Pokroky matematiky, fyziky a astronomie

Dana Glückhaufová; Milan Vlach

Diskrétní úlohy dynamického programování

Pokroky matematiky, fyziky a astronomie, Vol. 13 (1968), No. 4, 201--224

Persistent URL: http://dml.cz/dmlcz/138701

POKROKY MATEMATIKY, FYZIKY A ASTRONOMIE
ROČNÍK XIII — ČÍSLO 4

DISKRÉTNÍ ÚLOHY DYNAMICKÉHO PROGRAMOVÁNÍ

DAGMAR GLÜCKAUFOVÁ, MILAN VLACH, Praha

V souvislosti s prudkým rozvojem automatizace v technice a vlivem matematické formulace řady úloh v různých vědních oborech po druhé světové válce vznikla potřeba řešit značné množství úloh matematického charakteru, které buď nelze (anebo jen s velkými obtížemi) řešit prostředky klasické matematické analýzy. Širokou třídu takových úloh představují úlohy vedoucí k maximalizaci nebo minimalizaci funkcí za určitých omezujících podmínek neboli tzv. optimalizační úlohy.

Pro některé třídy optimalizačních úloh byly vypracovány matematické teorie a metody, které byly již v dostatečné míře prověřeny praktickým použitím v řadě oborů. Sem je možno zařadit např. metody lineárního programování, které umožňují maximalizovat, resp. minimalizovat lineární funkce při lineárních omezeních.

Cílem této práce je seznámit čtenáře se základními myšlenkami a postupy užívanými k řešení optimalizačních úloh vystupujících v souvislosti s potřebou regulace tzv. víceetapových rozhodovacích procesů.[1]) Tyto postupy se označují jako dynamické programování.

Metod dynamického programování lze užít (a také bylo užito) v tak mnoha svým obsahem různorodých úlohách, že je téměř nemožné v kratším textu zachytit v úplnosti celou oblast aplikací. Kromě toho reálná náplň konkrétních úloh může zbytečně komplikovat výklad i pochopení základních myšlenek. Z těchto (a i jiných) důvodů byl zvolen následující postup výkladu.

V prvé části je podána obecná charakteristika procesů, které jsou základem teorie dynamického programování, a jsou v obecném tvaru zformulovány postupy pro řešení příslušných optimalizačních úloh.

Dále jsou na jedné z charakteristických úloh dynamického programování, na tzv. úloze optimálního rozdělení zdrojů, demonstrovány jednotlivé numerické postupy. Tato úloha se studuje na různých úrovních složitosti, což umožňuje ilustrovat většinu nejužívanějších postupů. Celá tato část se zabývá diskrétním deterministickým případem víceetapových rozhodovacích procesů.

---

[1] Oblast použití těchto postupů je však podstatně širší, neboť řadu úloh, které na první pohled nemají s víceetapovým procesem nic společného, lze uměle na etapy rozčlenit a zformulovat je jako úlohy dynamického programování.

Druhá část je věnována případům diskrétních stochastických rozhodovacích procesů. Vzhledem k poměrně značné náročnosti na matematický aparát a vzhledem k menší rozpracovanosti ve srovnání s právě zmíněnými případy nezabýváme se v této práci metodami, které se týkají spojitých deterministických a spojitých stochastických rozhodovacích procesů a tzv. adaptivních procesů.

Pokud jde o dnes již značně rozsáhlou literaturu týkající se dynamického programování a jeho aplikací, nepovažujeme za účelné předložit čtenáři dlouhý seznam prací, nelze-li současně udat jejich alespoň orientační charakteristiku. Proto uvádíme pouze následující krátký výběr většinou dostupných knih, jichž jsme ve větší nebo menší míře použili při psaní tohoto článku a v nichž lze nalézt dostatečně bohaté odkazy.

Elementární výklad metody dynamického programování v případě diskrétních úloh s konečným počtem etap lze nalézt v práci

[1] E. S. Ventcel: *Elementy dinamičeskogo programmirovanija,* Izdatelstvo Nauka Moskva, 1964.

Základní prací pojednávající o dynamickém programování je kniha

[2] R. Bellman: *Dynamic Programming,* Princeton University Press, Princeton, New Jersey, 1957. (Ruský překlad: *R. Bellman: Dinamičeskoje programmirovanije,* Izdatelstvo innostrannoj literatury, Moskva 1960.)

Převážně numerickým otázkám algoritmů pro řešení úloh dynamického programování pro samočinné počítače je věnována práce

[3] R. Bellman, S. Dreyfus: *Applied Dynamic Programming,* Princeton University Press, Princeton, New Jersey 1952. (Ruský překlad: *R. Bellman, S. Drejfus: Prikladnyje zadači dinamičeskogo programmirovanija,* Izdatelstvo Nauka, Moskva 1960.)

Velmi podnětnou knihou pro další rozvoj metod dynamického programování může být práce

[4] R. Bellman: *Adaptive Control Processes: A Guided Tour,* Princeton University Press, Princeton, New Jersey 1961. (Ruský překlad: *R. Bellman: Procesy regulirovanija s adaptacijej,* Izdatelstvo Nauka, Moskva 1964.)

Poměrně méně dostupnou je práce

[5] G. Hadley: *Nonlinear and Dynamic Programming,* London 1964.

Souvislosti dynamického programování s markovskými procesy je věnována práce

[6] R. Howard: *Dynamic Programming and Markov Processes,* Cambridge, Mass.: Technology Press, 1960. (Ruský překlad: *R. Howard: Dinamičeskoje programmirovanije u markovskije procesy,* Sovetskoje radijo, Moskva 1964.)

[7] Sanford M. Roberts: *Dynamic Programming in Chemical Engineering and Process Control,* Academic Press, New York, London 1964. (Ruský překlad: *Dinamičeskoje programmirovanije v processach chimičeskoj technologiji i metody upravlenija,* Moskva 1965.)

[8] A. Ter-Manuelianc: *Dynamické programování v hospodářské praxi.* Ekonomicko-matematický obzor, ročník *3*, 1966.

1. Víceetapové rozhodovací procesy; princip optimálnosti

Uvažujme určitou reálnou soustavu, která je idealizována do té míry, že její stav lze v okamžiku t popsat vektorem

$$\mathbf{p}(t) = (p^{(1)}(t), p^{(2)}(t), \ldots, p^{(k)}(t)),$$

který nazveme stavovým vektorem. Proměnné $p^{(1)}(t), p^{(2)}(t), \ldots, p^{(k)}(t)$ nazveme stavovými proměnnými. Pojem soustavy nebudeme zatím blíže specifikovat, abychom mohli zachytit libovolný reálný proces, ať již jde o proces fyzikální, biologický či ekonomický. Např. v určité ekonomické soustavě mohou stavové proměnné vyjadřovat výrobní kapacity nebo zásoby vzájemně závislých odvětví.

Budeme se dále zabývat regulovatelnými soustavami, tj. soustavami, jejichž stav se může měnit v závislosti na čase, přičemž můžeme nějakým způsobem změny stavu ovlivňovat.

Předpokládejme nejprve, že jde o diskrétní proces, tj. že změny stavu soustavy probíhají pouze v diskrétních časových okamžicích

$$t_1, t_2, \ldots, t_n, \ldots, (t_1 < t_2 < \ldots < t_n < \ldots).$$

Nechť v těchto časových okamžicích je stav soustavy popsán postupně vektory

$$\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_n, \ldots,$$

kde

$$\mathbf{p}_i = \mathbf{p}(t_i) = (p^{(1)}(t_i), p^{(2)}(t_i), \ldots, p^{(k)}(t_i)).$$

Vektory $\mathbf{p}_i$ mohou náležet pouze určité přípustné množině D, jež je uvažovanou soustavou určena. Je-li dán stav $\mathbf{p}_1$ uvažované soustavy v počátečním okamžiku t_1, lze chování soustavy charakterizovat jistou transformací T množiny D do sebe. Regulovatelnost soustavy můžeme chápat tak, že místo jediné transformace T máme k dispozici určitou množinu přípustných transformací a regulování soustavy záleží ve výběru jediné transformace v každém z uvažovaných okamžiků. Tento výběr lze chápat jako provedení jistého rozhodnutí q; nový stav je tedy funkcí stavu předchozího a provedeného rozhodnutí:

$$(1.1)\quad \mathbf{p}_2 = T(\mathbf{p}_1, q_1); \mathbf{p}_3 = T(\mathbf{p}_2, q_2); \ldots \mathbf{p}_n = T(p_{n-1}, q_{n-1}); \mathbf{p}_{n+1} = T(\mathbf{p}_n, q_n),$$

kde q_i značí rozhodnutí (výběr transformace) v okamžiku t_i.

Poněvadž rozhodnutí uskutečňujeme postupně, nejprve v okamžiku t_1, pak v okamžiku t_2 atd., nazýváme proces popsaného typu víceetapovým rozhodovacím procesem. Přesněji řečeno, jde o víceetapový rozhodovací proces diskrétního deterministického typu, neboť volba určitého rozhodnutí má za následek změnu stavu na jediný stav určený předchozím stavem a volbou rozhodnutí. O transformacích $p_{i+1} = T(p_i, q_i)$, $i = 1, 2, \ldots$ předpokládejme, že nezávisí explicitně na stavech, kterými soustava v okamžicích $t_1, t_2, \ldots, t_{i-1}$ prošla, nýbrž že závisí toliko na stavu p_i v předchozím kroku (který ovšem implicitně je výsledkem historie soustavy) a na rozhodnutí q_i v předchozím kroku. Tato podmínka na transformaci T se nazývá podmínka markovská.

Další otázkou je, podle jakých kritérií volíme v každém z uvažovaných časových okamžiků rozhodnutí q_i neboli jak provádíme jednotlivá rozhodnutí. Uvažujme pro jednoduchost n-etapový proces, tj. předpokládejme, že nás zajímá chování soustavy pouze v prvních n časových okamžicích $t_1, \ldots, t_n$. Takovému n-etapovému procesu přiřadíme určitou funkci

$$F(p_1, \ldots, p_n;\ q_1, \ldots, q_n),$$

kterou nazveme účelovou funkcí; budeme předpokládat, že cílem regulace procesu je nalézt rozhodnutí $q_1, q_2, \ldots, q_n$ tak, abychom dosáhli maximální hodnoty funkce F. Předpokládáme, že jednotlivá rozhodnutí q_i jsou charakterizována čísly (popřípadě uspořádanými k-ticemi čísel), která značíme rovněž symboly q_i.

Je-li dán stavový vektor p_1, pak vzhledem ke vztahům (1.1) lze vyjádřit proměnné p_i $(i > 1)$ pomocí $q_1, q_2, \ldots, q_{i-1}$ a maximalizovat vzhledem k rozhodnutím $q_1, q_2, \ldots$ $\ldots, q_n$ funkci

$$G(q_1, \ldots, q_n) = F(p_1, T(p_1, q_1,), \quad T(T(p_1, q_1), q_2), \ldots, q_1, \ldots, q_n).$$

Dostáváme tak zvláštní případ maximalizace funkce n proměnných. Funkce G je málokdy definována tak, aby bylo možno řešit tuto úlohu prostředky klasické analýzy, např. řešením soustavy rovnic

$$\frac{\partial G}{\partial q_i} = 0, \quad i = 1, 2, \ldots, n.$$

Dříve než ukážeme řešení uvedené úlohy metodou dynamického programování, musíme učinit další předpoklady o funkci F. Základním požadavkem, který budeme klást na funkci F, je, aby byla definována pro každé n, tj. aby byla dána posloupnost

$$F(p_1; q_1), \quad F(p_1, p_2;\ q_1, q_2), \ldots$$

a aby tuto definici bylo možno formulovat rekurentně tak, že $F(p_1, p_2, \ldots, p_n;$ $q_1, q_2, \ldots, q_n)$ lze definovat pomocí $F(p_1, p_2, \ldots, p_{n-1}; q_1, q_2, \ldots, q_{n-1})$ a p_n, q_n; tato vlastnost, podobná markovské podmínce na transformaci $p_{i+1} = T(p_i, q_i)$, se také nazývá markovská vlastnost. Tuto vlastnost mají zejména separabilní funkce, tj. funkce, které lze uvést na tvar

$$F(p_1, p_2, \ldots, p_n;\ q_1, q_2, \ldots, q_n) = \sum_{i=1}^{n} g_i(p_i, q_i). \tag{1.2}$$

Ukazuje se, že většině regulovatelných procesů ekonomického charakteru lze rozumným způsobem přiřadit separabilní účelovou funkci.

Ukažme si nyní přístup dynamického programování v případě maximalizace funkce (1, 2). Předpokládejme, že maximum existuje, a abychom zpočátku výklad zbytečně nekomplikovali, předpokládejme ještě, že veličiny q_i mohou nabývat pouze konečně mnoha různých hodnot a že funkce g_i jsou omezené.

Místo abychom řešili tuto úlohu pro jisté dané $\boldsymbol{p}_1$ a n, budeme řešit celou množinu podobných úloh. Abychom pochopili, o jakou množinu úloh jde, stačí si uvědomit, že hodnota maxima funkce (1.2) závisí na počáteční hodnotě stavového vektoru $\boldsymbol{p}_1$ a na počtu etap příslušného rozhodovacího procesu, tj. na hodnotě n.[1] Označíme proto příslušné maximum symbolem $f_n(\boldsymbol{p}_1)$, tj.

$$(1.3) \qquad f_n(\boldsymbol{p}_1) = \max_{q_1,\dots,q_n} [g_1(\boldsymbol{p}_1, q_1) + g_2(\boldsymbol{p}_2, q_2) + \dots + g_n(\boldsymbol{p}_n, q_n)] .$$

První člen posloupnosti funkcí $\{f_n(\boldsymbol{p}_n)\}$ lze určit maximalizací funkce jedné proměnné:

$$(1.4) \qquad f_1(\boldsymbol{p}_1) = \max_{q_1} g_1(\boldsymbol{p}_1, q_1) .$$

Najdeme nyní rekurentní vztah, pomocí kterého lze vytvářet posloupnost $\{f_n(\boldsymbol{p}_1)\}$,

$$(1.5) \qquad f_n(\boldsymbol{p}_1) = \max_{q_1} \max_{q_2} \dots \max_{q_n} [g_1(\boldsymbol{p}_1, q_1) + \dots + g_n(\boldsymbol{p}_n, q_n)] =$$
$$= \max_{q_1} \{g_1(\boldsymbol{p}_1, q_1) + \max_{q_2} \max_{q_3} \dots \max_{q_n} [g_2(\boldsymbol{p}_2, q_2) + \dots + g_n(\boldsymbol{p}_n, q_n)]\} .$$

Podle našeho označení je

$$\max_{q_2} \max_{q_3} \dots \max_{q_n} [g_2(\boldsymbol{p}_2, q_2) + \dots + g_n(\boldsymbol{p}_n, q_n)] = f_{n-1}(\boldsymbol{p}_2) ,$$

neboť jde zřejmě o proces s počátečním stavem $\boldsymbol{p}_2$, který má $n - 1$ etap. Můžeme tedy psát

$$(1.6) \qquad f_n(\boldsymbol{p}_1) = \max_{q_1} [g_1(\boldsymbol{p}_1, q_1) + f_{n-1}(\boldsymbol{p}_2)] ,$$

kde $\boldsymbol{p}_2 = T(\boldsymbol{p}_1, q_1)$ čili

$$(1.7) \qquad f_n(\boldsymbol{p}_1) = \max_{q_1} [g_1(\boldsymbol{p}_1, q_1) + f_{n-1}(T(\boldsymbol{p}_1, q_1))] .$$

Tento rekurentní vztah lze získat také na základě tzv. principu optimálnosti, který vyjadřuje základní myšlenku dynamického programování. Pro snazší formulaci tohoto principu zavedeme tato označení: Posloupnost přípustných rozhodnutí $\{q_1, \dots, q_n\}$ nazveme strategií. Strategii vedoucí k maximální hodnotě účelové funkce nazveme optimální strategií. Je-li účelová funkce markovského typu, vyznačuje se

[1]) Považujeme tedy $\boldsymbol{p}_1$ a n od tohoto okamžiku za proměnné.

optimální strategie $\{q_1, \ldots, q_n\}$ při n-etapovém procesu vlastností, která byla nazvána *princip optimálnosti*:

Je-li $\{q_1, \ldots, q_n\}$ optimální strategie n-etapového procesu s počátečním stavem $\boldsymbol{p}_1$, potom posloupnost rozhodnutí $\{q_2, \ldots, q_n\}$ tvoří optimální strategii $(n - 1)$-etapového procesu s počátečním stavem $\boldsymbol{p}_2$ (do něhož se soustava dostala ze stavu $\boldsymbol{p}_1$ provedením rozhodnutí q_1).

Na základě principu optimálnosti můžeme získat rekurentní vztah (1.7) pouze logickou úvahou. Při libovolném stavu $\boldsymbol{p}_1$ a při libovolném počátečním rozhodnutí q_1 musí podle principu optimálnosti (funkce (1.2) má markovskou vlastnost) být maximální hodnota účelové funkce (1.2) pro $(n - 1)$-etapový proces rovna hodnotě $f_{n-1}(T(\boldsymbol{p}_1, q_1))$. Pro libovolné počáteční rozhodnutí q_1 je hodnota účelové funkce (1.2) v případě n-etapového procesu s počátečním stavem p_1 rovna:

$$q_1(\boldsymbol{p}_1, q_1) + f_{n-1}(T(\boldsymbol{p}_1, q_1)) .$$

Počáteční rozhodnutí musí být vybráno tak, aby účelová funkce n-etapového procesu nabývala maximální hodnoty, tj.

$$(1.8) \qquad f_n(\boldsymbol{p}_1) = \max_{q_1} [g_1(\boldsymbol{p}_1, q_1) + f_{n-1}(T(\boldsymbol{p}_1, q_1))] .$$

V následujícím odstavci přejdeme ke konkrétním výpočetním postupům založeným na metodě dynamického programování.

2. Optimální rozdělení zdrojů jako úloha dynamického programování

Předpokládejme, že máme k dispozici jistá omezená množství x, y, z různých ekonomických zdrojů. Úvahu provedeme pro tři zdroje; je však zřejmé, že platí pro libovolný počet zdrojů. Ekonomickým zdrojem může být např. surovina, pracovní síly, stroje, investice apod. Každého ze zdrojů lze užít několika různými způsoby — řekněme n způsoby. Výsledný efekt použití části daného množství x, resp. y, resp. z kterýmkoliv z možných způsobů je znám a nezávisí na množstvích, která byla použita ostatními způsoby. Je otázka, jak velké části jednotlivých zdrojů máme přidělit pro různá možná použití, chceme-li získat maximální celkový efekt (přitom zdrojů se ještě nepoužívá).

Označíme-li množství zdrojů použitá i-tým způsobem $(i = 1, 2, \ldots, n)$ x_i, resp. y_i, resp. z_i a odpovídající efekt $g_i(x_i, y_i, z_i)$, kde g_i značí funkci vyjadřující výsledky i-tého způsobu použití zdrojů, a je-li celkový výsledek vyjádřen součtem dílčích výsledků (předpokládáme, že všechny dílčí výsledky jsou vyjádřeny ve stejných jednotkách), vede uvažovaný problém k maximalizaci funkce

$$(1.9) \qquad F(x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n, z_1, z_2, \ldots, z_n) = \sum_{i=1}^{n} g_i(x_i, y_i, z_i)$$

na množině určené podmínkami

$$\sum_{i=1}^{n} x_i \leqq x\,, \quad x_i \geqq 0\,; \quad \sum_{i=1}^{n} y_i \leqq y\,, \quad y_i \geqq 0\,; \quad \sum_{i=1}^{n} z_i \leqq z\,, \quad z_i \geqq 0\,. \tag{1.10}$$

Předpokládejme, že funkce g_i jsou neklesající v jednotlivých proměnných a že $g_i(0, 0, 0) = 0$ [1]).

Povšimněme si, že v případě lineárních funkcí g_i jde o úlohu lineárního programování. Ukažme, jak lze uvedenou úlohu chápat jako úlohu dynamického programování.

Předpokládejme, že jednotlivá množství rozdělujeme pro různá možná užití postupně; nejprve přidělíme zdroje pro první způsob, pak pro druhý atd. Za vektor $\boldsymbol{p}_1$ popisující počáteční stav soustavy můžeme považovat vektor udávající počáteční množství x, y, z, tj., vektor

$$\boldsymbol{p}_1 = (x, y, z)\,.$$

Počátečním rozhodnutím q_1 bude vektor částí jednotlivých zdrojů přidělených pro první způsob použití, tj. volba hodnot x_1, y_1, z_1. Připustnost rozhodnutí je stanovena podmínkami (1.10), tj. hodnoty x_1, y_1, z_1 musí splňovat podmínky $0 \leqq x_1 \leqq x$, $0 \leqq y_1 \leqq y$; $0 \leqq z_1 \leqq z$. Toto rozhodnutí má za následek změnu počátečního stavu popsaného vektorem $\boldsymbol{p}_1$ na stav charakterizovaný vektorem

$$\boldsymbol{p}_2 = (x - x_1, y - y_1, z - z_1)\,.$$

Další rozhodnutí q_2 záleží ve volbě hodnot x_2, y_2, z_2 tak, aby byly splněny podmínky

$$0 \leqq x_2 \leqq x - x_1\,; \quad 0 \leqq y_2 \leqq y - y_1\,; \quad 0 \leqq z_2 \leqq z - z_1\,.$$

Výsledkem rozhodnutí q_2 je změna stavu soustavy na stav popsaný vektorem

$$\boldsymbol{p}_3 = (x - x_1 - x_2,\ y - y_1 - y_2,\ z - z_1 - z_2)\,.$$

Konečný stav bude popsán vektorem

$$\boldsymbol{p}_{n+1} = \left(x - \sum_{i=1}^{n} x_i,\ y - \sum_{i=1}^{n} y_i,\ z - \sum_{i=1}^{n} z_i\right),$$

kde x_i, y_i, z_i splňují podmínky

$$0 \leqq x_i \leqq x - \sum_{k=1}^{i-1} x_k\,; \quad 0 \leqq y_i \leqq y - \sum_{k=1}^{i-1} y_k\,; \quad 0 \leqq z_i \leqq z - \sum_{k=1}^{i-1} z_k\,.$$

Maximální hodnota funkce (1.9) závisí pouze na počátečním stavu $\boldsymbol{p}_1 = (x, y, z)$

---

[1]) U ekonomicky rozumných úloh bývá často splněna i podmínka, že se stoupajícím využitím každého zdroje přírůstky funkce g_i nevzrůstají.

a na počtu kroků (různých způsobů užití jednotlivých zdrojů) n. Píšeme tedy

$$f_n(\mathbf{p}_1) = f_n(x, y, z) = \max_{q_1, q_2, \dots, q_n} F(x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_n, z_1, z_2, \dots, z_n) =$$
$$= \max_{q_1, q_2, \dots, q_n} \sum_{i=1}^{n} q_i(x_i, y_i, z_i),$$

kde q_i představuje volbu hodnot x_i, y_i, z_i. Na základě principu optimálnosti dostáváme pro $n \geqq 2$ tento rekurentní vztah pro posloupnosti $\{f_n(\mathbf{p}_1)\}$:

$$f_n(x, y, z) = \max_{0 \leqq x_n \leqq x} \max_{0 \leqq y_n \leqq y} \max_{0 \leqq z_n \leqq z} [q_n(x_n, y_n, z_n) +$$
$$+ f_{n-1}(x - x_n, y - y_n, z - z_n)$$

s počáteční podmínkou

$$f_1(x, y, z) = \max_{q_1} g_1(x_1, y_1, z_1) = g_1(x, y, z).$$ [1])

Užití různých postupů, umožňujících na základě získaného rekurentního vztahu získat numerické řešení, ukážeme na řadě speciálních případů uvedené úlohy.

Začněme poměrně jednoduchým případem jednoho zdroje, přičemž hodnoty x a x_i mohou být pouze celočíselné; jde tedy o maximalizaci funkce

(1.11) $$F(x_1, x_2, \dots, x_n) = \sum_{i=1}^{n} g_i(x_i)$$

na množině určené podmínkami

(1.12) $$\sum_{i=1}^{n} x_i \leqq x \quad x_i \in \{0, 1, 2, \dots, x\}, \quad i = 1, 2, \dots, n.$$

O funkcích $g_i(x_i)$ budeme předpokládat, že jsou neklesající a že $g_i(0) = 0$. V tomto případě je

(1.12') $$f_1(x) = \max_{x_1 \in \{0, 1, \dots, x\}} g_1(x_1) = g_1(x),$$
$$f_n(x) = \max_{x_n \in \{0, 1, \dots, x\}} [g_n(x_n) + f_{n-1}(x - x_n)], \quad n \geqq 2.$$

Abychom získali číselné řešení, tabelujeme nejprve funkci $f_1(x)$, pak $f_2(x)$ atd. až funkci $f_n(x)$ a současně zaznamenáváme ty hodnoty (označme je $\bar{x}_i$) proměnných x_i, které dávají odpovídající maxima $f_i(x)$; tyto hodnoty ovšem závisí na veličině x, tj. tabelujeme současně funkce $\bar{x}_1(x), \bar{x}_2(x), \dots, \bar{x}_n(x)$. Chceme-li nyní určit maximální hodnotu účelové funkce a jí odpovídající optimální rozdělení určitého množství x^*, nalezneme nejprve v získaných tabulkách maximální hodnotu $f_n(x^*)$ a jí odpovídající

[1]) Je zřejmé, že očíslování jednotlivých zdrojů je zcela libovolné a je proto možné na rozdíl od postupu na str. 204 a následujících začít rozhodování od n-tého zdroje počínajíc.

maximalizující hodnotu $\bar{x}_n(x^*)$; ostatní proměnné pak musí vyhovovat podmínce $\sum_{i=1}^{n-1} x_i \leqq x^* - \bar{x}_n(x^*)$ a v tabulce funkce $\bar{x}_{n-1}$ tedy najdeme hodnotu $\bar{x}_{n-1}(x^* - \bar{x}_n(x^*))$. Stejně postupujeme dále až v tabulce $\bar{x}_1$ najdeme hodnotu $x_1(x^* - \sum_{k=0}^{n-2} x_{n-k}(x^*))$. Takto nalezené hodnoty udávají optimální rozdělení množství x^*.

Ilustrujme pro názornost tento postup konkrétním příkladem pro $n = 3$ a $x = 0, 1, \ldots, 10$, je-li účelová funkce

$$F(x_1, x_2, x_3) = g_1(x_1) + g_2(x_2) + g_3(x_3)$$

dána těmito funkcemi:

$x_1 = x_2 = x_3$	$g_1(x_1)$	$g_2(x_2)$	$g_3(x_3)$
0	0	0	0
1	3	2	4
2	10	10	10
3	22	18	22
4	28	26	31
5	34	35	37
6	45	42	40
7	58	48	40
8	64	55	40
9	65	59	40
10	66	60	40

Tabulky udávající hodnoty funkcí

$$f_1(x) = \max_{x_1 \in \{0,1,\ldots,x\}} g_1(x_1), \quad \bar{x}_1(x),$$

$$f_2(x) = \max_{x_2 \in \{0,1,\ldots,x\}} [g_2(x_2) + f_1(x - x_2)], \quad \bar{x}_2(x),$$

$$f_3(x) = \max_{x_3 \in \{0,1,\ldots,x\}} [g_3(x_3) + f_2(x - x_3)], \quad x_3(x),$$

jsou zachyceny v tabulce na následující straně.

Pro některé hodnoty x lze maximální hodnoty $f_2(x)$, resp. $f_3(x)$ dosáhnout několika možnými volbami hodnot proměnné x_2, resp. x_3. Např. pro $x = 2$ dostáváme:

$$\begin{aligned} &\text{je-li} \quad x_2 = 0, \quad \text{je} \quad g_2(0) + f_1(2) = 0 + 10 = 10; \\ &\text{je-li} \quad x_2 = 1, \quad \text{je} \quad g_2(1) + f_1(1) = 2 + 3 = 5; \\ &\text{je-li} \quad x_2 = 2, \quad \text{je} \quad g_2(2) + f_1(0) = 10 + 0 = 10. \end{aligned}$$

x	$f_1(x)$	$\bar{x}_1(x)$	$f_2(x)$	$\bar{x}_2(x)$	$f_3(x)$	$\bar{x}_3(x)$
0	0	0	0	0	0	0
1	3	1	3	0	4	1
2	10	2	10	0; 2	10	0; 2
3	22	3	22	0	22	0; 3
4	28	4	28	0	31	4
5	34	5	35	5	37	5
6	45	6	45	0	45	0
7	58	7	58	0	58	0
8	64	8	64	0	64	0
9	65	9	68	2	68	0; 1; 2
10	66	10	76	3	80	3

Maximální hodnoty $f_2(2) = 10$ lze dosáhnout volbou $x_2 = 0$ nebo volbou $x_2 = 2$. Tyto možnosti jsou v tabulce jasně zachyceny.

Z uvedené tabulky již snadno určíme maximální hodnotu účelové funkce i odpovídající optimální řešení pro libovolnou z uvažovaných hodnot x. Všimněme si např. případů $x = 9$, $x = 3$ a $x = 2$, ve kterých existuje několik alternativních optimálních řešení. Tak pro $x = 9$ je $f_3(x) = 68$ a volíme-li

$$\bar{x}_3(9) = 0\,, \quad \text{je} \quad \bar{x}_2(9 - \bar{x}_3(9)) = \bar{x}_2(9) = 2$$

a

$$\bar{x}_1(9 - \bar{x}_3(9) - \bar{x}_2(9)) = \bar{x}_1(7) = 7\,.$$

Je tedy optimální řešení toto:

$$(x_1, x_2, x_3) = (7, 2, 0)\,.$$

Volíme-li $\bar{x}_3(9) = 1$ nebo $\bar{x}_3(9) = 2$, dostaneme rovněž optimální řešení

$$(x_1, x_2, x_3) = (8, 0, 1) \quad \text{nebo} \quad (x_1, x_2, x_3) = (7, 0, 2)\,.$$

Pro $x = 3$ je $f_3(3) = 22$ s odpovídajícími optimálními řešeními

$$(x_1, x_2, x_3) = \begin{cases} (3, 0, 0) \\ (0, 0, 3) \end{cases}$$

a pro $x = 2$ je $f_3(2) = 10$ s optimálními řešeními

$$(x_1, x_2, x_3) = \begin{cases} (2, 0, 0) \\ (0, 2, 0) \\ (0, 0, 2) \end{cases}$$

Pro představu o efektivnosti popsaného způsobu řešení uveďme několik údajů. Kdybychom maximum funkce (1.11) určovali srovnáváním všech možných hodnot, kterých tato funkce nabývá, museli bychom její hodnoty vyčíslit nejméně v $\binom{n + x - 1}{x}$ bodech, neboť toto číslo udává dolní hranici počtu elementů množiny určené podmínkami (1.12). Užitím metody dynamického programování je počet srovnávaných hodnot podstatně nižší, neboť je roven číslu

$$x\left[n + \frac{(n-1)(x+1)}{2}\right] + n\,.$$

Pro $n = 5$ a $x = 20$ je:

$$\binom{n + x - 1}{x} = 10\,626\,,$$

$$x\left[n + \frac{(n-1)(x+1)}{2}\right] + n = 945\,.$$

Rozdíl mezi těmito čísly ještě prudce stoupá s rostoucím n.

Uvažujme nyní další příklad, který lze v přeneseném slova smyslu také chápat jako zvláštní případ uvedené obecné úlohy nazvané optimální rozdělení zdrojů.

Máme naložit určitý dopravní prostředek předměty různých typů. Nechť n je počet různých typů a nechť

v_i značí cenu jednoho předmětu i-tého typu,
w_i značí váhu jednoho předmětu i-tého typu,
x_i značí počet naložených předmětů i-tého typu a konečně
z značí váhovou kapacitu uvažovaného dopravního prostředku.

Je otázka, jak volit čísla x_i, abychom získali náklad co největší ceny. Snadno nahlédneme, že jde o maximalizaci funkce

$$F(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} x_i v_i$$

na množině určené podmínkami

$$\sum x_i w_i \leqq z\,, \quad x_i \text{ celé a nezáporné}\,;$$

kde $v_i > 0$; $w_i > 0$.

Zanedbáme-li požadavek celočíselnosti proměnných x_i, je řešení úlohy snadné. Najdeme index i, pro který je hodnota výrazu v_i/w_i největší. Je-li to index i_1, pak naložíme pouze předměty typu i_1 a získáme náklad maximální ceny $z/w_{i_1} \cdot v_{i_1}$. Kdybychom pro respektování požadavku celočíselnosti proměnných x_i zaokrouhlili na celá čísla řešení získaná bez požadavku celočíselnosti, mohli bychom získat náklad, který nemusí být optimální.

Nechť např.

$$\begin{aligned} n &= 3\,; & z &= 100\,; & & \\ w_1 &= 40\,; & w_2 &= 45\,; & w_3 &= 60\,; \\ v_1 &= 20\,; & v_2 &= 75\,; & v_3 &= 102\,. \end{aligned}$$

Bez požadavku celočíselnosti získáme řešení:

$$x_1 = 0\,; \quad x_2 = 0\,; \quad x_3 = \frac{100}{60}\,,$$

neboť

$$\frac{v_1}{w_1} = \frac{20}{40} = 0{,}50\,; \quad \frac{v_2}{w_2} = \frac{75}{45} \doteq 1{,}67\,; \quad \frac{v_3}{w_3} = \frac{102}{60} = 1{,}70\,.$$

Získáme tak náklad ceny

$$F\left(0, 0, \frac{100}{60}\right) = 0 \times 20 + 0 \times 75 + 102 \times \frac{100}{60} = 170\,.$$

Kdybychom zaokrouhlili $\frac{100}{60} \doteq 1{,}66$ na 1, tj. kdybychom naložili jeden předmět třetího typu, můžeme ještě naložit předměty o váze $100 - 60 = 40$. Můžeme tedy ještě naložit jeden předmět prvního typu; získáme tak náklad $x_1 = 1$, $x_2 = 0$, $x_3 = 1$ o ceně

$$F(1, 0, 1) = 1 \times 20 + 0 \times 75 + 1 \times 102 = 122\,.$$

Tento náklad však není optimální, protože např. náklad

$$x_1 = 0\,, \quad x_2 = 2\,, \quad x_3 = 0$$

má větší cenu:

$$F(0, 2, 0) = 0 \times 20 + 2 \times 75 + 0 \times 102 = 150\,.$$

Tento náklad je vzhledem k danému kritériu optimální, třebaže proti prvnímu nákladu nevyužívá plně váhovou kapacitu daného dopravního prostředku.

Při řešení metodou dynamického programování užijeme opět posloupnosti funkcí $f_n(z)$ daných rekurentním vztahem:

$$f_n(z) = \max_{x_n \in \{0, 1, \ldots, z/w_n\}} \{x_n v_n + f_{n-1}(z - x_n w_n)\}\,, \quad n \geqq 2\,,$$

$$f_1(z) = \left[\frac{z}{w_1}\right] . \, v_1\,,$$

kde symbol $[x]$ značí největší celé číslo nepřevyšující x.

Jde-li např. o dva typy předmětů, pro které

$$w_1 = 20\,, \qquad w_2 = 18\,,$$
$$v_1 = 72\,, \qquad v_2 = 60\,,$$

a uvažujeme-li z z intervalu $\langle 0,100 \rangle$, dostaneme postupně tabulky

z	$f_1(z)$	$\bar{x}_1(z)$
0—19	0	0
20—39	72	1
40—59	144	2
60—79	216	3
80—99	288	4
100	360	5

z	$f_2(z)$	$x_2(z)$
0—17	0	0
18—19	60	1
20—35	72	0
36—37	120	2
38—39	132	1
40—53	144	0
54—55	180	3
56—57	192	2
58—59	204	1
60—71	216	0
72—73	240	4
74—75	252	3
76—77	264	2
78—79	276	1
80—89	288	0
90—91	300	5
92—93	312	4
94—95	324	3
96—97	336	2
98—99	348	1
100	360	0

Např. pro dopravní prostředek o kapacitě 95 příslušných váhových jednotek je optimální náklad tvořen dvěma předměty prvního typu a třemi předměty druhého typu o celkové váze 94 jednotek a o ceně $f_2(95) = 324$ příslušných peněžních jednotek.

3. Numerické otázky dynamického programování

Vraťme se na okamžik k úloze záležející v maximalizaci funkce

$$F(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} g_i(x_i)$$

na množině určené podmínkami

$$\sum_{i=1}^{n} x_i = x\,; \quad x_i \geqq 0\,,$$

kde pro funkce g_i platí $g_i(0) = 0$.

Na rozdíl od dříve formulované úlohy upustili jsme nyní od požadavku celočíselnosti. Povšimneme si problémů, které tím vznikají. Označíme-li opět

$$f_n(x) = \max_{x_1, x_2, \ldots, x_n} F(x_1, x_2, \ldots, x_n)\,,$$

kde $\sum_{i=1}^{n} x_i = x$, $x_i \geqq 0$, dostaneme, na základě principu optimálnosti, rekurentní vztah

$$f_n(x) = \max_{0 \leqq x_n \leqq x} [g_n(x_n) + f_{n-1}(x - x_n)]\,; \quad n \geqq 2\,, \quad x \geqq 0\,,$$

$$f_1(x) = g_1(x)\,; \quad x \geqq 0\,.$$

Posloupnost funkcí $f_n(x)$ a odpovídajících optimálních hodnot $\bar{x}_i(x)$ zpravidla nelze vyjádřit analyticky a musíme proto přistoupit k tabelaci a interpolaci. Základní postup, který lze v mnoha směrech zdokonalit, je tento:

Pro zadání hodnot funkce $f_n(x)$ na intervalu $\langle 0, x_0 \rangle$ užijeme určitým způsobem vybrané uzlové body, např. určíme hodnoty v bodech

$$x = 0, \Delta, 2\Delta, \ldots, R\Delta = x_0$$

a každou funkci posloupnosti $\{f_n(x)\}$ tabelujeme pouze v těchto bodech. Hodnoty $f_n(x)$ pro x různá od uzlových bodů určujeme interpolací. Jaké interpolace užijeme, záleží na požadované přesnosti. Při stanovení tabulek pro $f_n(x)$ a $\bar{x}_n(x)$ postupujeme tedy takto:

Nejprve určíme

$$f_1(k\Delta) = g_1(k\Delta)\,, \quad k = 0, 1, \ldots, R\,,$$

potom

$$f_2(x) = \max_{k \in \{0, 1, \ldots, R\}} [g_2(k\Delta) + f_1(x - k\Delta)]\,,$$

kde x může nabývat pouze hodnot $0, \Delta, 2\Delta, \ldots, R\Delta$. Současně určujeme i ty hodnoty proměnné x_2, které dávají maximální hodnoty $f_2(x)$. Je dobré si uvědomit, že popsaný postup představuje jen jisté schéma, které lze zlepšit, známe-li konkrétní vlastnosti funkce $F(x_1, x_2, \ldots, x_n)$; kromě toho jsme se ještě nezmínili o některých nutných předpokladech. Např. interpolace lze užít v případě, že jde o spojité funkce. Jsou-li funkce g_i $(i = 1, 2, \ldots, n)$ spojité, dá se dokázat, že i funkce $f_n(x)$ jsou spojité, avšak funkce $x_n(x)$ již být spojité nemusí.

Jestliže funkce $f_n(x)$ mají spojité derivace až do jistého řádu, lze s úspěchem použít metody aproximace polynomy, která dovoluje uspokojivě překonat řadu obtíží, např. potíže vznikající při větším množství tabelovaných hodnot v důsledku omezené kapacity paměti samočinných počítačů. Základní myšlenka této metody záleží v aproximaci funkcí $f_n(x)$ lineárními kombinacemi funkcí určitého úplného systému funkcí, takže místo tabulky funkce $f_n(x)$ uchováváme jen koeficienty jejího rozvoje pomocí funkcí uvedeného systému.

Dosud se uvedené příklady týkaly soustav, jejichž stav byl popsán jedinou stavovou proměnnou $\boldsymbol{p}_1(x)$. Nyní přejdeme k příkladům, ve kterých je stav soustavy popsán stavovým vektorem o více stavových proměnných, např. vektorem $\boldsymbol{p} = (x, y)$.

Analogií předchozí úlohy bude v tomto případě úloha s dvěma zdroji x, y, tj. úloha maximalizovat funkci

$$F(x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n) = \sum_{i=1}^{n} g_i(x_i, y_i) \tag{1.13}$$

na množině S určené podmínkami

$$\sum_{i=1}^{n} x_i = x \; ; \quad x_i \geqq 0 \,, \qquad \sum_{i=1}^{n} y_i = y \; ; \quad y_i \geqq 0 \,. \tag{1.14}$$

Na počátku této kapitoly jsme ukázali, že na základě principu optimálnosti získáme rekurentní vztah:

$$f_n(x, y) = \max_{0 \leqq x_n \leqq x} \; \max_{0 \leqq y_n \leqq y} \left[g_n(x_n, y_n) + f_{n-1}(x - x_n, y - y_n)\right], \quad n \geqq 2 \,,$$

$$f_1(x, y) = g_1(x, y) \,,$$

kde

$$f_n(x, y) = \max_{S} F(x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n) \; ;$$

množina S je určena podmínkami (1.14).

Tohoto rekurentního vztahu lze užít k numerickému řešení v zásadě stejným způsobem jako ve výše uvedených příkladech. Říkáme v zásadě, neboť při tom narážíme na řadu praktických potíží vyvolaných omezenou kapacitou paměti a omezenou rychlostí samočinného počítače. Stačí si uvědomit, že jde o tabelaci a interpolaci funkcí dvou proměnných. Tabelujeme-li funkci $f_n(x, y)$ v oblasti $0 \leqq x \leqq M$, $0 \leqq y \leqq M$ pro body $x = k\Delta$, $y = l\Delta$, $k, l = 1, 2, \ldots, [M/\Delta]$, musíme určit funkční hodnoty v $([M/\Delta] + 1)^2$ bodech místo v $([M/\Delta] + 1)$ bodech, jak tomu bylo v jednorozměrném případě. Tyto okolnosti mají za následek, že v případě většího počtu stavových proměnných leží přímá aplikace takového postupu za hranicemi dnešních možností samočinných počítačů. Proto jsou velmi důležité různé metody umožňující alespoň zčásti tyto obtíže obejít. Ukážeme si na uvedeném příkladu dvě z nich:

a) *Metoda Lagrangeových multiplikátorů*

Vraťme se tedy k maximalizaci funkce (1,13) na množině určené podmínkami (1,14). Nahradíme tuto úlohu úlohou maximalizovat funkci

$$F = \sum_{i=1}^{n} g_i(x_i, y_i) - \lambda \sum_{i=1}^{n} y_i , \tag{1.15}$$

(přičemž λ považujeme za pevné a nezáporné) na množině určené podmínkami

$$\sum_{i=1}^{n} x_i = x ; \quad x_i \geqq 0 ; \quad y_i \geqq 0 .$$

Definujme pomocnou funkci

$$h_i(x_i, \lambda) = \max_{y_i \geqq 0} [g_i(x_i, y_i) - \lambda y_i] . \tag{1.16}$$

Aby maximum bylo konečné předpokládejme, že $\lim\limits_{y_i \to \infty} [g_i(x_i, y_i)]/y_i = 0$. Při pevném λ jde pak o úlohu maximalizovat funkci

$$h_1(x_1) + h_2(x_2) + \ldots + h_n(x_n)$$

při podmínkách

$$x_1 + x_2 + \ldots + x_n = x ; \quad x_i \geqq 0 .$$

Dostaneme tak úlohu, o níž jsme se již zmínili na začátku tohoto odstavce. Řešení $(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_n)$ této úlohy závisí na hodnotě parametru λ, tj. $\bar{x}_i = \bar{x}_i(x, \lambda)$. Hodnoty $(\bar{y}_1, \ldots, \bar{y}_n)$, které dávají maximum v (1,16), také závisí na λ, tj. $\bar{y}_i = \bar{y}_i(\lambda)$. Hodnotu λ měníme tak dlouho, až dosáhneme splnění podmínky

$$\bar{y}_1(\lambda) + \bar{y}_2(\lambda) + \ldots + \bar{y}_n(\lambda) = y .$$

Získané hodnoty $\bar{x}_i, \bar{y}_i$; $(i = 1, 2, \ldots, n)$ dávají zřejmě řešení původní úlohy.

b) *Metoda postupných aproximací*

Uvažujeme opět úlohu maximalizovat funkci (1,13) při podmínkách (1.14) a zvolíme dále nějakou aproximaci $(x_1^{(0)}, x_2^{(0)}, \ldots, x_n^{(0)})$ optimálních hodnot $(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_n)$. Určíme maximum funkce

$$F = \sum_{i=1}^{n} g_i(x_i^{(0)}, y_1)$$

na množině určené podmínkami

$$\sum_{i=1}^{n} y_i = y\,, \quad y_i \geqq 0\,.$$

Avšak to je opět úloha, ve které vystupují pouze funkce jedné proměnné a kterou tedy řešíme obvyklým způsobem na základě rekurentního vztahu

$$f_n(y) = \max_{0 \leqq y_n \leqq y} \left[g_n(x_n^{(0)}, y_n) + f_{n-1}(y - y_n)\right], \quad n \geqq 2\,,$$

$$f_1(y) = g_1(x_1^{(0)}, y)\,.$$

Označme optimální řešení této úlohy $(y_1^{(0)}, y_2^{(0)}, \ldots, y_n^{(0)})$. Nyní maximalizujeme funkci:

$$F = \sum_{i=1}^{n} g_i(x_i, y_i^{(0)})$$

na množině určené podmínkami

$$\sum_{i=1}^{n} x_i = x\,, \quad x_i \geqq 0\,.$$

Úlohu opět řešíme pomocí rekurentního vztahu

$$f_n(x) = \max_{0 \leqq x_n \leqq x} \left[g_n(x_n, y_n^{(0)}) + f_{n-1}(x - x_n)\right], \quad n \geqq 2\,,$$

$$f_1(x) = g_1(x_1, y_1^{(0)})\,.$$

Řešení této úlohy – označme je $(x_1^{(1)}, x_2^{(1)}, \ldots, x_n^{(1)})$ – užijeme jako nové aproximace optimálních hodnot $(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_n)$ a celý proces opakujeme. Dostaneme tak dvě posloupnosti

$$(x_1^{(k)}, x_2^{(k)}, \ldots, x_n^{(k)})\,, \quad (y_1^{(k)}, y_2^{(k)}, \ldots, y_n^{(k)})\,; \quad k = 1, 2, \ldots$$

které za určitých předpokladů konvergují k hledanému optimálnímu řešení. Často však nejsou tyto předpoklady splněny a získané posloupnosti konvergují k přípustným řešením dávajícím pouze relativní maxima. Různým výběrem počáteční aproximace $(x_1^{(0)}, x_2^{(0)}, \ldots, x_n^{(0)})$ můžeme v takovém případě získat řadu poloh relativních maxim a mezi nimi stanovit polohu absolutního maxima. Úlohu výběru absolutního extrému z relativních extrémů lze řešit až na základě znalostí konkrétních vlastností optimalizované funkce.

Abychom naznačili další možnosti metody postupných aproximací, uvažujme jeden ze základních typů funkcionálních rovnic vystupujících při užití metody dynamického programování:

$$f(p) = \max_{q} \left[g(\mathbf{p}, q) + f(T(\mathbf{p}, q))\right]. \tag{1.17}$$

Tuto rovnici můžeme chápat jako limitní případ vztahu

$$f_n(p) = \max_q [g(p, q) + f_{n-1}(T(p, q))]$$

pro $n \to \infty$.

Ve funkcionální rovnici (1.17) vystupují dvě neznámé funkce – funkce $f(p)$ a $q(p)$. Tyto funkce nejsou vzájemně nezávislé. Je-li známa funkce $f(p)$, lze funkci $q(p)$ určit maximalizací pravé strany v (1.17). Je-li známa funkce $q(p) = \hat{q}(p)$ vedoucí k maximu $f(p)$, lze $f(p)$ určit řešením rovnice

$$f(p) = g(p, \hat{q}) + f(T(p, \hat{q}))\,.$$

Při řešení funkcionální rovnice metodou postupných aproximací lze užít pravidla dvou způsobů.

(1) Volíme počáteční aproximaci $f_{(0)}(p)$ a určíme posloupnost funkcí $f_k(p)$ na základě rekurentního vztahu

$$f_k(p) = \max_q [g(p, q) + f_{k-1}(T(p, q)]\,.$$

Při maximalizaci získáváme současně i posloupnost funkcí $q_k(p)$.

Ukažme si pro názornost tento postup v případě, že $p = x$, $q = y$, $g(p, q) = \sqrt{(y)} + x - y$, $T(p, q) = ay + b[x - y]$, kde a, b jsou konstanty, pro které platí $0 < a < 1$, $0 < b < 1$ a také $0 \leqq y \leqq x$, tj. řešme rovnici

$$(1.17a) \qquad f(x) = \max_{0 \leqq y \leqq x} [\sqrt{(y)} + x - y + f(ay + b[x - y])]\,.$$

Zvolme

$$f_0 = ax\,,$$

potom

$$(1.17b) \qquad f_1(x) = \max_{0 \leqq y \leqq x} [\sqrt{(y)} + x - y + a\{ay + b[x - y]\}]\,.$$

Za předpokladu, že maximalizující hodnota y_1 je v intervalu $(0, x)$ (v opačném případě bude maximalizující hodnota buď $y_1 = 0$ nebo $y_1 = x$), dostaneme maximalizující hodnotu řešením rovnice

$$\frac{d}{dy} [\sqrt{(y)} + x - y + a\{ay + b[x - y]\}] = 0\,.$$

Snadno se lze přesvědčit, že v tomto případě dostaneme:

$$y_1(x) = \frac{1}{4[1 - a(a - b)]^2}\,.$$

Dosazením do (1,17b) dostaneme

$$f_1(x) = x(1 + ab) + \frac{1}{4[1 - a(a - b)]^2}\,.$$

Nyní celý postup opakujeme s tím, že úlohu (1,17a) hraje vztah

$$f_2(x) = \max_{0 \leqq y \leqq x} [\sqrt{(y)} + x - y + f_1(ay + b[x - y])] .$$

Postupujeme-li takto dále, dostaneme volbou $f_0(x) = ax$ pro ta x, pro která $0 \leqq$ $\leqq y_k(x) \leqq x$ tyto posloupnosti $\{y_k(x)\}$, $\{f_k(x)\}$:

$$y_k(x) = \frac{1}{4[1 - \{\sum_{n=0}^{k-1} b^n + ab^{k-1}\}(a - b)]^2} ,$$

$$f_k(x) = [\sum_{n=0}^{k-1} b^n + ab^k]\, x + \sum_{s=0}^{k-1} \frac{1}{4[1 - \sum_{n=0}^{s-1} b^n + ab^s\}(a - b)]} ,$$

$$k = 1, 2, \ldots$$

Volíme-li počáteční aproximaci $f_0(x)$ jinak, můžeme ovšem dostat jiné posloupnosti. Nechť se čtenář pokusí najít posloupnosti odpovídající volbě $f_0(x) = 0$, a budou-li se lišit od našich posloupností, nechť ukáže, že v limitě pro $k \to \infty$ vedou za určitých předpokladů ke stejnému řešení.

(2) Volíme počáteční aproximaci $q_0(\boldsymbol{p})$ funkce $q(\boldsymbol{p})$ a určíme funkci $f_0(\boldsymbol{p})$ jako řešení rovnice

$$f_0(\boldsymbol{p}) = g(\boldsymbol{p}, q_0) + f_0(T(\boldsymbol{p}, q_0)) .$$

Další aproximace $q_k(\boldsymbol{p}), f_k(\boldsymbol{p})$ získáme maximalizací ze vztahů

(1.17c) $$f_k(\boldsymbol{p}) = \max_q [g(\boldsymbol{p}, q) + f_{k-1}(T(\boldsymbol{p}, q))] .$$

Ilustrujme naznačený postup řešením stejné rovnice jako v případě (1). Volme

$$y_0(x) = 0 .$$

Rovnice (1,17a) bude mít pak tvar

$$f_0(x) = x + f_0(bx) .$$

Odtud dostaneme postupně $f_0(bx) = bx + f_0(b^2x)$, $f_0(b^2x) = b^2x + + f_0(b^3x), \ldots$, takže

$$f_0(x) = x + bx + b^2x + \ldots = \frac{x}{1 - b} .$$

Potom vztah (1,17c) bude mít pro $k = 1$ tvar

$$f_1(x) = \max_{0 \leqq y \leqq x} \left[\sqrt{(y)} + x - y + \frac{ay + b[x - y]}{1 - b}\right] .$$

Za předpokladu, že maximalizující hodnota y_1 leží v $(0, x)$ (což je splněno pro $x > \frac{1}{4}[(1-b)/(1-a)]^2$), dostaneme řešením rovnice

$$\frac{\mathrm{d}}{\mathrm{d}y}\left[\sqrt{(y)} + x - y + \frac{ay + b[x-y]}{1-b}\right] = 0$$

další aproximaci

$$y_1(x) = \frac{1}{4}\left(\frac{1-b}{1-a}\right)^2,$$

které odpovídá

$$f_1(x) = \frac{1}{4}\left(\frac{1-b}{1-a}\right) + \frac{x}{1-b}.$$

Postupujeme-li takto dále, tj. dosadíme-li $f_1(x)$ do (1,17c) a řešíme-li získanou rovnici, dostáváme postupně tyto posloupnosti $\{y_k(x)\}$, $\{f_k(x)\}$:

$$y_k(x) = \frac{1}{4}\left(\frac{1-b}{1-a}\right)^2,$$

$$f_k(x) = \frac{k}{4}\left(\frac{1-b}{1-a}\right) + \frac{x}{1-b}, \quad k = 1, 2, \ldots$$

I když se tyto posloupnosti liší od posloupností získaných prvním způsobem, lze se přesvědčit, že za určitých předpokladů vedou pro $k \to \infty$ ke stejnému řešení.

Nakonec je třeba připomenout, že zpravidla nelze postupné aproximace vyjádřit explicitně jako v našem příkladě, ale že musíme jednotlivé aproximace tabelovat.

4. Dopravní problém jako úloha dynamického programování

Vzhledem ke značné popularitě dopravních problémů ukážeme na závěr této kapitoly řešení těchto úloh metodami dynamického programování.

Z míst $A_1, A_2, \ldots, A_m$, ve kterých je k dispozici $a_1, a_2, \ldots, a_m$ jednotek určité produkce, máme dodat $b_1, b_2, \ldots, b_n$ jednotek uvažované produkce do míst $B_1, B_2, \ldots$ $\ldots, B_n$. Převoz x_{ij} jednotek produkce z místa A_i do místa B_j si vyžádá určité náklady vyjádřené hodnotou funkce $g_{ij}(x_{ij})$. Předpokládejme dále, že jde o vyvážený dopravní problém, tj. že platí

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j .$$

Dále jako obvykle vyžadujeme, aby

$$\sum_{j=1}^{n} x_{ij} = a_i , \quad i = 1, 2, \ldots, m ,$$

$$\sum_{i=1}^{m} x_{ij} = b_j\,, \quad j = 1, 2, \ldots, n\,,$$

$$x_{ij} \geqq 0\,, \quad i = 1, 2, \ldots, m\,,$$

$$j = 1, 2, \ldots, n\,.$$

Při těchto podmínkách je úkolem minimalizovat celkové dopravní náklady vyjádřené funkcí

$$F = \sum_{i=1}^{m} \sum_{j=1}^{n} g_{ij}(x_{ij})\,.$$

Jestliže funkce g_{ij} mají tvar $g_{ij}(x_{ij}) = d_{ij}x_{ij}$, kde d_{ij} jsou konstanty, dostáváme úlohu lineárního programování. Avšak známé metody pro lineární případ nelze užít v případě nelineárních funkcí g_{ij} kromě některých speciálních případů funkcí.

Uvažujme nejprve případ dvou míst A_1, A_2 a n míst $B_1, B_2, \ldots, B_n$. Proces uspokojování potřeb v místech $B_1, B_2, \ldots, B_n$ budeme považovat za dynamický proces, tj. nejprve uspokojíme potřeby v místě B_n, pak B_{n-1} atd.

Pro $a_1, a_2 > 0$, $n = 1, 2, \ldots$ označíme $f_n(a_1, a_2)$ dopravní náklady odpovídající optimálnímu řešení úlohy s množstvími a_1, a_2 v místech A_1, A_2 a s pevnými požadavky $b_1, b_2, \ldots, b_n$ v místech $B_1, B_2, \ldots, B_n$.

Náklady spojené s uspokojením požadavků v místě B_n jsou

$$g_{1n}(x_{1n}) + g_{2n}(x_{2n})\,,$$

kde $x_{1n} + x_{2n} = b_n$. Tímto krokem však zmenšíme počáteční množství a_1, a_2 v místech A_1, A_2 na hodnoty $a_1 - x_{1n}$, $a_2 - x_{2n}$. Na základě principu optimálnosti dostaneme rekurentní vztah[1])

$$f_n(a_1, a_2) = \min_{M} \left[g_{1n}(x_{1n}) + g_{2n}(x_{2n}) + f_{n-1}(a_1 - x_{1n}, a_2 - x_{2n})\right],$$

kde M je množina určená podmínkami

$$x_{1n} + x_{2n} = b_n\,,$$

$$0 \leqq x_{1n} \leqq a_1\,,$$

$$0 \leqq x_{2n} \leqq a_2\,;$$

$f_{n-1}(x, y)$ značí náklady spojené s uspokojením požadavků v $B_1, \ldots, B_{n-1}$ v úhrnné výši x, y z místa A_1, resp. A_2.

---

[1]) Abychom se vyhnuli nedorozumění, je třeba připomenout, že funkce f_n je třeba určit postupně pro $n = 1, 2, \ldots$, až index dosáhne počtu míst B_i původní úlohy; musíme tedy uvažovat hodnoty funkcí $f_n(a_1, a_2)$ i v jiných bodech než v bodě určeném původními, pevně danými hodnotami a_1, a_2; zde podobně jako ve vztahu (1.12) symbol x, stejně jako symboly a_1, a_2 označují proměnné.

Pro $n = 1$ je zřejmě

$$f_1(a_1, a_2) = g_{11}(a_1) + g_{21}(a_2).$$

Tohoto přístupu lze ovšem formálně užít i v případě, že počet míst A_i je větší než dvě, avšak pak se setkáváme s potížemi, které vznikají, jsou-li funkce f_n funkcemi většího počtu proměnných. Tyto obtíže lze, jak již jsme se zmínili, řešit metodou Lagrangeových multiplikátorů nebo metodou postupných aproximací.

Nejprve využijeme podmínky vyváženosti, které jsme dosud nijak nepoužili. Podle této podmínky je

$$a_1 + a_2 = \sum_{j=1}^{n} b_j.$$

To znamená, že při pevných hodnotách $b_1, b_2, \ldots, b_n$ lze hodnotu a_2 určit pomocí a_1, takže funkce $f_n(a_1, a_2)$ přejde ve funkci jedné proměnné $\hat{f}_n(a_1)$. Výše získaný rekurentní vztah přejde ve vztah

$$\hat{f}_n(a_1) = \min_{x_{1n}} \{g_{1n}(x_{1n}) + g_{2n}(b_n - x_{1n}) + \hat{f}_{n-1}(a_1 - x_{1n}),$$

kde x_{1n} musí vyhovovat podmínkám

$$0 \leqq a_1 - x_{1n},$$

$$0 \leqq b_n - x_{1n} \leqq \sum_{j=1}^{n} b_j - a_1.$$

Proměnná a_1 ve funkci $\hat{f}_n(a_1)$ se mění v intervalu $(0, \sum_{j=1}^{n} b_j)$.

Je-li tedy počet míst A rovný $m > 2$, lze úlohu řešit pomocí posloupnosti funkcí $m - 1$ proměnných. Užitím metody Lagrangeových multiplikátorů lze počet proměnných dále snížit. Uvažujme např. úlohu, ve které $m = 3$. V místě A_i je k dispozici množství a_i $(i = 1, 2, 3)$. Předpokládejme nejprve, že v místech A_2, A_3 máme k dispozici neomezená množství produkce. Místo funkce

$$F = \sum_{i=1}^{3} \sum_{j=1}^{n} g_{ij}(x_{ij}) \tag{1.18}$$

zavedeme pomocnou funkci

$$G = \sum_{i=1}^{3} \sum_{j=1}^{n} g_{ij}(x_{ij}) + \lambda \sum_{j=1}^{n} x_{2j} + \sum_{j=1}^{n} x_{3j}\ ^{1}). \tag{1.19}$$

Označíme $f_n(a_1)$ minimální hodnotu funkce (1.19), při pevném λ, na množině určené podmínkami:

$$\sum_{j=1}^{n} x_{ij} = a_1,$$

$$x_{1j}, x_{2j}, x_{3j} \geqq 0.$$

[1]) Vzhledem k podmínce vyváženosti nemusíme užívat dvou Lagrangeových multiplikátorů.

Uspokojením požadavků v místě B_n získáme na základě principu optimálnosti rekurentní vztah

$$(1.20) \qquad f_n(a_1) = \min_M \left[g_{1n}(x_{1n}) + g_{2n}(x_{2n}) + g_{3n}(x_{3n}) + \lambda x_{2n} + x_{3n} + f_{n-1}(a_1 - x_{1n})\right],$$

kde množina M je určena podmínkami

$$x_{1n} + x_{2n} + x_{3n} = b_n\,,$$
$$0 \leqq x_{1n} \leqq a_1\,,$$
$$x_{2n}, x_{3n} \geqq 0\,.$$

Zavedeme pomocnou funkci

$$g_n(x_{1n}, \lambda) = \min_{\overline{M}} \left[g_{2n}(x_{2n}) + g_{3n}(x_{3n}) + \lambda x_{2n} + x_{3n}\right],$$

kde množina $\overline{M}$ je určena podmínkami

$$x_2 n + x_{3n} = b_n - x_{1n}\,,$$
$$x_{2n}, x_{3n} \geqq 0\,.$$

Potom rekurentní vztah (1.20) bude mít tvar

$$f_n(a_1) = \min_{0 \leqq x_{1n} \leqq b_n} \left[g_{1n}(x_{1n}, \lambda) + g_n(x_{1n}, \lambda) + f_{n-1}(a_1 - x_{1n})\right],$$

kde parameter λ měníme tak dlouho, až množství produkce převážené z A_2 bude rovno a_2. Potom bude i $a_3 = \sum_{j=1}^{3} b_j - a_1 - a_2$.

V práci [3] jsou uvedeny číselné výsledky pro úlohu s $n = 10$:

$$F = \sum_{i-1}^{3} \sum_{j=1}^{10} (a_{ij}x_{ij} + b_{ij}x_{ij}^2 + c_{ij}(x_{ij}))\,,$$

kde

j	a_{1j}	b_{1j}	c_{1j}	a_{2j}	b_{2j}	c_{2j}	a_{3j}	b_{3j}	c_{3j}	bj
1	1,0	0	0	3,1	0	2	7	0	0	25
2	2,0	0	1	4,1	0	0	3	0	0	40
33	3,0	0,01	0	2,1	0	0	9	0	0	60
4	1,5	0	0	1,1	0,1	0	1	0	0	30
5	2,5	0	0	2,6	0	0	1	0	0	20
6	5,0	−0,01	10	3,0	0	0	2	0	5	30
7	3,0	0	0	1,0	0,2	5	4	0	0	35
8	6,0	0	0	2,0	0	0	3	0	6	30
9	6,0	−0,05	8	2,0	0	0	5	0	0	25
10	6,0	0	0	5,0	0,01	0	6	0	0	40

$a_1 = 100$, $a_2 = 80$, $a_3 = 155$. Sloupec c_{ij} je třeba chápat takto: $c_{ij}(x_{ij}) = 0$, je-li $x_{ij} = 0$ a $c_{ij}(x_{ij}) = c_{ij}$, je-li $x_{ij} > 0$.

Řešení popsanou metodou získáme při volbě $\lambda = 2{,}0$ a dostaneme toto řešení

j	x_{1j}	x_{2j}	x_{3j}
1	25	0	0
2	40	0	0
3	5	55	0
4	0	0	30
5	0	0	20
6	0	0	30
7	30	0	5
8	0	0	30
9	0	25	0
10	0	0	40

Celkové náklady činí 847,75.

Při metodě postupných aproximací postupujeme takto: Nejprve zvolíme libovolným způsobem množství přepravované z míst $A_3, A_4, \ldots, A_m$; tím bude uspokojena část požadavků v místech $B_1, B_2, \ldots, B_n$. Při takto pevně stanovených hodnotách $x_{3j}, x_{4j}, \ldots, x_{mj}$, $j = 1, 2, \ldots, n$ určujeme množství x_{1j}, x_{2j} převážné z míst A_1, A_2 optimálním způsobem. Řešení této úlohy lze provést standardním způsobem pomocí posloupností funkcí f_n jedné proměnné. Nyní ponecháme hodnoty x_{4j}, $x_{5j}, \ldots, x_{mj}$ a za x_{1j} volíme získané minimální hodnoty a opakujeme postup minimalizace pro místa A_2, A_3. Postupujeme-li takto dále, získáme posloupnost postupných aproximací, která v některých případech konverguje k řešení dávajícímu absolutní minimum účelové funkce.

(Dokončení příště)