i$, pričom obdržíme zisk $2.5(1-u_i)x_i$. Celkove teda na rozdelenie v $(i+1)$ období bude k dispozícii $0.7u_i x_i + 0.4(1-u_i)x_i$ finančných prostriedkov a celkový zisk za i -te obdobie bude $2u_i x_i + 2.5(1-u_i)x_i$. Predpokladajme, že na začiatku 0-tého obdobia máme dané $x_0 = a > 0$ finančných prostriedkov. Cieľom je maximalizovať zisk za k období.

Proces rozdeľovania prostriedkov možno popísať ako diskretný, deterministický systém, ktorého stav v i -tom okamihu je určený výškou prostriedkov x_i . Vstupom do tohoto systému je veličina u_i , ktorá ovplyvňuje - riadi správanie sa systému. Preto ju nazývame aj riadiacou premennou. Dynamiku systému možno popísať diferenčnou rovnicou

$$x_{i+1} = 0.7u_i x_i + 0.4(1-u_i)x_i, \quad i = 0, \dots, k-1,$$

pre ktorú máme začiatočnú podmienku

$$x_0 = a.$$

Riadiaca premenná musí spĺňať ohraničenia na riadenie

$$u_i \in [0, 1].$$

Cieľom je maximalizovať účelovú (cenovú) funkciu - zisk

$$J = \sum_{i=0}^{k-1} [2u_i x_i + 2.5(1-u_i)x_i].$$

1.2. Všeobecná schéma úloh optimálneho riadenia

Abstrakciou z príkladov 1-5 teraz sformulujeme všeobecnú schému základnej úlohy optimálneho riadenia v dvoch alternatívnych formách - diskretnej a kontinuálnej. Táto schéma zahŕňa v sebe nielen príklady 1-5, ale aj ďalšie úlohy, s ktorými sa stretneme.

Majme objekt (systém), ktorého okamžitý stav je popísaný bodom x z množiny X a ktorého okamžitý vstup je popísaný bodom u z množiny U (ak ide o kontinuálny systém, žiadame $X \subset \mathbb{R}^n$). Správanie objektu pozorujeme v čase, ktorý je v kontinuálnej verzii daný hodnotami $t \in [0, T]$, v diskretnej verzii hodnotami $i \in \{0, \dots, k\}$. Hodnoty bodov x, u spolu s množinami X, U sa tým stávajú závislými od času pričom ich správanie v čase v kontinuálnej verzii (v diskretnej verzii) opisuje rovnica:

$$\dot{x}(t) = f(t, x(t), u(t)), \quad t \in [0, T], \quad (x_{i+1} = F_i(x_i, u_i), \quad i = 0, \dots, k-1). \quad (1.4)$$

Funkciu $u(t)$, $t \in [0, T]$, (postupnosť $\{u_i\}$, $i = 0, \dots, k-1$), ktorá každému časovému okamžiku t (i) priradzuje hodnotu z danej množiny vstupov $U(t)$, $t \in [0, T]$, ($\{U_i\}$, $i = 0, \dots, k-1$) budeme nazývať riadením. Riadenie teda spĺňa nasledovnú podmienku (tzv. ohraničenia na riadenie)

$$u(t) \in U(t), \quad t \in [0, T], \quad (u_i \in \{U_i\}, \quad i = 0, \dots, k-1). \quad (1.5)$$

Ohraničenia na stavovú premennú zapíšeme v tvare

$$x(t) \in X(t), \quad t \in [0, T], \quad (x_i \in X_i, \quad i = 0, \dots, k-1). \quad (1.6)$$

Spomedzi stavových ohraňení osobitne vyčleníme ohraničenia na začiatku a konci celého procesu. Začiatočnou podmienkou nazveme

$$x(0) \in P, \quad (x_0 \in P), \quad (1.7)$$

kde $P \subset X(0)$, P je množina povolených stavov na začiatku procesu v čase $t = 0$ ($i = 0$), tzv. začiatočná množina.

Koncovou (cieľovou) podmienkou nazveme

$$x(T) \in C, \quad (x_k \in C), \quad (1.8)$$

kde $C \subset X(T)$, C je daná množina povolených cieľových stavov, stavov na konci procesu v čase $t = T$ ($i = k$).

Kvalitu procesu môžeme merať účelovou (cenovou) funkciou

$$J(x(t), u(t), T) = \int_0^T f^0(t, x(t), u(t))dt, \quad (J(\{x_i\}, \{u_i\}, k) = \sum_{i=0}^{k-1} f_i^0(x_i, u_i)), \quad (1.9)$$

kde f^0 je daná reálna funkcia.

Zvoľme riadenie $u(t)$, $(\{u_i\})$ spĺňajúce (1.5) a začiatočný stav $x(0)$, (x_0) spĺňajúci (1.7). Nech $x(t)$ $(\{x_i\})$ je riešenie rovnice (1.4) pri zvolenom riadení so zvolenou začiatočnou podmienkou. Takéto riešenie nazývame odozvou na riadenie $u(t)$, $(\{u_i\})$ a začiatočný stav $x(0)$, (x_0) . Je zrejmé, že táto odozva môže, ale nemusí spĺňať podmienky (1.6) a (1.8). V prípade že odozva spĺňa tieto podmienky hovoríme, že riadenie je prípustné. To znamená, že prípustné je riadenie, ktoré spolu so svojou odozvou spĺňajú všetky podmienky (1.4)-(1.8).

Úlohou optimálneho riadenia je spomedzi všetkých prípustných riadení $u(t)$ nájsť také v ktorom účelová funkcia (1.9) nadobúda minimálnu hodnotu. Toto riadenie nazývame optimálnym. Pritom $T(k)$ môže byť dané (úloha s pevným časom), alebo možno minimum hľadať aj vzhľadom naň (úloha s voľným časom).

Poznámky.

1. Stavom nejakého objektu nazývame takú informáciu o ňom, ktorej znalosť v čase t_0 spolu so znalosťou vstupu na intervale $[t_0, t_1]$, $t_1 \geq t_0$ plne určuje stav objektu v čase t_1 . Skutočnosť, že x má vlastnosť stavu v diskretnom modeli, je zrejmá; aby x malo vlastnosť stavu v kontinuálnom modeli, treba od funkcie f , ako aj od riadení $u(t)$ požadovať určitú regularitu, za platnosti ktorej riešenia rovnice (1.8) existujú na intervale $[0, T]$ a jednoznačne závisia od začiatočných podmienok.

2. Do všeobecnej schémy možno zahrnúť i úlohy, v ktorých sa hľadá maximum tak, že sa ako účelová funkcia vezme záporne vzatá maximalizovaná funkcia.

1.3. Základné problémy teórie optimálneho riadenia

Samozrejme, prvoradým cieľom teórie optimálneho riadenia sú metódy pre hľadanie optimálnych riadení. Čo však rozumieť pod metódou hľadania trochu spresníme.

Už najjednoduchšie príklady nám ukážu, že hľadanie uzavretých formúl pre optimálne riadenie je beznádejné. Existencia vysoko výkonných počítačov zvädza k predstave, že diskretizáciou všetkých kontinuálnych veličín možno prejsť na úlohu minimalizácie funkcie konečného počtu hodnôt, ktorú možno riešiť prebratím všetkých možností. Ukážeme si neskôr, že aj táto cesta je pri trocha zložitejších problémoch neschodná, a to pre veľký rozsah úlohy, ktorá takto vznikne.

Snahou teórie optimálneho riadenia je vytvoriť metódy, pomocou ktorých by bolo možné čo najviac zúžiť množinu riadení, ktoré prichádzajú do úvahy ako kandidáti na optimálne riadenie, a tým rozsah prípadnej diskretizovanej úlohy systematického preberania možností zúžiť na únosnú mieru. V numerických metódach ide zasa o to, nerobiť tento prieskum celkom naslepo (alebo, ako budeme hovoriť pasívne), ale cieľavedome a úsporne sa približovať k optimu.

Ďalším cieľom teórie je skúmať vlastnosti optimálnych riadení. Pod tým treba rozumieť, že chceme niečo vyvodiť o optimálnych riadeniach bez toho, že by sme ich poznali, priamo z charakteru úlohy (podobne ako v teórii stability ide o rozhodnutie o stabilite diferenciálnej rovnice bez toho, že by sme poznali jej riešenia). Je to dôležité jednak

preto, že presné optimálne riadenie zriedkakedy vieme nájsť, jednak nám takáto informácia niekedy môže radikálne obmedziť množinu kandidátov na optimálne riadenie.

2. Dynamické programovanie

2.1. Princíp optimality a optimálna spätná väzba

Uvažujme o diskrétnej úlohe optimálneho riadenia s pevným časom, ktorá je daná stavovou diferenčnou rovnicou

$$x_{i+1} = F_i(x_i, u_i), \quad i = 0, \dots, k-1,$$

ohraničeniami na stavovú a riadiacu premennú

$$x_i \in X_i, \quad u_i \in U_i$$

začiatočným bodom x_0 , cieľovou množinou C a účelovou funkciou

$$J(x_0, \mathcal{U}) = \sum_{i=0}^{k-1} f_i^0(x_i, u_i).$$

Túto úlohu v závislosti od x_0 označíme $D_0(x_0)$ a nazveme ju *úlohou optimálneho prechodu z bodu x_0 do množiny C na intervale $[0, k]$* . Riadením pre $D_0(x_0)$ rozumieme postupnosť $\mathcal{U} = \{u_0, \dots, u_{k-1}\}$, kde $u_i \in U_i$, pre každé $i = 0, \dots, k-1$. Odozvou $\mathcal{X} = \{x_0, \dots, x_k\}$ na riadenie $\mathcal{U}$ nazývame riešenie stavovej diferenčnej rovnice pre zvolené $\mathcal{U}$ a daný začiatočný stav x_0 . V prípade, že odozva $\mathcal{X} = \{x_0, \dots, x_k\}$ na riadenie $\mathcal{U}$ spĺňa všetky ohraňovania, t.j. $x_i \in X_i$, $i = 0, \dots, k-1$ a $x_k \in C$, budeme riadenie $\mathcal{U}$ nazývať *prípustným*.

Naším cieľom v tejto a nasledujúcej časti je odvodiť pre $D_0(x_0)$ nutné a postačujúce podmienky optimálnosti v tvare tzv. Bellmanovej rovnice dynamického programovania. Táto rovnica dáva metódu na riešenie úloh. Princíp spočíva v tom, že sa úloha zaradí do celého systému úloh a jej riešenie sa dostane postupným riešením jednoduchších úloh, pričom pri riešení nasledujúcej úlohy sa vždy využije riešenie predchádzajúcej. Oproti pasívnemu hľadaniu predstavuje tento postup podstatnú úsporu a vedie k pojmu optimálnej spätnej väzby.

Úlohu $D_0(x_0)$ vnoríme do systému úloh $\mathcal{D} = \{D_j(x); \quad j \in [0, k-1], \quad x \in X_j\}$, kde $D_j(x)$ je úloha optimálneho prechodu z bodu x do množiny C na intervale $[j, k]$: spomedzi všetkých riadení $\mathcal{U}_j = \{u_j, \dots, u_{k-1}\}$ takých, že $u_i \in U_i$ ($i = j, \dots, k-1$), ktorých odozva $\mathcal{X}_j = \{x_j, x_{j+1}, \dots, x_k\}$ so začiatočnou hodnotou $x_j = x$ spĺňa ohraňovania $x_i \in X_i$ ($i = j, \dots, k-1$) a $x_k \in C$, treba nájsť také, pri ktorom funkcia

$$J_j(x, \mathcal{U}_j) = \sum_{i=j}^{k-1} f_i^0(x_i, u_i)$$

nadobúda minimum.

Veta 2.1. Bellmanov princíp optimality. Ak $\hat{U} = \{\hat{u}_0, \dots, \hat{u}_j, \dots, \hat{u}_{k-1}\}$, $\hat{X} = \{\hat{x}_0, \dots, \hat{x}_j, \dots, \hat{x}_k\}$ je optimálne riadenie a jeho odozva pre úlohu $D_0(\hat{x}_0)$, potom pre každé $j \in [0, k-1]$ je $\hat{U}_j = \{\hat{u}_j, \dots, \hat{u}_{k-1}\}$ a $\hat{X}_j = \{\hat{x}_j, \dots, \hat{x}_k\}$ optimálne riadenie a optimálna odozva pre úlohu $D_j(\hat{x}_j)$.

Dôkaz. Sporom. Nech pre niektoré $j \in [0, k-1]$ riadenie $\hat{U}_j$ nie je optimálne pre úlohu $D_j(\hat{x}_j)$. Potom musí existovať prípustné riadenie $\bar{U}_j = \{\bar{u}_j, \dots, \bar{u}_{k-1}\}$ a jeho odozva $\bar{X}_j = \{\bar{x}_j, \dots, \bar{x}_k\}$ pre $D_j(\hat{x}_j)$, $\hat{x}_j = \bar{x}_j$ také, že

$$J(\hat{x}_j, \bar{U}_j) = \sum_{i=j}^{k-1} f_i^0(\bar{x}_i, \bar{u}_i) < \sum_{i=j}^{k-1} f_i^0(\hat{x}_i, \hat{u}_i) = J(\hat{x}_j, \hat{U}_j)$$

Definujme nové riadenie pre úlohu $D_0(\hat{x}_0)$ predpisom $\bar{U} = \{\hat{u}_0, \dots, \hat{u}_{j-1}, \bar{u}_j, \dots, \bar{u}_{k-1}\}$. Potom zrejme $\bar{X} = \{\hat{x}_0, \dots, \hat{x}_j, \bar{x}_{j+1}, \dots, \bar{x}_k\}$ je odozva na $\bar{U}$, riadenie $\bar{U}$ je prípustné a platí

$$J(\hat{x}_0, \bar{U}) < J(\hat{x}_0, \hat{U})$$

a teda $\hat{U}$ nie je optimálne. $\square$

Pre každé $j \in [0, k-1]$ a $x \in X_j$ definujme $\Gamma_j(x)$ ako množinu tých $u \in U_j$, že existuje optimálne riadenie $\hat{U}_j = \{\hat{u}_j, \dots\}$ pre úlohu $D_j(x)$ také, že $\hat{u}_j = u$. Poznamenajme, že pre niektoré j , x môže byť množina $\Gamma_j(x)$ prázdna.

Veta 2.2. Ak $\hat{U}$, $\hat{X}$ je optimálne riadenie a odozva pre úlohu $D_0(\hat{x}_0)$, potom platí $\hat{u}_i \in \Gamma_i(\hat{x}_i)$ pre všetky $i = 0, \dots, k-1$. Naopak, ak pre nejaké riadenie $\hat{U}$ a jeho odozvu $\hat{X}$ platí $\hat{u}_i \in \Gamma_i(\hat{x}_i)$ pre všetky $i = 0, \dots, k-1$, potom $\hat{U}$, $\hat{X}$ sú optimálne pre úlohu $D_0(\hat{x}_0)$.

Dôkaz. Prvá časť je priamym dôsledkom Vety 2.1. Druhú časť dokážeme tak, že dokážeme indukciou platnosť výroku W_j :

Ak $\hat{U}_j$ a $\hat{X}_j$ sú riadenie a jeho odozva pre úlohu $D_j(\hat{x}_j)$ spĺňajúce $\hat{u}_i \in \Gamma_i(\hat{x}_i)$ pre $i = j, \dots, k-1$, potom $\hat{U}_{j+1}$, $\hat{X}_{j+1}$ sú optimálne pre úlohu $D_j(\hat{x}_j)$.

Výrok W_{k-1} zrejme platí.

Predpokladajme, že platí W_{j+1} . Dokážeme platnosť výroku W_j .

Nech $\hat{X}_j$, $\hat{U}_j$ spĺňajú predpoklady výroku W_j , t.j., pre $\hat{X}_j$, $\hat{U}_j$ platí $\hat{u}_i \in \Gamma_i(\hat{x}_i)$, $i = j, \dots, k-1$. Z toho však vyplýva, že $\hat{U}_{j+1} = \{\hat{u}_{j+1}, \dots\}$, $\hat{X}_{j+1} = \{\hat{x}_{j+1}, \dots\}$ spĺňajú predpoklady výroku W_{j+1} . Teda podľa indukčného predpokladu $\hat{U}_{j+1}$ je optimálne riadenie pre úlohu $D_{j+1}(\hat{x}_{j+1})$.

Na druhej strane, podľa definície Γ_j , z predpokladu $\hat{u}_j \in \Gamma_j(\hat{x}_j)$ vyplýva, že existuje optimálne riadenie $\bar{U}_j = \{\hat{u}_j, \bar{u}_{j+1}, \dots\}$ pre úlohu $D_j(\hat{x}_j)$. Keďže riadenia $\hat{U}_j$ a $\bar{U}_j$ majú rovnaký prvý člen, druhý člen ich odozvy je rovnaký a rovný $\hat{x}_{j+1}$. Z princípu optimality vyplýva

$$J_{j+1}(\hat{x}_{j+1}, \hat{U}_{j+1}) = J_{j+1}(\hat{x}_{j+1}, \bar{U}_{j+1})$$

a teda aj

$$J_j(\hat{x}_j, \hat{U}_j) = J_j(\hat{x}_j, \bar{U}_j).$$

Z toho vyplýva, že aj $\hat{\mathcal{U}}_j$ je optimálne riadenie a výrok W_j je dokázaný. $\square$

Z vety vyplýva, že ak v_i sú ľubovoľné selekcie z Γ_i , t.j. ľubovoľné funkcie v_i také, že $v_i(x) \in \Gamma_i(x)$, $i = 0, \dots, k-1$, potom optimálne riadenie pre úlohu $D_0(x_0)$ môžeme generovať pomocou funkcie v_i vzťahom

$$u_i = v_i(x_i).$$

Inak povedané, optimálne odozvy sú riešeniami diferenčnej rovnice

$$x_{i+1} = F_i(x_i, v_i(x_i)), \quad i = 0, \dots, k-1.$$

Funkcii v_i hovoríme *optimálna spätná väzba* alebo *syntéza optimálneho riadenia*. Názov spätná väzba je odôvodnený tým, že funkcia v_i určuje, ako musí vstup systému reagovať na jeho okamžitý stav, aby ho udržal v optimálnom režime.

Hodnota optimálneho riadenia v určitom časovom okamihu je funkciou iba stavu systému v tomto okamihu. Inak, na určenie hodnoty optimálneho riadenia v danom časovom okamihu nie je potrebné poznať z histórie procesu nič okrem okamžitého stavu, teda ani začiatkový stav. Syntéza optimálneho riadenia je konečným cieľom riešenia úlohy optimálneho riadenia. Generovanie optimálneho riadenia pomocou spätnej väzby má totiž oproti programovému riadeniu tú veľkú výhodu, že reaguje na odchýlky procesu od optimálneho režimu. Presnejšie, ak sa systém následkom vonkajšieho vzruchu odchýli od optimálneho režimu, riadenie, generované optimálnou spätnou väzbou, bude opäť optimálne pre nový odchýlený začiatkový stav.

2.2. Rovnica dynamického programovania

Predpokladajme, že pre každé $j \in [0, k-1]$, $x \in X_j$ je splnená nasledovná podmienka: Ak existuje prípustné riadenie pre $D_j(x)$, potom existuje aj optimálne $\hat{\mathcal{U}}_j$ pre $D_j(x)$ a označíme $V_j(x) = J_j(x, \hat{\mathcal{U}}_j) = \min\{J_j(x, \mathcal{U}_j); \mathcal{U}_j \text{ prípustné pre } D_j(x)\}$. Inak povedané označíme $V_j(x)$ minimálnu hodnotu účelovej funkcie pre úlohu $D_j(x)$ (funkcia V nemusí byť pre všetky j , x definovaná). Funkciu V_j nazývame *hodnotovou funkciou* pre systém úloh $D_j = \{D_j(x); x \in X_j\}$; funkciu V nazývame *hodnotovou funkciou* pre systém úloh $\mathcal{D}$. Z definície V vyplýva

$$V_{k-1}(x) = \min_{\mathcal{U}_{k-1}} J_{k-1}(x, \mathcal{U}_{k-1}) = \min_{u_{k-1}} f_{k-1}^0(x, u_{k-1}) \quad (2.1)$$

lebo $\mathcal{U}_{k-1} = \{u_{k-1}\}$. Samozrejme minimum berieme len cez prípustné riadenia. Podobne dostaneme

$$\begin{aligned} V_j(x) &= \min_{\mathcal{U}_j} J_j(x, \mathcal{U}_j) = \min_{\mathcal{U}_j} [f_j^0(x, u_j) + \sum_{i=j+1}^{k-1} f_i^0(x_i, u_i)] = \\ \min_{\mathcal{U}_j} [f_j^0(x, u_j) + J_{j+1}(x_{j+1}, \mathcal{U}_{j+1})] &= \min_{u_j} \min_{\mathcal{U}_{j+1}} [f_j^0(x, u_j) + J_{j+1}(x_{j+1}, \mathcal{U}_{j+1})] = \end{aligned} \quad (2.2)$$

$$\begin{aligned}
&= \min_{u_j} [f_j^0(x, u_j) + \min_{\mathcal{U}_{j+1}} J_{j+1}(x_{j+1}, \mathcal{U}_{j+1})] = \\
&= \min_{u_j} [f_j^0(x, u_j) + V_{j+1}(x_{j+1})] = \min_{u_j} [f_j^0(x, u_j) + V_{j+1}(F_j(x, u_j))], \quad j = 0, \dots, k-2,
\end{aligned}$$

Minimum sa berie iba cez prípustné riadenia, resp. tie $u_j \in U_j$, pre ktoré existuje prípustné riadenie $\mathcal{U}_j$ začínajúce s u_j . Ak označíme $V_k(x) = 0$ pre $x \in C$ môžeme vzťahy (2.1) a (2.2) zhrnúť do vzťahu

$$V_j(x) = \min_{u_j \in U_j} [f_j^0(x, u_j) + V_{j+1}(F_j(x, u_j))], \quad j = 0, \dots, k-1 \quad (2.3)$$

ktorý sa nazýva *Bellmanova rovnica dynamického programovania*. Platí veta:

Veta 2.3. *Funkcia V je hodnotová funkcia a v je optimálna spätná väzba pre systém úloh $\mathcal{D}$ vtedy a len vtedy, ak sú splnené vzťahy*

$$V_j(x) = f_j^0(x, v_j(x)) + V_{j+1}(F_j(x, v_j(x))) = \min_{u \in U_j} [f_j^0(x, u) + V_{j+1}(F_j(x, u))]$$

pre $j = 0, \dots, k-1$ a pre všetky x, u také, pre ktoré sú všetky výrazy v predchádzajúcom vzťahu definované. Minimum berieme cez tie u , pre ktoré existuje prípustné $\mathcal{U}_j$ začínajúce s u . Pritom $V_k(x) = 0$ pre $x \in C$.

Dôkaz. Časť "len vtedy" tvrdenia je už dokázaná. Časť "vtedy" dokážeme indukciou. Pre $j = k-1$ veta zrejme platí:

$$V_{k-1}(x) = f_{k-1}^0(v_{k-1}(x)) + 0 = \min_{u \in U_{k-1}} f_{k-1}^0(x, u) = \min_{\mathcal{U}_{k-1}} J_{k-1}(x, u)$$

a teda $v_{k-1}(x) = u$ je optimálna spätná väzba a V_{k-1} je hodnotová funkcia. Nech veta platí pre $i = j+1, \dots, k-1$. Chceme dokázať, že platí aj pre $i = j, \dots, k-1$.

Nech $V_j, V_{j+1}, \dots, V_k$ vyhovujú rovnici dynamického programovania, $x \in X_j$. Z indukčného predpokladu vyplýva, že $V_{j+1}, \dots, V_k$ sú hodnotové funkcie pre príslušné úlohy. Treba ešte dokázať, že aj V_j je hodnotovou funkciou pre úlohy D_j . Nech V_j nie je minimálna hodnota účelovej funkcie pre $D_j(x)$. Potom existuje riadenie $\bar{\mathcal{U}}_j$ a jeho odozva $\bar{\mathcal{X}}_j$ taká, že $J_j(x, \bar{\mathcal{U}}_j) < V_j(x)$. To znamená

$$V_j(x) > J_j(x, \bar{\mathcal{U}}_j) = f_j^0(x, \bar{u}_j) + \sum_{i=j+1}^{k-1} f_i^0(\bar{x}_i, \bar{u}_i) = f_j^0(x, \bar{u}_j) + J_{j+1}(\bar{x}_{j+1}, \bar{\mathcal{U}}_{j+1}) \geq$$

$$f_j^0(x, \bar{u}_j) + V_{j+1}(\bar{x}_{j+1}) = f_j^0(x, \bar{u}_j) + V_{j+1}(F_j(x, \bar{u}_j)) \geq \min_{u_j \in U_j} [f_j^0(x, u_j) + V_{j+1}(F_j(x, u_j))]$$

kde nerovnosť pri prechode na nový riadok sme dostali využitím indukčného predpokladu. Dostali sme spor s tým, že $V_j, V_{j+1}, \dots$ spĺňali rovnicu a teda V_j je hodnotová funkcia.

Nech teraz aj funkcie $v_j, v_{j+1}, \dots, v_{k-1}$ spĺňajú RDP. Zrejme podľa indukčného predpokladu v_{j+1} je optimálnou spätnou väzbou. Teda optimálne riadenie $\hat{\mathcal{U}}_{j+1}$ pre $D_{j+1}(x_{j+1})$,

kde $x_{j+1} = F_j(x, v_j(x))$, existuje a $V_{j+1}(x_{j+1})$ je minimálna hodnota účelovej funkcie. Zostrojme riadenie $\hat{\mathcal{U}}_j = (v_j(x), \hat{\mathcal{U}}_{j+1})$. Platí

$$J_j(\hat{\mathcal{U}}_j) = f_j^0(x, v_j(x)) + \sum_{i=j+1}^{k-1} f_i^0(x_i, \hat{u}_i) = f_j^0(x, v_j(x)) + V_{j+1}(\hat{x}_{j+1}) = V_j(x).$$

Z toho vyplýva, že $\hat{\mathcal{U}}_j$ je optimálne riadenie a teda $v_j(x) \in \Gamma_j(x)$. $\square$

2.3. Riešenie úloh

Rovnica dynamického programovania dáva možnosť rekurentne počítať optimálnu spätnú väzbu a hodnotovú funkciu "od konca". Ukážeme si to na nasledujúcich príkladoch.

Príklad 1. Metódou dynamického programovania riešte UOR danú

$$\begin{aligned} x_{i+1} &= x_i + u_i, & i &= 0, 1, \\ x_0 &= 1, \\ u_i &\in [-3, 1], \\ J &= \sum_{i=0}^1 f_i^0(x_i, u_i), \end{aligned}$$

kde

$$f_0^0(x, u) = x^2 - 0,5u^2 - 2u - 1,5; \quad f_1^0(x, u) = x^2 + u^2 + 4u + 3.$$

Riešenie: Pretože úloha má voľný koniec, tak $V_2(x) = 0$ pre každé x .

Pre $k = 1$ dostávame

$$V_1(x) = \min_{u \in [-3, 1]} (x^2 + u^2 + 4u + 3 + V_2(F)) = x^2 + 4 - 8 + 3 = x^2 - 1,$$

pretože funkcia $u^2 + 4u$ nadobúda minimum v $u = -2$, t.j. $v_1(x) = -2$.

Pre $k = 0$ dostávame

$$\begin{aligned} V_0(x) &= \min_{u \in [-3, 1]} (x^2 - 0,5u^2 - 2u - 1,5 + V_1(x + u)) = \\ &= \min_{u \in [-3, 1]} (x^2 - 0,5u^2 - 2u - 1,5 + (x + u)^2 - 1). \end{aligned}$$

Pre zjednodušenie môžeme počítať hodnotovú funkciu $V_0(x)$ priamo v bode $x_0 = 1$, t.j.

$$\begin{aligned} V_0(1) &= \min_{u \in [-3, 1]} (1 - 0,5u^2 - 2u - 1,5 + (1 + u)^2 - 1) = \\ &= \min_{u \in [-3, 1]} 0,5(u^2 - 1) = -0,5 \end{aligned}$$

a toto minimum sa dosahuje pri $u_0 = 0$ Vypočítajme teraz odozvu na riadenie. Pretože $x_0 = 1$, $u_0 = 0$, tak $x_1 = x_0 + u_0 = 1 + 0 = 1$ a pretože $u_1 = v_1(x) = -2$, tak $x_2 = x_1 + u_1 = 1 - 2 = -1$

Optimálne riadenie a odozva teda sú

$$\hat{\mathcal{U}} = \{u_0, u_1\} = \{0, -2\}; \quad \hat{\mathcal{X}} = \{x_0, x_1, x_2\} = \{1, 1, -1\}$$

Príklad 2. Optimálne rozdeľovanie zdrojov.

Úlohu sformulovanú v prvej kapitole budeme riešiť v prípade $k = 2$.

Úpravou výrazov v diferenčnej rovnici a v účelovej funkcii dostávame nasledovnú úlohu

$$x_{i+1} = 0,3u_i x_i + 0,4x_i, \quad i = 0, 1$$

$$u_i \in [0, 1]$$

$$x_0 = a > 0$$

$$\min J = \sum_{i=0}^1 [0, 5u_i x_i - 2, 5x_i].$$

Pretože úloha je s voľným koncom, je $V_2(x) = 0$ pre každé x .

Pre $k = 1$ a ľubovoľné $x > 0$ dostávame

$$V_1(x) = \min_{u_1 \in [0,1]} [0, 5u_1 x - 2, 5x] = -2, 5x,$$

pretože minimum rastúcej lineárnej funkcie sa dosiahlo v ľavom koncovom bode intervalu, t.j. v $v_1(x) = u_1 = 0$.

Pre $k = 0$ a $x = a$ dostávame

$$V_0(a) = \min_{u_0 \in [0,1]} [f_0^0(a, u_0) + V_1(F_0(a, u_0))] =$$

$$\min_{u_0 \in [0,1]} [0, 5u_0 a - 2, 5a - 2, 5F_0(a, u_0)] = \min_{u_0 \in [0,1]} [0, 5u_0 a - 2, 5a - 2, 5(0, 3u_0 a + 0, 4a)] =$$

$$\min_{u_0 \in [0,1]} [0, 5u_0 a - 2, 5a - 0, 75u_0 a - a] = \min_{u_0 \in [0,1]} [-0, 25u_0 a - 3, 5a] = -3, 75a$$

kde minimum sa dosiahlo pre $v_0(a) = u_0 = 1$.

Optimálne riadenie uvedeného systému na dobu dvoch období je teda dané týmto pravidlom: v prvom období je treba použiť všetky prostriedky prvým spôsobom a v druhom období všetky ostatné prostriedky druhým spôsobom.

Pre uvedené dva príklady bolo možné vypočítať optimálnu hodnotu účelovej funkcie a optimálnu spätnú väzbu z rovnice dynamického programovania analyticky. Toto je však málokedy možné a tieto dva príklady sú do určitej miery "školské", umožňujúce si overiť pochopenie hlavnej myšlienky metódy.

Nasledujúci príklad je iného druhu. Predstavuje veľmi jednoduchý variant rozsiahlej triedy úloh optimálnej obnovy a je pre nás poučný hlavne z dvoch dôvodov. V prvom rade uvidíme, že hodnoty riadenia v tomto príklade nemajú číselný, ale logický charakter. Na druhej strane si na tomto príklade ukážeme metodiku priameho rekurentného výpočtu hodnotovej funkcie a optimálnej spätnej väzby pre úlohy s konečným počtom hodnôt stavu a riadenia.

Príklad 3. Stroj na lisovanie šťavy z ríbezlí sa zanáša a podľa toho, koľko bolo na ňom vylisované sa jeho výkon znižuje a to tak, že po každom kile vylisovaných ríbezlí sa čas lisovania nasledujúceho kila zvýši o 20 minút. Vyčistenie stroja trvá 30 minút. Ríbezle sú balené po 1 kg, ktorý po načatí treba ihneď spracovať. Celkove treba spracovať k kg ríbezlí.

Použitím metódy dynamického programovania rozhodnite, či a po ktorom kile stroj čistiť, aby sa žiadané množstvo spracovalo za najkratší čas.

Sformulujme problém ako úlohu optimálneho riadenia. Je to možné urobiť viacerými spôsobmi aj preto, že úloha nie je dostatočne sformulovaná. Zo zadania nie je napríklad jasné, či na začiatku máme k dispozícii čistý stroj, alebo či už na ňom bolo niekoľko kíl ríbezlí vylisovaných, či na konci musíme stroj vyčistiť a pod. Úlohu teda doformulujeme, a budeme predpokladať, že na začiatku máme stroj, na ktorom bolo vylisovaných (po poslednom čistení) $a \geq 0$ kilo ríbezlí a na konci necháme stroj nevyčistený.

Zavedieme nasledovné označenia:

i – poradové číslo kilogramu, ktoré sa lisuje, $i = 1, \dots, k$

x_i – počet spracovaných kg od posledného čistenia pri lisovaní i -teho kg

u_i – stroj čistíme $u_i = \check{C}$ alebo nečistíme $u_i = N$ pred spracovaním i -teho kila

Časová postupnosť krokov v i -tom intervale je teda nasledovná. Najprv máme informáciu x_i o stupni znečistenia stroja. V zápatí príde naše rozhodnutie u_i o čistení resp. nečistení a prípadne sa stroj vyčistí. Potom nasleduje lisovanie i -teho kila a nakoniec si poznačíme, že na stroji bolo od posledného čistenia vylisovaných x_{i+1} kilo ríbezlí. Predanú úlohu teda dostaneme diferenčnú rovnicu

$$x_{i+1} = \begin{cases} x_i + 1, & \text{ak } u_i = N \\ 1, & \text{ak } u_i = \check{C} \end{cases}, \quad i = 1, \dots, k,$$

začiatočnú podmienku

$$x_1 = a \geq 0,$$

koncovú podmienku

$$x_{k+1} \text{ voľné}$$

a účelovú funkciu

$$J = \sum_{i=1}^k f_i^0(x_i, u_i), \quad \text{kde} \quad f_i^0(x_i, u_i) = \begin{cases} 30, & \text{ak } u_i = \check{C} \\ 20x_i, & \text{ak } u_i = N \end{cases}.$$

Všimnime si, že sme dostali úlohu, s konečným počtom hodnôt Túto úlohu už nie je možné riešiť analyticky, ale použijeme rekurentnú schému uvedenú ako Tabuľka 1 na nasledujúcej strane, pomocou ktorej natabelujeme príslušné hodnoty vstupujúce do rovnice dynamického programovania postupne, začínajúc od konca, t.j. od $i = k + 1$. Keď je tabuľka vyplnená, môžeme z nej vyčítať optimálnu hodnotu účelovej funkcie a zrekonštruovať optimálne riadenie a jeho odozvu. Optimálnu hodnotu účelovej funkcie pre začiatočnú hodnotu $a = x_0$ prečítame v stĺpci $V_i(x)$, pre hodnotu $i = 0$ a $x = a$. Postupne od konca tabuľky smerom nahor nájdeme aj optimálne riadenie(a) a jeho odozvu.

Kvôli jednoduchosti budeme úlohu riešiť pre prípad $k = 5$ a $a = 0$. V tomto prípade vyjde optimálna hodnota účelovej funkcie $V_0(0) = 100$ a optimálne sú až tri riadenia:

$$\begin{aligned}\mathcal{U}_1 &= \{N, \check{C}, N, \check{C}, N\}, & \mathcal{X}_1 &= \{0, 1, 1, 2, 1, 2\} \\ \mathcal{U}_2 &= \{N, N, \check{C}, \check{C}, N\}, & \mathcal{X}_2 &= \{0, 1, 2, 1, 1, 2\} \\ \mathcal{U}_3 &= \{N, N, \check{C}, N, \check{C}\}, & \mathcal{X}_3 &= \{0, 1, 2, 1, 2, 1\}.\end{aligned}$$

Tabuľka 1.

i	x	u	$f_i^0(x, u)$	$F_i(x, u)$	$f_i^0 + V_{i+1}(F_i)$	$V_i(x)$	$v_i(x)$
6						0	
5	$\boxed{1}$	$\check{C}$ $\boxed{N}$	30 20	$\boxed{1}$ $\boxed{2}$	$30 + 0 = 30$ $20 + 0 = 20$	20	$\boxed{N}$
5	2	$\check{C}$ N	30 40	1 3	$30 + 0 = 30$ $40 + 0 = 40$	30	$\check{C}$
5	3	$\check{C}$ N	30 60	1 4	$30 + 0 = 30$ $60 + 0 = 60$	30	$\check{C}$
4	1	$\check{C}$ N	30 20	1 2	$30 + 20 = 50$ $20 + 30 = 50$	50	$N, \check{C}$
4	$\boxed{2}$	$\boxed{\check{C}}$ N	30 40	$\boxed{1}$ 3	$30 + 20 = 50$ $40 + 30 = 70$	50	$\boxed{\check{C}}$
4	3	$\check{C}$ N	30 60	1 4	$30 + 20 = 50$ $60 + 30 = 90$	50	$\check{C}$
3	$\boxed{1}$	$\check{C}$ $\boxed{N}$	30 20	$\boxed{1}$ $\boxed{2}$	$30 + 50 = 80$ $20 + 50 = 70$	70	$\boxed{N}$
3	2	$\check{C}$ N	30 40	1 3	$30 + 50 = 80$ $40 + 50 = 90$	80	$\check{C}$
3	3	$\check{C}$ N	30 60	1 4	$30 + 50 = 80$ $60 + 50 = 110$	80	$\check{C}$
2	$\boxed{1}$	$\boxed{\check{C}}$ N	30 20	$\boxed{1}$ 2	$30 + 80 = 100$ $20 + 80 = 100$	50	$N, \boxed{\check{C}}$
2	2	$\check{C}$ N	30 40	1 3	$30 + 70 = 100$ $40 + 80 = 120$	100	$\check{C}$
2	3	$\check{C}$ N	30 60	1 4	$30 + 70 = 100$ $60 + 80 = 140$	100	$\check{C}$
1	$\boxed{0}$	$\check{C}$ $\boxed{N}$	30 0	$\boxed{1}$ $\boxed{1}$	$30 + 100 = 130$ $0 + 100 = 100$	$\boxed{100}$	$\boxed{N}$

Príklad 4. Na príklade o optimálnej spotrebe z Odseku 1.1 si ukážeme ako možno použiť metódu dynamického programovania na riešenie úloh s kontinuálnym charakterom stavových resp. riadiacich premenných. V tomto príklade kontinuálne veličiny nahradíme diskretnými tak, aby stavová premenná nadobúdala iba hodnoty celých čísel. Úlohu budeme riešiť pre prípad $r = 0,5$, $k = 3$, $\delta = 0$, $x_0 = 3$ a $x_3 = 2$. Dostaneme tak nasledovnú úlohu:

$$x_{i+1} = 1,5x_i - u_i, \quad u_i \geq 0, \quad x_i \in N, \quad i = 0, 1, 2, 3$$

$$x_0 = 2, \quad x_3 = 2,$$

$$\min J = \sum_{i=0}^2 (-\ln u_i).$$

Pre takúto úlohu dostaneme nasledovnú tabuľku:

Tabuľka 2.

i	x	u	$f_i^0(x, u)$	$F_i(x, u)$	$f_i^0 + V_{i+1}(F_i)$	$V_i(x)$	$v_i(x)$
3	2					0	
2	2	1	$\ln 1$	2	$0 + 0 = 0$	0	1
2	3	2,5	$-\ln 2,5$	2	$0 - \ln 2,5 = -\ln 2,5$	$-\ln 2,5$	2,5
2	4	4	$-\ln 4$	2	$0 - \ln 4 = -\ln 4$	$-\ln 4$	4
1	2	1	$-\ln 1$	2	$0 + 0 = 0$	0	1
1	3	$\begin{matrix} 0,5 \\ \textstyle\boxed{1,5} \\ 2,5 \end{matrix}$	$\begin{matrix} -\ln 0,5 \\ -\ln 1,5 \\ -\ln 2,5 \end{matrix}$	$\begin{matrix} 4 \\ \textstyle\boxed{3} \\ 2 \end{matrix}$	$\begin{matrix} -\ln 0,5 - \ln 4 = -\ln 2 \\ -\ln 1,5 - \ln 2,5 = -\ln 3,75 \\ -\ln 2,5 - 0 = -\ln 2,5 \end{matrix}$	$-\ln 3,75$	1,5
1	4	$\begin{matrix} 1 \\ 2 \\ 3 \\ 4 \end{matrix}$	$\begin{matrix} -\ln 1 \\ -\ln 2 \\ -\ln 3 \\ -\ln 4 \end{matrix}$	$\begin{matrix} 5 \\ 4 \\ 3 \\ 2 \end{matrix}$	$\begin{matrix} 0 - \ln 5,5 = -\ln 5,5 \\ -\ln 2 - \ln 4 = -\ln 8 \\ -\ln 3 - \ln 2,5 = -\ln 7,5 \\ -\ln 4 - 0 = -\ln 4 \end{matrix}$	$-\ln 8$	2
0	3	$\begin{matrix} 0,5 \\ \textstyle\boxed{1,5} \\ 2,5 \end{matrix}$	$\begin{matrix} -\ln 0,5 \\ -\ln 1,5 \\ -\ln 2,5 \end{matrix}$	$\begin{matrix} 4 \\ \textstyle\boxed{3} \\ 2 \end{matrix}$	$\begin{matrix} -\ln 0,5 - \ln 8 = -\ln 4 \\ -\ln 1,5 - \ln 3,75 = -\ln 5,625 \\ -\ln 2,5 - 0 = -\ln 2,5 \end{matrix}$	$-\ln 5,625$	1,5

Z jej hodnôt vyčítame, že optimálna hodnota účelovej funkcie je $V_0(3) = -\ln 5,625$,

optimálne riadenie je $\mathcal{U} = \{1, 5; 1, 5; 2, 5\}$ a jeho odozva je $\mathcal{X} = \{3, 3, 3, 2\}$.

2.4. Poznámky

V prípade, že $X_i = X$, $U_i = U$ sú konečné množiny nezávislé od i , možno výpočet pomocou rovnice dynamického programovania schematicky znázorniť nasledovne:

```

1. Read   $k, X, U, C, F_i(x, u), f_i^0(x, u), x_0$ 
   if     $x \in C$  then   $V_k(x) := 0$ 
   if     $x \notin C$  then  $V_k(x) := \infty$ 
2. For    $j = k - 1, \dots, 0$  do
       For  $x \in X$  do
           For  $u \in U$  do
                $f_j^0 := f_j^0(x, u)$ 
                $F_j := F_j(x, u)$ 
                $g_j(u) := f_j^0 + V_{j+1}(F_j)$ 
                $V_j(x) := \min_u g_j(u)$ 
                $v_j(x) := \arg \min_u g_j(u)$ 
3. Set    $x_j := x_0$ 
       For  $j = 0, \dots, k - 1$  do
            $v_j := v_j(x_j)$ 
           write  $j, v_j, x_j$ 
            $x_{j+1} := F_j(x_j, v_j)$ 

```

Predpokladajme, že počet prvkov množiny X je p , počet prvkov množiny U je q ; ďalej, že cieľová množina C aj začiatková množina P je jednobodová. Ak by sme hľadali optimálne riadenie pasívne prebratím všetkých možností, potom by sme museli urobiť rádovo q^k výpočtov funkcií f_i^0, F_i . Pre rekurentný výpočet podľa schémy v prílohe ich treba rádovo kpq . Pri väčšom počte krokov k to značí podstatnú úsporu. Poznamenajme však, že za túto úsporu platíme vyššími pamäťovými nárokmi. Zatiaľ čo pri prehľadávaní všetkých možností si pamätáme iba poslednú najlepšiu možnosť, v prípade rekurentného výpočtu podľa Bellmanovej rovnice si musíme pamätať hodnoty $V_j(x)$ pre všetky x a j , t.j. spolu kp hodnôt. Úspora v počte vykonaných operácií sa prejaví výraznejšie, ak ide o úlohy so stavmi, opísanými vektormi, ktorých zložky majú kontinuálny charakter. Ak totiž každú zo zložiek diskretizujeme do N hodnôt a dimenzia vektora je n , dostávame $q = N^n$, čo už pri malých n môže byť veľmi veľké číslo. Pre väčšie n môže byť objem výpočtov i pri použití metódy dynamického programovania taký veľký, že sa nedá zvládnuť ani vysoko výkonným počítačom.

Ak ide o diskretizáciu úlohy s kontinuálnym charakterom stavu, často treba pri výpočte V_j použiť hodnotu V_{j+1} v takom bode, ktorý nie je tabelovaný. Vtedy treba buď interpolovať alebo zaokrúhľovať $F_j(x, u)$. Pri riešení kontinuálnych úloh diskretizáciou dostávame vo všeobecnosti len približné riešenie pôvodnej úlohy. Čím hustejšiu sieť diskretizácie volíme, tým je väčšia pravdepodobnosť nájsť presnejšie riešenie, ale stúpajú pamäťové a

výpočtové nároky.

Práve tak, ako sme optimálnu spätnú väzbu konštruovali rekurentne pre j klesajúce ("zozadu"), mohli by sme ju konštruovať aj pre j rastúce spredu. Vyplýva to z modifikácie princípu optimality. Určité drobné komplikácie formálneho charakteru vznikajú tým, že funkcia F nemusí byť pre pevné u jednoznačná. Príčina toho, prečo sa V_j konštruujú väčšinou zozadu má však hlbšie korene a súvisí s interpretáciou v_j ako optimálnej spätnej väzby, čo sa markantne prejaví v stochastickom dynamickom programovaní, kde už postup obrátiť nemožno.

2.5. Autonómne úlohy s voľným časom.

Úlohy na nekonečnom horizonte.

Rovnicu dynamického programovania sme odvodili pre *neautonómnu* úlohu s *pevným* časom, t.j. pre úlohu:

$$x_{i+1} = F_i(x_i, u_i), \quad i = 0, \dots, k-1, \quad k \text{ dané}$$

$$x_i \in X_i, \quad u_i \in U_i$$

$$x_0 = \hat{x}_0, \quad x_k \in C$$

$$J(x_0, \mathcal{U}) = \sum_{i=0}^{k-1} f_i^0(x_i, u_i).$$

Pre túto úlohu sme odvodili funkcie $V_j(x)$, $v_j(x)$, ktoré v skutočnosti záviseli aj od k , t.j. mali sme vlastne funkcie $V_{j,k}(x)$, $v_{j,k}(x)$, len sme tam k neuvádzali, lebo bolo doteraz stále to isté - pevne zvolené.

Autonómne úlohy sú úlohy, v ktorých F_i , f_i^0 , X_i , U_i nezávisia od i , t.j., pre všetky i sú stále tie isté, nemenia sa s časom. *Autonómne úlohy s pevným časom* sú špeciálnym prípadom neautonómnych úloh s pevným časom a teda celá teória Vety 2.1-2.3 ostávajú v plnom rozsahu v platnosti. Z autonómnosti úlohy však nevyplýva autonómnosť funkcií $V_{j,k}(x)$, $v_{j,k}(x)$. Tieto však budú závislé len od dĺžky $[j, k]$:

$$V_{j,k}(x) = V_{k-j}(x), \quad v_{j,k}(x) = v_{k-j}(x).$$

Úlohy s voľným časom sú úlohy, v ktorých nie je k vopred dané. V týchto úlohách sa teda optimalizuje aj vzhľadom na k . Riadením sú konečné postupnosti typu $\{u_0, \dots, u_i, \dots, u_k\}$, $u_i \in U_i$. Vety 2.1 a 2.2 platia aj pre úlohy s voľným časom. Dôkazy sú len malou obmenou dôkazov pre úlohy s pevným časom. Funkcie $V_{j,k}(x)$, $v_{j,k}(x)$ nezávisia od k , ani od dĺžky intervalu $[j, k]$, sú funkciou len konkrétneho j (a x). Veta 2.3. však platí len ako nutná podmienka, pritom nevieme definovať $V_k(x)$ a teda získaný rekurentný vzťah nemá v prípade *neautonómnej úlohy s voľným časom* praktický význam. V tomto prípade musíme teda pre každé pevne zvolené k riešiť úlohu s pevným časom na $[0, k]$ a spomedzi takto získaných riešení vybrať to, v ktorom účelová funkcia nadobúda najmenšiu hodnotu.

Autonómne úlohy s voľným časom. Situácia je trochu iná v prípade že je úloha s voľným časom autonómna. Funkcie $V_{j,k}(x)$, $v_{j,k}(x)$, teraz už nezávisia ani len od j , t.j.

$$V_{j,k}(x) = V(x), \quad v_{j,k}(x) = v(x).$$

Podobnými úvahami ako sme odvodili rovnicu dynamického programovania pre úlohu s pevným časom, dostaneme, že ak V je hodnotová funkcia a v je optimálna spätná väzba, potom V, v vyhovujú funkcionálnej rovnici

$$V(x) = f^0(x, v(x)) + V(F(x, v(x))) = \min_{u \in U} [f^0(x, u) + V(F(x, u))]. \quad (2.4)$$

Táto funkcionálna rovnica už nie je rekurentným vzťahom, a teda nedáva návod na výpočet funkcie V . Navyše, (2.4) je len nutnou podmienkou, postačujúcosť už nie je zrejmé a vo všeobecnosti ani neplatí.

S podobným problémom sa stretávame aj pri *autonómnych úlohách s nekonečným horizontom*. Tak sa nazývajú úlohy optimálneho riadenia ekonomického charakteru, v ktorých $T = \infty$ a riadeniami sú nekonečné postupnosti typu $\{u_0, \dots, u_i, \dots\}$, $u_i \in U_i$. Tu samozrejme treba zaručiť, že minimálna hodnota účelovej funkcie je konečná, čo sa niekedy dosahuje zavedením "diskontového koeficientu" $\alpha = \frac{1}{1+r} < 1$ ($r > 0$) do účelovej funkcie, ktorá bude mať tvar

$$J = \sum_{i=0}^{\infty} \alpha^i f^0(x_i, u_i). \quad (1.9a)$$

Diskontový koeficient slúži na prepočet hodnoty v i -tom roku na hodnotu v 0-tom roku pri úrokovej sadzbe $100r$ percent. Ľahko sa dá ukázať, že pre úlohu optimálneho riadenia s účelovou funkciou (1.9a) má rovnice dynamického programovania tvar

$$V(x) = f^0(x, v(x)) + \alpha V(F(x, v(x))) = \min_{u \in U} [f^0(x, u) + \alpha V(F(x, u))]. \quad (2.4a)$$

Používajú sa dve techniky na riešenie funkcionálnej rovnice (2.4a):

(a) *Aproximácie v priestore hodnotových funkcií.*

- Zvolíme $V^{(0)}(x)$
- S daným $V^{(0)}$ riešime úlohu na minimum

$$\min_u [f^0(x, u) + \alpha V^{(0)}(F(x, u))] = [f^0(x, v^{(0)}(x)) + \alpha V^{(0)}(F(x, v^{(0)}(x)))];$$

Dostaneme $v^{(0)}(x)$

- Položíme

$$V^{(1)}(x) := [f^0(x, v^{(0)}(x)) + \alpha V^{(0)}(F(x, v^{(0)}(x)))];$$

atď.

(b) *Aproximácie v priestore riadení.* - Zvolíme $v^{(0)}(x)$ - S daným $v^{(0)}$ riešime funkcionálnu rovnicu pre $V^{(0)}(x)$:

$$V^{(0)}(x) = f^0(x, v^{(0)}(x)) + \alpha V^{(0)}(F(x, v^{(0)}(x))).$$

Dostaneme $V^{(0)}(x)$. - Nové $v^{(1)}$ dostaneme riešením

$$\min_u [f^0(x, u) + \alpha V^{(0)}(F(x, u))] = [f^0(x, v^{(1)}(x)) + \alpha V^{(0)}(F(x, v^{(1)}(x)))];$$

-atď ($v^{(1)}$ použijeme na výpočet $V^{(1)}$).

Poznamajme, že vo všeobecnosti nie je zaručené, že uvedené dve techniky konvergujú, resp. ak konvergujú, že konvergujú ku riešeniu (4) a toto riešenie je aj hodnotovou funkciou. Existujú však triedy úloh, pre ktoré je použitie niektorej z hore uvedených techník plne zdôvodnené.

2.6. Lineárne-kvadratická úloha

Rovnica dynamického programovania dáva možnosť rekurentne počítať optimálnu spätnú väzbu a hodnotovú funkciu "od konca". Analyticky je to však možné málokedy; okrem nasledujúceho príkladu lineárne-kvadratickej úlohy bez ohraničení, ktorá je dôležitá jednak pre numerické metódy, jednak pre stochastickú teóriu riadenia, sú známe len ojedinelé konkrétne úlohy, kde je to možné.

Uvažujme úlohu optimálneho riadenia danú diferenčnou rovnicou

$$x_{i+1} = A_i x_i + B_i u_i, \quad i = 0, \dots, k-1,$$

kde $x_i \in \mathbb{R}^n$, $u_i \in \mathbb{R}^m$, A_i sú matice typu $(n \times n)$ a B_i sú matice typu $(n \times m)$. Účelová funkcia je

$$J = \sum_{i=0}^{k-1} (x_i^T Q_i x_i + u_i^T R_i u_i),$$

kde R_i , Q_i sú symetrické matice typu $(n \times n)$ a $(m \times m)$, pričom $R_i > 0$, $Q_i \geq 0$. Úloha je bez ohraničení na riadenie a stavovú premennú, s voľným koncom, t.j. $U_i = \mathbb{R}^m$, $X_i = \mathbb{R}^n$, $C = \mathbb{R}^n$. Túto úlohu budeme nazývať lineárne kvadratickou úlohou a označovať (LQ).

Aplikujúc rovnicu dynamického programovania na túto úlohu dostávame, že $V_k(x) = 0$ pre každé x a pre $k-1$ platí

$$V_{k-1}(x) = \min_u [x^T Q_{k-1} x + u^T R_{k-1} u] = x^T Q_{k-1} x + \min_u [u^T R_{k-1} u] = x^T Q_{k-1} x$$

pretože $R_{k-1} > 0$; z toho súčasne vyplýva, že $v_{k-1}(x) = 0$.

Vidíme, že pre $j = k-1$ je V_{k-1} kladne semidefinitná kvadratická forma, z čoho možno usudzovať, že to bude aj pre $j < k-1$. Dokážeme to indukciou a súčasne dokážeme, že v_j

je lineárna funkcia x , $v_j(x) = H_j x$. Predpokladajme teda, že $V_{j+1}(x) = x^T W_{j+1} x$, kde W_{j+1} je kladne semidefinitná. Hľadáme riešenie Bellmanovej rovnice v tvare $V_j(x) = x^T W_j x$. Dostávame

$$\begin{aligned} V_j(x) &= x^T W_j x = \min_u [x^T Q_j x + u^T R_j u + (A_j x + B_j u)^T W_{j+1} (A_j x + B_j u)] = \\ &= x^T Q_j x + x^T A_j^T W_{j+1} A_j x + \\ &+ \min_u [u^T (R_j + B_j^T W_{j+1} B_j) u + u^T B_j^T W_{j+1} A_j x + x^T A_j^T W_{j+1} B_j u]. \end{aligned} \quad (2.5)$$

Označme funkciu, ktorej minimum hľadáme vo výraze (2.5) ako g , t.j.

$$g(u) = [u^T (R_j + B_j^T W_{j+1} B_j) u + u^T B_j^T W_{j+1} A_j x + x^T A_j^T W_{j+1} B_j u].$$

Zrejme

$$\begin{aligned} \nabla_u g &= 2(R_j + B_j^T W_{j+1} B_j)u + 2B_j^T W_{j+1} A_j x, \\ \nabla_u^2 g &= 2(R_j + B_j^T W_{j+1} B_j). \end{aligned}$$

Z $W_{j+1} \geq 0$ vyplýva $B_j^T W_{j+1} B_j \geq 0$ a teda $R_j + B_j^T W_{j+1} B_j > 0$. Z toho vyplýva, že g je konvexná a minimum nadobúda práve vo svojich stacionárnych bodoch, t.j. v bodoch, kde jej gradient podľa u je rovný nule.

Preto sa minimum dosiahne pre

$$u = -(R_j + B_j^T W_{j+1} B_j)^{-1} B_j^T W_{j+1} A_j x$$

z čoho vyplýva

$$v_j(x) = H_j x \quad (2.6)$$

kde

$$H_j = -(R_j + B_j^T W_{j+1} B_j)^{-1} B_j^T W_{j+1} A_j. \quad (2.7)$$

Dosadením (2.6) a (2.7) do (2.5) dostaneme

$$\begin{aligned} x^T W_j x &= \\ &= x^T [Q_j + A_j^T W_{j+1} A_j + H_j^T (R_j + B_j^T W_{j+1} B_j) H_j + H_j^T B_j^T W_{j+1} A_j + A_j^T W_{j+1} B_j H_j] x \\ &= x^T [Q_j + A_j^T W_{j+1} A_j - A_j^T W_{j+1} B_j^T (R_j + B_j W_{j+1} B_j)^{-1} B_j^T W_{j+1} A_j] x \end{aligned}$$

Odtiaľ

$$W_j = Q_j + A_j^T [W_{j+1} - W_{j+1} B_j (R_j + B_j^T W_{j+1} B_j)^{-1} B_j^T W_{j+1}] A_j. \quad (2.8)$$

Z toho vyplýva, že $V_j(x) = x^T W_j x$ a $v_j(x) = H_j x$. Z výrazu (2.8) je vidieť, že W_j je symetrická. Funkcia $V_j(x)$ je hodnotová funkcia pre LQ problém, kde účelová funkcia nadobúda iba nezáporné hodnoty. Z toho vyplýva, že $x^T W_j x = V_j(x) \geq 0$, pre každé $x \in \mathbb{R}^n$ a teda W_j je kladne semidefinitná. Tým sme dokázali nasledovnú vetu:

Veta 2.4. $V_j(x)$, $v_j(x)$ sú hodnotová funkcia a optimálna spätná väzba pre úlohu (LQ) vtedy a len vtedy, keď

$$V_j(x) = x^T W_j x,$$

$$v_j(x) = H_j x,$$

kde W_j je riešenie (2.8) s $W_{k-1} = Q_{k-1}$ a H_j je dané vzťahom (2.7) pre $j = 0, \dots, k-2$.

Pod autonómnou LQ úlohou na nekonečnom časovom horizonte, rozumieme úlohu

$$x_{i+1} = Ax_i + Bu_i, \quad i = 0, 1, \dots,$$

$$J = \sum_{i=0}^{\infty} (x_i^T Q x_i + u_i^T R u_i),$$

$$R, Q \text{ symetrické, } R > 0, \quad Q \geq 0, \quad U = \mathbb{R}^m, \quad X = \mathbb{R}^n, \quad C = \mathbb{R}^n.$$

Pre takéto úlohy je RDP len nutnou podmienkou optimality; hodnotová funkcia a optimálna spätná väzba nezávisia od i , teda: Ak $V(x)$, $v(x)$ sú hodnotová funkcia a optimálna spätná väzba potom spĺňajú

$$V(x) = x^T Q x + v^T(x) R v(x) + V(Ax + Bv(x)) = \min_u [x^T Q x + u^T R u + V(Ax + Bu)]$$

V triede funkcií typu $V(x) = x^T W x$, $W \geq 0$ má táto rovnica jediné riešenie a to $W \geq 0$, ktoré spĺňa tzv. Riccatiho maticovú rovnicu

$$W = Q + A^T [W - WB(R + B^T W B)^{-1} B^T W] A. \quad (2.9)$$

Iba v niektorých jednoduchých prípadoch možno túto rovnicu analyticky vyriešiť. Zvyčajne je však potrebné použiť nejakú vhodnú iteračnú metódu. Napr.

$$W^{(k+1)} = Q + A^T [W^{(k)} - W^{(k)} B (R + B^T W^{(k)} B)^{-1} B^T W^{(k)}] A, \quad W^{(0)} = 0,$$

o ktorej sa dá (pri špeciálnych predpokladoch) dokázať, že $W = \lim_{k \rightarrow \infty} W^{(k)}$.

Použitie uvedenej teórie pre LQ úlohu môžeme ilustrovať na nasledujúcom príklade.

Príklad. Riešte úlohu optimálneho riadenia

$$x_{i+1} = x_i - u_i, \quad i = 0, \dots, k-1,$$

$$J = \sum_{i=0}^{k-1} [u_i^2 + x_i^2].$$

Úloha je bez ohraničení na riadenie a stav, má voľný koniec. Nájdite hodnotovú funkciu a optimálnu spätnú väzbu v prípade (a) $k = 5$ a (b) $k = \infty$.

(a) Riešenie pre $k = 5$. Úloha je v tvare (LQ), kde $A = 1$, $B = -1$, $R = 1$ a $Q = 1$. Dosadením do vzorcov (2.8), (2.7), (2.6) dostávame

$$W_i = 1 + W_{i+1} - W_{i+1} (1 + W_{i+1})^{-1} W_{i+1} = \frac{1 + 2W_{i+1}}{1 + W_{i+1}}, \quad i = 0, \dots, 4$$

$$H_i = (1 + W_{i+1})^{-1} W_{i+1} = \frac{W_{i+1}}{1 + W_{i+1}}$$

$$W_5 = 0, \quad W_4 = 1, \quad W_3 = \frac{3}{2}, \quad W_2 = \frac{8}{5}, \quad W_1 = \frac{21}{13}, \quad W_0 = \frac{55}{34}$$

$$V_5 = 0, \quad V_4 = x^2, \quad V_3 = \frac{3}{2}x^2, \quad V_2 = \frac{8}{5}x^2, \quad V_1 = \frac{21}{13}x^2, \quad V_0 = \frac{55}{34}x^2$$

$$H_4 = 0, \quad H_3 = \frac{1}{2}, \quad H_2 = \frac{3}{5}, \quad H_1 = \frac{8}{13}, \quad H_0 = \frac{21}{34}$$

$$v_4(x) = 0, \quad v_3(x) = \frac{1}{2}x, \quad v_2(x) = \frac{3}{5}x, \quad v_1(x) = \frac{8}{13}x, \quad v_0(x) = \frac{21}{34}x$$

Optimálne riadenie a odozva pre zvolenú začiatočnú hodnotu vypočítame nasledovne

$$x_0 = 1, \quad u_0 = v_0(1) = \frac{21}{34}1 = \frac{21}{34}$$

$$x_1 = x_0 - u_0 = 1 - \frac{21}{34} = \frac{13}{34}, \quad u_1 = v_1\left(\frac{13}{34}\right) = \frac{8}{13} \frac{13}{34} = \frac{8}{34}$$

$$x_2 = x_1 - u_1 = \frac{13}{34} - \frac{8}{34} = \frac{5}{34}, \quad u_2 = v_2\left(\frac{5}{34}\right) = \frac{3}{5} \frac{5}{34} = \frac{3}{34}$$

$$x_3 = x_2 - u_2 = \frac{5}{34} - \frac{3}{34} = \frac{2}{34}, \quad u_3 = v_3\left(\frac{2}{34}\right) = \frac{1}{2} \frac{2}{34} = \frac{1}{34}$$

$$x_4 = x_3 - u_3 = \frac{2}{34} - \frac{1}{34} = \frac{1}{34}, \quad u_4 = v_4\left(\frac{1}{34}\right) = 0$$

$$x_5 = x_4 - u_4 = \frac{1}{34} - 0 = \frac{1}{34}$$

$$\hat{\mathcal{U}} = \{u_0, u_1, u_2, u_3, u_4, u_5\} = \left\{\frac{21}{34}, \frac{8}{34}, \frac{3}{34}, \frac{1}{34}, 0\right\}$$

$$\hat{\mathcal{X}} = \{x_0, x_1, x_2, x_3, x_4, x_5\} = \left\{1, \frac{13}{34}, \frac{5}{34}, \frac{2}{34}, \frac{1}{34}, \frac{1}{34}\right\}$$

(b) Riešenie pre $k = \infty$. V tomto prípade použijeme (2.9), (2.7) a (2.6).

Po dosadení dostávame Riccatiho rovnicu v tvare

$$W = \frac{1 + 2W}{1 + W},$$

ktorej kladným riešením je $W = \frac{1+\sqrt{5}}{2}$ a teda

$$H = \frac{W}{1 + W} = \frac{1 + \sqrt{5}}{3 + \sqrt{5}}, \quad v(x) = \frac{1 + \sqrt{5}}{3 + \sqrt{5}}x = \frac{\sqrt{5} - 1}{2}x.$$

Optimálnu odozvu dostaneme riešením rovnice

$$x_{i+1} = x_i - u_i = x_i - \frac{\sqrt{5}}{2}x_i + \frac{1}{2}x_i = \left(\frac{3}{2} - \frac{\sqrt{5}}{2}\right)x_i.$$

2.7. Iné uplatnenia dynamického programovania

Myšlienka vnorenia optimalizačnej úlohy do systému úloh, umožňujúceho rekurentný výpočet optimálneho riadenia, má široké uplatnenie.

2.7.1. Úlohy optimálneho riadenia s ohraňčeniami na stavovú premennú..

Metódu dynamického programovania možno použiť aj na také úlohy optimálneho riadenia, v ktorých vystupujú aj ohraňčenia typu $\phi_i(x_i, u_i) = 0$ alebo $\phi_i(x_i, u_i) \leq 0$. Zatiaľ čo pri iných metódach takéto ohraňčenia teóriu veľmi komplikujú, v tomto prípade nevznikajú žiadne komplikácie.

Príklad: Problém na úzke miesta. Uvažujme výrobný proces pozostávajúci z k krokov, kde v i -tom kroku je k dispozícii surovina o množstve x_i . Táto sa rozdeľuje na 3 časti. Jedna časť u_i je použitá na získanie ďalšej suroviny, druhá časť v_i je použitá na zvýšenie kapacity produkcie c_i , zvyšok je prístupný pre samotný výrobný proces. Pritom musí byť splnené, že

$$u_i \leq c_i.$$

Úlohou je maximalizovať celkové množstvo suroviny prístupnej pre výrobný proces za predpokladu, že rast x_i, c_i je daný diferenčnými rovnicami

$$x_{i+1} = (1 + \alpha)u_i \quad i = 0, \dots, k-1$$

$$c_{i+1} = c_i + \beta v_i, \quad \text{kde } \alpha, \beta > 0$$

a účelovou funkciou

$$J = \sum_{i=0}^{k-1} (x_i - u_i - v_i) \rightarrow \max_{u_i, v_i}$$

začiatkové a koncové podmienky na stavové veličiny x_i, c_i sú

$$x_0 = \hat{x}_0, \quad c_0 = \hat{c}_0, \quad \text{koniec je voľný.}$$

Ohraňčenia na riadenia u_i, v_i sú závislé aj od stavových premenných

$$0 \leq u_i, \quad 0 \leq v_i, \quad u_i + v_i \leq x_i \quad u_i \leq c_i.$$

Riešenie. Rovnica dynamického programovania má pre túto úlohu tvar

$$V_i(x_i, c_i) = \max_{u_i, v_i} [x_i - u_i - v_i + V_{i+1}((1 + \alpha)u_i, c_i + \beta v_i)], \quad i = k - 1, \dots, 0,$$

kde $u_i, v_i \leq 0$, $u_i + v_i \leq x_i$, $u_i \leq c_i$ a $V_k(x, c) = 0$ pre každé x, c . Pre $i = k - 1$ dostávame

$$V_{k-1}(x_{k-1}, c_{k-1}) = \max_{u_{k-1}, v_{k-1}} [x_{k-1} - u_{k-1} - v_{k-1} + 0] = x_{k-1},$$

kde

$$0 \leq u_{k-1}, \quad 0 \leq v_{k-1}, \quad u_{k-1} + v_{k-1} \leq x_{k-1}, \quad u_{k-1} \leq c_{k-1}$$

Teda maximum sa dosiahlo pre $u_{k-1} = 0$, $v_{k-1} = 0$. Pre $i = k - 2$ dostávame

$$\begin{aligned} V_{k-2}(x_{k-2}, c_{k-2}) &= \max_{u_{k-2}, v_{k-2}} [x_{k-2} - u_{k-2} - v_{k-2} + V_{k-1}((1 + \alpha)u_{k-2}, c_{k-2} + \beta v_{k-2})] = \\ &= \max_{u_{k-2}, v_{k-2}} [x_{k-2} - u_{k-2} - v_{k-2} + (1 + \alpha)u_{k-2}] = \max_{u_{k-2}, v_{k-2}} [x_{k-2} + \alpha u_{k-2} - v_{k-2}], \end{aligned}$$

kde maximum sa hľadá pri podmienkach

$$0 \leq u_{k-2}, \quad 0 \leq v_{k-2}, \quad u_{k-2} + v_{k-2} \leq x_{k-2}, \quad u_{k-2} \leq c_{k-2}$$

Maximum sa nadobúda pri daných ohraničeníach, ak $v_{k-2} = 0$ a zároveň u_{k-2} je maximálne možné spĺňajúce ohraničenia $0 \leq u_{k-2} \leq x_{k-2}$, $0 \leq u_{k-2} \leq c_{k-2}$. To znamená, že $u_{k-2} = \min\{x_{k-2}, c_{k-2}\}$ a teda

$$V_{k-2}(x_{k-2}, c_{k-2}) = x_{k-2} + \alpha \min\{x_{k-2}, c_{k-2}\}.$$

$$\begin{aligned} V_{k-3}(x_{k-3}, c_{k-3}) &= \max [x_{k-3} - u_{k-3} - v_{k-3} + V_{k-2}((1 + \alpha)u_{k-3}, c_{k-3} + \beta v_{k-3})] = \\ &= \max [x_{k-3} - u_{k-3} - v_{k-3} + (1 + \alpha)u_{k-3} + \alpha \min\{(1 + \alpha)u_{k-3}, c_{k-3} + \beta v_{k-3}\}] = \\ &= \max [x_{k-3} + \alpha u_{k-3} - v_{k-3} + \alpha \min\{(1 + \alpha)u_{k-3}, c_{k-3} + \beta v_{k-3}\}] \end{aligned}$$

Podobne pokračujeme ďalej, až kým nedostaneme $V_0(x_0, c_0)$.

2.7.2. Úlohy optimálneho riadenia s inými typmi účelových funkcií..

(a) *Účelová funkcia je multiplikatívneho typu.*

Príklad 1. Predstavme si zariadenie, ktoré pozostáva z k do série zapojených blokov. Správna funkcia zariadenia si vyžaduje, aby všetky jeho bloky bezchybne pracovali. Spoľahlivosť jednotlivých blokov možno zvyšovať paralelným zapájaním prvkov. Označíme $\varphi_i(u)$ pravdepodobnosť správnej funkcie i -tého bloku, ak obsahuje u paralelne zapojených prvkov. Nech je cena prvku i -tého bloku c_i . Úlohou je rozhodnúť, do ktorého bloku koľko prvkov zaradiť tak, aby ich celková cena neprevýšila danú hodnotu m , a aby pravdepodobnosť správnej funkcie zariadenia bola čo najvyššia.

Ak označíme x_i celkovú cenu prvkov v prvom až i -tom bloku, u_{i-1} počet prvkov v i -tom bloku, možno túto úlohu formulovať ako úlohu optimálneho riadenia

$$x_{i+1} = x_i + c_{i+1}u_i, \quad \hat{x}_0 = 0, \quad C = \{x; \quad x \leq m\}$$

$$J = - \prod_{i=0}^{k-1} \varphi_{i+1}(u_i),$$

ktorá sa od našej všeobecnej schémy líši iba tým, že J má tvar súčinu. Ľahko možno dokázať, že pre túto úlohu platí rekurentný vzťah

$$V_j(x) = \min_{u \geq 0} [\varphi_{j+1}(u) V_{j+1}(x + c_{j+1}u)], \quad i = 1, \dots, k-1,$$

$$V_k(x) = -1, \quad x \in C,$$

ktorý zodpovedá rovnici dynamického programovania.

(b) *Účelová funkcia je funkciou koncového stavu.* Ide o úlohu, ktorá sa od všeobecnej schémy líši iba účelovou funkciou, ktorá je v tvare $J = \phi(x_k)$. Pre takúto úlohu možno ľahko odvodiť RDP v tvare

$$V_i(x) = \min_u [V_{i+1}(F_i(x, u))], \quad i = 0, \dots, k-1$$

$$V_k(x) = \phi(x), \text{ pre } x \in C.$$

(c) *Účelová funkcia je max x_i .* DÚ.

2.7.3. Úlohy z iných oblastí optimalizácie.

Mnohé úlohy z iných oblastí optimalizácie sa dajú formulovať aj ako úlohy optimálneho riadenia.

(a) *Niektoré úlohy lineárneho a nelineárneho programovania.*

Príklad 2. Ide o klasickú úlohu lineárneho programovania - dopravný problém, a to v špeciálnom prípade dvoch výrobcov (alebo dvoch odberateľov). Nech dvaja výrobcovia vyrábajú p_1 , resp. p_2 jednotiek výrobku, ktorý sa dopravuje n odberateľom, odoberajúcim po rade $q_1, \dots, q_n$ jednotiek výrobku, pričom platí $p_1 + p_2 = q_1 + \dots + q_n$. Nech c_{ij} je cena dopravy jednotky výrobku od i -tého výrobcu k j -tému odberateľovi. Treba rozhodnúť, aké množstvo výrobku treba voziť od jednotlivých výrobcov k jednotlivým odberateľom, aby náklady na dopravu boli minimálne.

Ak označíme x_i^j , $j = 1, 2$ objem výrobku, ktorý sa od j -tého výrobcu dopraví celkovo k prvému až i -tému odberateľovi, u_{i-1} pomernú časť, ktorú i -tý odberateľ dostáva od prvého výrobcu, možno písať

$$x_{i+1}^1 = x_i^1 + q_{i+1}u_i, \quad x_0^1 = 0$$

$$x_{i+1}^2 = x_i^2 + q_{i+1}(1 - u_i), \quad x_0^2 = 0$$

$$C = \{(x^1, x^2); \quad x^1 = p_1, \quad x^2 = p_2\}, \quad u_i \in [0, 1]$$

$$J = \sum_{i=1}^k q_i [c_{1i} u_{i-1} + c_{2i} (1 - u_{i-1})] = \sum_{i=0}^{k-1} q_{i+1} [c_{1(i+1)} u_i + c_{2(i+1)} (1 - u_i)]$$

čo je úloha optimálneho riadenia, pre riešenie ktorej možno použiť metódu dynamického programovania. V tomto prípade to nebude výhodnejšie než obvyklé metódy pre riešenie úloh lineárneho programovania, ktoré sú veľmi účinné. Nemožno však vylúčiť, že pre nejakú z úloh nelineárneho programovania, pre riešenie ktorých neexistujú natoľko všeobecné metódy, by nemohol tento postup viesť k výhodnejšej metóde.

(b) *Niektoré úlohy teórie grafov.* Metóda dynamického programovania sa používa aj na riešenie mnohých úloh, ktoré nemajú formálny tvar úloh optimálneho riadenia. K typickým predstaviteľom takýchto úloh patrí úloha o najkratšej ceste a úloha obchodného cestujúceho. Takéto úlohy sa riešia buď metódou dynamického programovania alebo inými metódami napr. celočíselného programovania, metódy vetiev a hraníc. I keď metódy založené na princípe dynamického programovania pre tieto úlohy nevyžadujú komplikovanú terminológiu a formuláciu ako úlohy optimálneho riadenia - svojou podstatou sú to tiež úlohy optimálneho riadenia. Ukážeme si to na úlohe o najkratšej ceste.

Príklad 3: Úloha o najkratšej ceste. Uvažujme sieť ciest, kde body $M_1, \dots, M_N$ predstavujú mestá. Nech $d(M_i, M_j) = d_{ij}$ označuje priamu vzdialenosť z mesta M_i do mesta M_j a predpokladajme, že každá dvojica miest je spojená cestou. Treba nájsť najkratšiu cestu z mesta $M \in \{M_1, \dots, M_N\}$ do mesta M_N .

Naformulujme túto úlohu ako úlohu optimálneho riadenia. Označme i - poradové číslo mesta, kde $i = 0, \dots, k$

x_i - mesto, v ktorom sa nachádzame v čase i , pričom $x_i \in \{M_1, \dots, M_N\}$

u_i - riadenie, ktorým je naše rozhodnutie, že z mesta x_i pôjdeme do mesta x_{i+1} , t.j.,

$u_i \in \{M_1, \dots, M_N\} - \{x_i\}$. Dostávame teda úlohu optimálneho riadenia s diferenčnou rovnicou v tvare

$$x_{i+1} = u_i, \quad i = 1, \dots, k-1, \quad k \text{ voľné},$$

s začiatkovou a koncovou podmienkou na stavovú veličinu x_i

$$x_1 = \hat{M}, \quad x_k = M_N.$$

Účelová funkcia je

$$J = \sum_{i=0}^{k-1} d(x_i, u_i)$$

Z podmienok vyplýva, že ide o autonómnu úlohu s voľným časom. Pre takéto úlohy je rovnica dynamického programovania len nutnou podmienkou optimality. Nie je to rekurentný vzťah, ale funkcionálna rovnica, ako sme ukázali v časti 2.3. V našom prípade teda dostávame, že hodnotová funkcia pre daný problém spĺňa funkcionálnu rovnicu

$$V(x) = \min_{u \in \{M_1, \dots, M_N\} - x} [d(x, u) + V(u)].$$

Na riešenie takejto rovnice možno použiť metódu aproximácií v priestore hodnotových funkcií opísanú v časti 2.3. Podľa tejto metódy

$$V^{(0)}(x) = \begin{cases} 0, & \text{ak } x = M_N \\ \infty & \text{ak } x \neq M_N \end{cases}$$

$$V^{(k)}(x) = \begin{cases} \min_{u \in \{M_1, \dots, M_N\} - x} [d(x, u) + V^{(k-1)}(u)], & \text{ak } x \neq M_N \\ 0 & \text{ak } x = M_N \end{cases}, \quad k \geq 1.$$

Dá sa ukázať, že tento proces konverguje v N krokoch.

Poznamenajme, že v teórii grafov sa úloha o najkratšej ceste rieši aj metódou dynamického programovania. Úloha sa prirodzeným spôsobom (bez zložitej terminológie optimálneho riadenia) môže formulovať nasledovne

$$V(M_j) = \min_{M_i} (d_{ij} + V(M_i)), \text{ kde } M_i \neq M_j,$$

resp.

$$V(M_j) = f_j, \text{ kde } f_j = \min_{i \neq j} (d_{ij} + f_i)$$

a navrhuje sa iteračný postup pre nájdenie f_j , $i = 0, \dots, N-1$

$$f_j^{(k)} = \min_{i \neq j} (d_{ij} + f_i^{(k-1)}), \quad k \geq 1$$

$$f_i^{(0)} = \begin{cases} 0 & i = N \\ \infty & i \neq N \end{cases}$$

Vidieť, že ide o ten istý algoritmus ako v prípade aproximácií v priestore funkcií pre úlohu optimálneho riadenia. Použitie algoritmu ilustrujeme na príklade.

Príklad 4. Riešte úlohu o najkratšej ceste v prípade $N = 4$,

$$\begin{aligned} d(M_1, M_2) &= 4, & d(M_1, M_3) &= 2, & d(M_1, M_4) &= 4 \\ d(M_2, M_3) &= 1, & d(M_2, M_4) &= 1, & d(M_3, M_4) &= 4 \end{aligned}$$

Treba nájsť najkratšie cesty z miest M_1, M_2, M_3 do mesta M_4 .

Položme pre $\boxed{k=0}$ $V^{(0)}(M_1) = V^{(0)}(M_2) = V^{(0)}(M_3) = \infty$ a $V^{(0)}(M_4) = 0$.

Pre $\boxed{k=1}$ dostávame $V^{(1)}(M_4) = 0$ a

$$\begin{aligned} V^{(1)}(M_1) &= \min(d(M_1, M_2) + V^{(0)}(M_2), d(M_1, M_3) + V^{(0)}(M_3), (d(M_1, M_4) + V^{(0)}(M_4))) = \\ &= \min(\infty, \infty, 4 + 0) = 4, & v^{(1)}(M_1) &= M_4 \\ V^{(1)}(M_2) &= \min(d(M_2, M_3) + V^{(0)}(M_3), d(M_2, M_4) + V^{(0)}(M_4)) = \\ &= d(M_2, M_4) + V^{(0)}(M_4) = 1, & v^{(1)}(M_2) &= M_4 \\ V^{(1)}(M_3) &= d(M_3, M_4) = 4, & v^{(1)}(M_3) &= M_4 \end{aligned}$$

Pre $\boxed{k = 2}$ dostávame

$$V^{(2)}(M_4) = 0$$

$$V^{(2)}(M_1) = \min(d(M_1, M_2) + V^{(1)}, d(M_1, M_3) + V^{(1)}(M_3), d(M_1, M_4) + V^{(1)}(M_4)) = \min(4 + 1, 2 + 4, 4 + 0) = 4, \quad v^{(2)}(M_1) = M_4$$

$$V^{(2)}(M_2) = \min(d(M_2, M_1) + V^{(1)}(M_1), d(M_2, M_3) + V^{(1)}(M_3), d(M_2, M_4) + 0) = \min(4 + 4, 1 + 4, 1) = 1, \quad v^{(2)}(M_2) = M_4$$

$$V^{(2)}(M_3) = \min(d(M_3, M_1) + V^{(1)}(M_1), d(M_3, M_2) + V^{(1)}(M_2), d(M_3, M_4) + 0) = \min(2 + 4, 1 + 1, 4) = 2, \quad v^{(2)}(M_3) = M_2$$

Pre $\boxed{k = 3}$ dostávame

$$V^{(3)}(M_4) = 0$$

$$V^{(3)}(M_1) = \min(d(M_1, M_2) + V^{(2)}(M_2), d(M_1, M_3) + V^{(2)}(M_3), d(M_1, M_4) + 0) = \min(4 + 1, 2 + 2, 4) = 4, \quad v^{(3)}(M_1) = \{M_3, M_4\}$$

$$V^{(3)}(M_2) = \min(d(M_2, M_1) + V^{(2)}(M_1), d(M_2, M_3) + V^{(2)}(M_3), d(M_2, M_4) + 0) = \min(4 + 4, 1 + 2, 1) = 1, \quad v^{(3)}(M_2) = M_4$$

$$V^{(3)}(M_3) = 2 \quad v^{(3)}(M_3) = M_2$$

Pre $\boxed{k > 3}$ sa funkcie V a v už nemenia, napr. pre $k = 4$ a M_3 dostávame

$$V^{(4)}(M_3) = \min(d(M_3, M_1) + V^{(3)}(M_1), d(M_3, M_2) + V^{(3)}(M_2), d(M_3, M_4) + 0) = \min(2 + 4, 1 + 1, 4) = 2, \quad v^{(4)}(M_3) = M_2$$

Optimálne riešenia pre jednotlivé začiatkové stavy sú:

$$\text{Pre } x_0 = M_3 : \quad \hat{\mathcal{X}} = \{M_3, M_2, M_4\}, \quad \hat{J} = 2, \quad \hat{k} = 2.$$

$$\text{Pre } x_0 = M_2 : \quad \hat{\mathcal{X}} = \{M_2, M_4\}, \quad \hat{J} = 1, \quad \hat{k} = 1.$$

$$\text{Pre } x_0 = M_1 : \quad \hat{\mathcal{X}}_1 = \{M_1, M_3, M_2, M_4\}, \quad \hat{J} = 4, \quad \hat{k} = 3. \\ \hat{\mathcal{X}}_2 = \{M_1, M_4\}, \quad \hat{J} = 4, \quad \hat{k} = 1.$$