



POLSKA AKADEMIA NAUK
Instytut Badań Systemowych

WSPOMAGANIE DECYZJI INWESTYCYJNYCH

**Roman Kulikowski,
Marek Libura,
Leon Słomiński**



WSPOMAGANIE DECYZJI INWESTYCYJNYCH

Polska Akademia Nauk • Instytut Badań Systemowych

Seria: BADANIA SYSTEMOWE
tom 21

Redaktor naukowy:

Prof. dr hab. Jakub Gutenbaum

Warszawa 1998

Roman KULIKOWSKI

Marek LIBURA

Leon SŁOMIŃSKI

**WSPOMAGANIE DECYZJI
INWESTYCYJNYCH**

Publikację opiniowali do druku:

Prof. dr hab. Maria Podgórska
Doc. dr hab. Leszek S. Zaremba

Książka powstała w wyniku realizacji projektu badawczego
finansowanego przez KOMITET BADAŃ NAUKOWYCH

Copyright © by Instytut Badań Systemowych PAN
Warszawa 1998

ISBN 83-85847-09-X
ISSN 0208-8029



Biblioteczna

Gench

44006

Część III.

Metody

optymalizacyjne

dla potrzeb

wspomagania decyzji

inwestycyjnych

4. Zadania programowania liniowego mieszanego

Zadania programowania liniowego mieszanego stanowią podstawową klasę zadań programowania matematycznego, do której sprowadzają się problemy optymalizacyjne przedstawione w Części II. Ogólny zapis zadania programowania liniowego mieszanego różni się od zapisu programowania liniowego ciągłego jedynie wprowadzeniem warunków całkowitoliczbowości na niektóre zmienne zadania i ma postać następującą:

$$\begin{aligned} \min (\text{albo max}) \quad & \sum_{i=1}^n c_i x_i \\ & \sum_{i=1}^n a_{ij} x_i \sim b_j, \quad j = 1, \dots, m \\ & l_i \leq x_i \leq u_i, \quad i = 1, \dots, n \\ & x_i \in \mathbb{Z} \text{ dla } i \in I \subseteq \{1, \dots, n\}, \end{aligned} \quad (\text{PLM})$$

gdzie, jak poprzednio, symbol \sim oznacza jeden z symboli \leq , $=$ albo \geq . Zmienne x_i dla $i \in I$ noszą nazwę *zmiennych całkowitoliczbowych* zadania, pozostałe nazywane są *zmiennymi ciągłymi*.

W przypadku, gdy $I = \{1, \dots, n\}$, mamy do czynienia z wspomnianym już wcześniej zadaniem programowania całkowitoliczbowego liniowego. Zmienne $x_i \in \mathbb{Z}$, dla których $l_i = 0$ oraz $u_i = 1$, noszą nazwę *zmiennych binarnych*.

Pojawienie się warunków całkowitoliczbowości zmienia w sposób zasadniczy trudności obliczeniowe związane z rozwiązywaniem zadania. Mimo istotnego postępu w konstrukcji algorytmów dla zadań programowania liniowego mieszanego zdarza się, że nawet zadania ze stosunkowo niewielką liczbą zmiennych całkowitoliczbowych wymagają bardzo dużego nakładu obliczeń, lub nie udaje się ich rozwiązywać w akceptowalnym czasie. Ponadto trudność rozwiązywania ta-

kich zadań w istotny sposób zależy od wyboru jednej z wielu zwykle równoważnych ich postaci (patrz np. Garfinkel, Nemhauser (1978), Williams (1993)).

W Części II szczegółowo omówiono przyczyny, dla których zachodzi potrzeba wprowadzania zmiennych całkowitoliczbowych przy formułowaniu zadań optymalizacyjnych dla potrzeb wspomagania decyzji na rynku finansowym. Tutaj przypomnimy tylko najważniejsze z tych przyczyn, natomiast kolejne punkty niniejszego rozdziału poświęcimy technikom modelowania z użyciem zmiennych całkowitoliczbowych i binarnych.

Można wyróżnić następujące dwa główne powody pojawiania się warunków całkowitoliczbowości w zadaniach optymalizacyjnych.

i. Całkowitoliczbowość wynika w sposób naturalny z niepodzielności pewnych wielkości.

Przykłady takich sytuacji były prezentowane w poprzednich rozdziałach i obejmują między innymi przypadki, gdy pewne instrumenty finansowe sprzedawane są w dużych, niepodzielnych pakietach. Należy tu jednak podkreślić, że nie jest konieczne wprowadzanie zmiennych całkowitoliczbowych wówczas, gdy zaokrąglenie niecałkowitoliczbowych rozwiązań nie prowadzi do znacznej procentowej zmiany wartości rozwiązania lub naruszenia jego dopuszczalności. W takich sytuacjach wystarczające jest zwykle użycie znacznie prostszego obliczeniowo zadania ze zmiennymi ciągłymi i dokonanie zaokrąglenia rozwiązania. W ogólnym przypadku trudno jest podać regułę, kiedy niepodzielnym wielkościom muszą w zadaniu odpowiadać zmienne całkowitoliczbowe, a kiedy wystarczające jest zastosowanie zmiennych ciągłych i dokonanie zaokrąglenia. Często decyzja taka może być dopiero wynikiem analizy otrzymanego rozwiązania zadania ciągłego i jego zaokrąglenia.

ii. Całkowitoliczbowość wynika z potrzeby wprowadzenia dodatkowych (zwykle binarnych) zmiennych, służących do modelo-

wania różnego rodzaju warunków logicznych lub do przekształcenia postaci zadania.

W odróżnieniu od „naturalnych” zmiennych decyzyjnych omawianych wyżej, te dodatkowe zmienne można traktować jako zmienne „sztuczne”. Istnieje kilka typowych sytuacji, w których takie zmienne są wprowadzane. Omówimy je szczegółowo w dalszej części rozdziału przy przedstawianiu technik modelowania z wykorzystaniem zmiennych binarnych.

4.1. Modelowanie z użyciem zmiennych binarnych

4.1.1. Modelowanie prostych sytuacji decyzyjnych typu TAK-NIE

Jest to najbardziej typowa sytuacja, w której pojawiają się zmienne binarne. Zwykle w słownym opisie problemu optymalizacyjnego mamy tu do czynienia z ustalonym skończonym zbiorem decyzji. Zmienna binarna, przyporządkowana konkretnej decyzji z tego zbioru, modeluje prosty wybór typu TAK-NIE i przyjmuje wartość 1, jeśli decyzja ta jest podjęta, a wartość 0 w przeciwnym przypadku. Zmienna taka jest następnie wykorzystywana do konstrukcji funkcji celu i ograniczeń zadania.

4.1.2. Tworzenie zmiennych wskaźnikowych dla zmiennych zadania

Zmienne wskaźnikowe są dodatkowymi zmiennymi binarnymi, służącymi do rozróżniania sytuacji, w których zachodzą lub nie zachodzą warunki dotyczące innych zmiennych zadania.

Rozważmy najprostszą z takich sytuacji. Załóżmy, że mamy w zadaniu zdefiniowaną zmienną (ciągłą lub całkowitoliczbową) $x \geq 0$ i chcemy rozróżnić dwa przypadki:

- i. $x = 0$

ii. $x > 0$.

Takie rozróżnienie może być potrzebne na przykład wtedy, gdy zmienna x odpowiada udziałowi akcji danego przedsiębiorstwa w portfelu i chcemy kontrolować liczbę różnych akcji w portfelu. Do tego potrzebne jest przyporządkowanie zmiennej ciągłej x dodatkowej zmiennej binarnej $\delta \in \{0,1\}$, wskazującej, z którą z dwu powyższych sytuacji: $x = 0$, $x > 0$, mamy do czynienia. Przyporządkowanie to może mieć postać pary implikacji

$$x > 0 \rightarrow \delta = 1, \quad (21)$$

$$x = 0 \rightarrow \delta = 0. \quad (22)$$

Modelowanie pary implikacji (21), (22) przy pomocy nierówności liniowych jest możliwe wówczas, gdy znane są liczby rzeczywiste dodatnie M i m , takie że spełnione są następujące warunki:

1° zmienna x nie może przyjmować wartości większych niż M ;

2° jeśli $x > 0$, to $x \geq m$.

Wówczas para implikacji (21), (22) jest modelowana w zadaniu poprzez wprowadzenie dwóch dodatkowych ograniczeń liniowych

$$x - M\delta \leq 0, \quad (23)$$

$$x - m\delta \geq 0. \quad (24)$$

Łatwo sprawdzić, że istotnie, spełnienie ograniczenia (23) pociąga za sobą warunek (21), a ograniczenie (24) wymusza (22).

W praktycznych problemach dobór wartości M nie sprawia zwykle kłopotu. Często w zadaniu mamy w postaci jawnej podane ograniczenie górne na zmienną x i wówczas jako wartość M można wziąć po prostu wartość tego ograniczenia. Natomiast wprowadzenie wartości m może się czasem wydawać zabiegiem sztucznym i wymaga odwołania się do interpretacji zmiennej x lub do przyjęcia jako wartości m dostatecznie małej liczby, porównywalnej z dokładnością

obliczeń. W niektórych sytuacjach wartość m wynika w sposób naturalny z warunków zadania, co ilustruje poniższy przykład.

Przykład 4. Załóżmy, że mając do dyspozycji n różnych akcji chcemy skonstruować portfel zawierający nie mniej niż k_1 , ale nie więcej niż k_2 różnych akcji, gdzie k_1 oraz $k_2 \geq k_1$ są danymi liczbami. Dodatkowo, w przypadku, gdy akcje danego typu są włączane do portfela, wymagane jest, aby ich udział w portfelu nie był mniejszy niż 1%. Załóżmy ponadto, że pozostałe wymagania dotyczące konstrukcji portfela są takie, jak w Przykładzie 2 i zajmijmy się wyłącznie układem ograniczeń zadania. Pierwotny układ ograniczeń tego zadania ma postać

$$\begin{aligned} \sum_{i=1}^n x_i &= 1 \\ \sum_{i=1}^n \bar{r}_i x_i &= r \\ x_i &\geq 0, \end{aligned} \tag{25}$$

gdzie, jak poprzednio, x_i ma interpretację udziału akcji i w portfelu.

Dla zapisania warunków dywersyfikacji i warunków minimalnego udziału akcji w portfelu wprowadźmy dla każdej zmiennej x_i , $i = 1, \dots, n$, zmienną wskaźnikową $z_i \in \{0, 1\}$, spełniającą warunek $z_i = 1 \leftrightarrow x_i > 0$. Wymaga to dołączenia do układu warunków (25) pary ograniczeń (23), (24) dla każdego $i = 1, \dots, n$. Zauważmy, że każda ze zmiennych x_i , $i = 1, \dots, n$, spełnia warunek $x_i \leq 1$, a zatem jako wspólną wartość M można przyjąć $M = 1$. Ponadto warunek minimalnego udziału akcji w portfelu w sposób naturalny narzuca wspólną dla wszystkich zmiennych x_i wartość $m = 0,01$. Zatem układ ograniczeń, tworzących zmienne wskaźnikowe dla $i = 1, \dots, n$, ma postać

$$\begin{aligned} x_i - z_i &\leq 0, \\ x_i - 0,01 z_i &\geq 0. \end{aligned} \tag{26}$$

Mając tak utworzone zmienne wskaźnikowe z_i , $i = 1, \dots, n$, łatwo jest już zapisać warunek dywersyfikacji portfela w postaci pary ograniczeń

$$\begin{aligned} \sum_{i=1}^n z_i &\geq k_1, \\ \sum_{i=1}^n z_i &\leq k_2. \end{aligned} \tag{27}$$

Pełny układ ograniczeń zadania z warunkami dywersyfikacji i warunkami na minimalne udziały jest więc następujący:

$$\begin{aligned} \sum_{i=1}^n x_i &= 1 \\ \sum_{i=1}^n \bar{r}_i x_i &= r \\ x_i - z_i &\leq 0, \quad i = 1, \dots, n \\ x_i - 0,01 z_i &\geq 0, \quad i = 1, \dots, n \\ \sum_{i=1}^n z_i &\geq k_1 \\ \sum_{i=1}^n z_i &\leq k_2 \\ x_i &\geq 0, \quad i = 1, \dots, n \\ z_i &\in \{0, 1\}, \quad i = 1, \dots, n. \end{aligned} \tag{28}$$

W Dodatku 5.3 podane są wydruki sformułowania oraz rozwiązania omawianego tu zadania dla przykładowych danych.

□

4.1.3. Tworzenie zmiennych wskaźnikowych dla ograniczeń

W wielu sytuacjach przy modelowaniu różnorodnych warunków logicznych potrzebne jest użycie dodatkowych zmiennych binarnych, pozwalających rozróżnić sytuację, w której dane ograniczenie zachodzi, od sytuacji, w której ono nie zachodzi. Zmienne takie noszą nazwę zmiennych wskaźnikowych dla ograniczeń.

Podamy tu sposoby tworzenia zmiennych wskaźnikowych dla ograniczeń liniowych o postaci $a^T x \leq b$, $a^T x \geq b$, $a^T x = b$, gdzie $a \in \mathbb{R}^n$, $b \in \mathbb{R}$, a x jest n -elementowym wektorem zmiennych zadania. Należy jednak podkreślić, że metoda tworzenia zmiennych wskaźnikowych nie zależy od tego, czy mamy do czynienia z ograniczeniem liniowym, czy nieliniowym.

Rozważmy ograniczenie

$$a^T x \leq b \quad (29)$$

i załóżmy, że chcemy utworzyć zmienną binarną, spełniającą następujące warunki:

$$a^T x \leq b \rightarrow \delta = 1, \quad (30)$$

$$a^T x > b \rightarrow \delta = 0. \quad (31)$$

Sytuacja jest tu bardzo podobna do tej, która dotyczyła tworzenia zmiennych wskaźnikowych dla innych zmiennych zadania i została opisana w poprzednim punkcie. Na to, by pomóc modelować warunki (30), (31) w postaci dodatkowych ograniczeń liniowych, potrzebna jest znajomość trzech liczb, M , m oraz $\varepsilon > 0$, takich że:

- i. M jest oszacowaniem od góry wartości wyrażenia $a^T x - b$;
- ii. m jest oszacowaniem od dołu wartości wyrażenia $a^T x - b$;
- iii. ε spełnia następującą implikację:

jeśli $a^T x > b$ to $a^T x \geq b + \varepsilon$.

O ile dobór wartości M i m nie sprawia zwykle kłopotów, o tyle wybór odpowiedniej wartości ε może rodzić pewne problemy. Jeśli z warunków zadania wynika, że wartości wyrażenia $a^T x - b$ są całkowitoliczbowe, to wówczas wystarczy przyjąć $\varepsilon = 1$. W przeciwnym przypadku jako wartość ε należy wziąć dostatecznie małą liczbą, którą można interpretować jako dokładność obliczeń, stosowaną przy badaniu naruszenia ograniczenia $a^T x \leq b$. Podobnie, jak w przypadku zmiennych wskaźnikowych omawianych w poprzednim punkcie, problem doboru wartości ε nie występuje, gdy chcemy jedynie modelować implikację (31). Zauważmy, że jest ona równoważna warunkowi

$$\delta = 1 \rightarrow a^T x \leq b. \quad (32)$$

Łatwo sprawdzić, że powyższy warunek można zapisać w postaci ograniczenia

$$a^T x + M\delta \leq M + b. \quad (33)$$

Jeśli bowiem $\delta = 1$, to (33) wymusza ograniczenie $a^T x \leq b$. Natomiast dla $\delta = 0$ warunek (33) jest spełniony dla dowolnego wektora x , co wynika z definicji M .

Modelowanie implikacji (30) wymaga już użycia stałych m oraz ε . Zauważmy, że (30) jest równoważne warunkowi

$$\delta = 0 \rightarrow a^T x > b, \quad (34)$$

co można przepisać, wykorzystując wprowadzoną wielkość ε , w postaci implikacji

$$\delta = 0 \rightarrow a^T x \geq b + \varepsilon. \quad (35)$$

Łatwo sprawdzić, że powyższy warunek można zastąpić nierównością

$$a^T x - (m - \varepsilon)\delta \geq b + \varepsilon. \quad (36)$$

Zatem, dla utworzenia zmiennej wskaźnikowej, spełniającej oba warunki (30), (31), należy dołączyć do ograniczeń zadania dwie nierówności (33), (36).

Tworzenie zmiennej wskaźnikowej dla nierówności $a^T x \geq b$, to znaczy zmiennej $\delta \in \{0,1\}$, spełniającej warunki

$$a^T x \geq b \rightarrow \delta = 1, \quad (37)$$

$$a^T x < b \rightarrow \delta = 0, \quad (38)$$

odbywa się w podobny sposób. Można pokazać, że warunek (37) jest równoważny nierówności

$$a^T x - (M + \varepsilon)\delta \leq b - \varepsilon, \quad (39)$$

gdzie, jak poprzednio, M jest oszacowaniem od góry wartości wyrażenia $a^T x - b$, natomiast ε jest dobrane w taki sposób, że spełniona jest implikacja

$$a^T x < b \rightarrow a^T x \leq b - \varepsilon. \quad (40)$$

Podobnie, dla zapisania warunku (38), czyli równoważnego warunku $\delta = 1 \rightarrow a^T x \geq b$, należy wprowadzić dodatkowe ograniczenie

$$a^T x + m\delta \geq m + b, \quad (41)$$

gdzie m jest oszacowaniem od dołu wartości wyrażenia $a^T x - b$.

Analogicznie może być wprowadzona zmienna wskaźnikowa $\delta \in \{0,1\}$ dla ograniczenia równościowego $a^T x = b$, spełniająca warunki

$$a^T x = b \rightarrow \delta = 1, \quad (42)$$

$$a^T x \neq b \rightarrow \delta = 0. \quad (43)$$

Modelowania implikacji (42) polega na wprowadzeniu dwóch pomocniczych zmiennych binarnych δ_1, δ_2 , spełniających zależności

$$a^T x \geq b \rightarrow \delta_1 = 1, \quad (44)$$

$$a^T x \leq b \rightarrow \delta_2 = 1, \quad (45)$$

oraz na powiązaniu ich ze zmienną wskaźnikową δ poprzez wprowadzenie warunku

$$\delta_1 = 1 \text{ oraz } \delta_2 = 1 \rightarrow \delta = 1. \quad (46)$$

Warunek (46) można wprowadzić do zadania poprzez dołączenie ograniczenia

$$\delta_1 + \delta_2 - \delta \leq 1. \quad (47)$$

Natomiast warunki (44) i (45) modeluje się wprowadzając ograniczenia (39) i (36) odpowiednio ze zmiennymi δ_1 i δ_2 .

Modelowanie implikacji (43) jest prostsze. Zauważmy, że jest ona równoważna warunkowi

$$\delta = 1 \rightarrow a^T x = b. \quad (48)$$

Oznacza to, że

$$\delta = 1 \rightarrow a^T x \leq b \text{ oraz } a^T x \geq b, \quad (49)$$

co łatwo już zapisać wprowadzając parę ograniczeń (33) i (41).

Widzimy więc, że utworzenie zmiennej wskaźnikowej δ , spełniającej warunki (42) i (43), wymaga dołączenia do zadania następującego układu ograniczeń:

$$\begin{aligned}
 a^T x - (M + \varepsilon)\delta_1 &\leq b - \varepsilon \\
 a^T x - (m - \varepsilon)\delta_2 &\geq b + \varepsilon \\
 a^T x + M\delta &\leq M + b \\
 a^T x + m\delta &\geq m + b \\
 \delta_1 + \delta_2 - \delta &\leq 1 \\
 \delta_1, \delta_2, \delta &\in \{0, 1\}.
 \end{aligned}
 \tag{50}$$

Stałe M i m są równe, jak poprzednio, odpowiednio ograniczeniom od góry i od dołu na wartości wyrażenia $a^T x - b$, natomiast ε jest dobrane w taki sposób, aby były spełnione implikacje

$$\text{jeśli } a^T x < b, \text{ to } a^T x \leq b - \varepsilon$$

oraz

$$\text{jeśli } a^T x > b, \text{ to } a^T x \geq b + \varepsilon.$$

W kolejnych punktach podamy przykłady zastosowania przedstawionej wyżej metody tworzenia zmiennych wskaźnikowych.

4.1.4. Modelowanie warunków logicznych z użyciem binarnych zmiennych decyzyjnych i zmiennych wskaźnikowych

Przy formułowaniu problemu optymalizacyjnego w postaci zadania programowania matematycznego często zachodzi potrzeba modelowania różnego rodzaju warunków logicznych, które formalnie można zapisać w języku rachunku zdań. Rozważmy bardzo uproszczony przykład takiej sytuacji.

Przykład 5. Załóżmy, że chcemy modelować następujący warunek:

(W) Jeśli do portfela akcji włączamy co najmniej dwie spośród czterech spółek A_1, A_2, A_3, A_4 , to również musimy wprowadzić do portfela co najmniej jedną ze spółek A_5, A_6 .

W powyższym warunku mamy dwa zdania:

Zdanie P : „W portfelu są co najmniej dwie spółki spośród A_1, A_2, A_3, A_4 ”;

Zdanie Q : „W portfelu jest co najmniej jedna spółka spośród A_5, A_6 ”.

Zgodnie z warunkami zadania należy modelować implikację

$$P \rightarrow Q.$$

Niech $\delta_i, i=1, \dots, 5$, będzie zmienną równą 1 w przypadku, gdy spółka A_i jest w portfelu, oraz równą 0, w przeciwnym przypadku. Jeśli zmiennymi pierwotnymi w zadaniu są zmienne ciągłe $x_i, i=1, \dots, 5$, interpretowane jako udziały akcji spółek w portfelu (tak jak to było w Przykładach 2, 3, 4), to zmienne binarne $\delta_i, i=1, \dots, 5$, należy utworzyć jako zmienne wskaźnikowe dla zmiennych x_i zgodnie z zasadami opisanymi w punkcie 4.1.2.

Zdanie P jest równoważne warunkowi

$$\delta_1 + \delta_2 + \delta_3 + \delta_4 \geq 2, \quad (51)$$

a zdaniu Q odpowiada warunek

$$\delta_5 + \delta_6 \geq 1. \quad (52)$$

Aby modelować implikację $P \rightarrow Q$, należy utworzyć zmienne wskaźnikowe δ' i δ'' odpowiednio dla ograniczeń (51) oraz (52). Mając takie zmienne, możemy zapisać implikację $P \rightarrow Q$ w postaci równoważnego ograniczenia liniowego

$$\delta' - \delta'' \leq 0. \quad (53)$$

Tworzenie zmiennych wskaźnikowych δ' i δ'' przebiega zgodnie z metodą opisaną w punkcie 4.1.3.. Zmienne te mają spełniać warunki

$$\delta_1 + \delta_2 + \delta_3 + \delta_4 \geq 2 \rightarrow \delta' = 1, \quad (54)$$

$$\delta_5 + \delta_6 \geq 1 \rightarrow \delta'' = 1. \quad (55)$$

Aby zapisać warunek (54) w postaci nierówności liniowej zgodnie z (39), należy wziąć $M = 2$ i $\varepsilon = 1$. Otrzymujemy równoważne ograniczenie zadania

$$\delta_1 + \delta_2 + \delta_3 + \delta_4 - 3\delta' \leq 1. \quad (56)$$

Podobnie dla (55), biorąc $M = 1$, $\varepsilon = 1$, mamy na podstawie (39)

$$\delta_5 + \delta_6 - 2\delta'' \leq 0. \quad (57)$$

A zatem dla zapisania warunku (W) należy utworzyć zmienne wskaźnikowe δ_i , $i = 1, \dots, 6$, a następnie dołączyć do zadania trzy ograniczenia (53), (56) i (57) oraz dwie zmienne binarne δ' , δ'' . \square

Podany wyżej przykład dotyczył modelowania implikacji $P \rightarrow Q$. W analogiczny sposób przebiega modelowanie innych wyrażeń logicznych. Załóżmy, że mamy do czynienia z dwoma zdaniem P , Q , i że zdanie P jest równoważne wyrażeniu $\delta_P = 1$, a zdanie Q wyrażeniu $\delta_Q = 1$, gdzie δ_P , δ_Q są zmiennymi przyjmującymi wartości 0 albo 1. Łatwo sprawdzić, że wówczas zachodzą następujące równoważności przedstawione w poniższej tabeli (symbole \vee , \wedge , \rightarrow , \leftrightarrow , \neg oznaczając odpowiednio dyzjuncję, koniunkcję, implikację, równoważność i negację):

Wyrażenie logiczne	Równoważne ograniczenia
$P \vee Q$	$\delta_P + \delta_Q \geq 1$
$P \wedge Q$	$\delta_P = 1; \delta_Q = 1$
$P \rightarrow Q$	$\delta_P - \delta_Q \leq 0$
$P \leftrightarrow Q$	$\delta_P - \delta_Q = 0$
$\neg P$	$\delta_P = 0$

Modelowanie bardziej złożonych wyrażeń logicznych w postaci ograniczeń jest procesem wieloetapowym i wymaga pewnej wprawy. Ogólny schemat takiego postępowania polega na przekształceniu poszczególnych prostych członów wyrażenia logicznego w ograniczenia, tworzeniu dla tych ograniczeń zmiennych wskaźnikowych, a następnie wykorzystaniu tych zmiennych do zapisu zdań, będących równo-
ważnikami rozpatrywanych członów. Prowadzi to do stopniowego upraszczania wyrażenia logicznego. Zwykle stosuje się też przekształcanie zarówno wyrażenia pierwotnego, jak i wyrażeń pośrednich, do równoważnych, wygodniejszych do modelowania postaci, z wykorzystaniem rachunku zdań. W literaturze opisywane są podejścia do automatyzacji takiego procesu modelowania (patrz McKinnon, Williams (1989)).

4.1.5. Modelowanie z użyciem zbiorów typu SOS

Zbiory typu SOS (*Special Ordered Sets*) są podzbiorami zmiennych zadania, spełniającymi pewne szczególne ograniczenia. Ich wyróżnienie wynika zarówno z wygody modelowania (patrz Przykład 6), jak i z faktu, że w pakietach obliczeniowych są one traktowane w specjalny sposób, co może prowadzić do bardziej efektywnej strategii rozwiązywania zadania.

Zwykle wyróżnia się dwa typy zbiorów SOS:

- 1) *Zbiór SOS typu 1* - Jest to zbiór zmiennych zadania, z których dokładnie jedna może być różna od zera w dowolnym rozwiązaniu dopuszczalnym.
- 2) *Zbiór SOS typu 2* - Jest to *uporządkowany* zbiór zmiennych zadania, z których co najwyżej dwie mogą być różne od zera. Ponadto, jeśli dwie zmienne są różne od zera, to muszą to być kolejne zmienne zbioru.

W niektórych pakietach stosuje się pewne modyfikacje powyższych definicji. Na przykład w pakiecie CPLEX 3.0 zbiór SOS typu 1

jest definiowany jako zbiór tych zmiennych zadania, z których co najwyżej jedna jest różna od zera.

Zbiory SOS typu 1 są używane w sytuacji, gdy chcemy modelować warunki typu: „należy wybrać jeden wariant spośród danego zbioru wariantów decyzji”. Załóżmy, że rozważamy n możliwych wariantów decyzji ponumerowanych liczbami naturalnymi od 1 do n . Niech δ_i , $i = 1, \dots, n$, będzie zmienną przyporządkowaną wariantowi i . Chcemy, by przyjęciu i -tego wariantu decyzji odpowiadał warunek $\delta_i = 1$. Wymaganie: „wybieramy jeden spośród n wariantów decyzji” może być modelowane poprzez nałożenie warunków binarności na zmienne δ_i , tzn.

$$\delta_i = \{0,1\}, \quad i = 1, \dots, n$$

i dołączenie ograniczenia

$$\delta_1 + \dots + \delta_n = 1. \quad (58)$$

Jeśli jednak przyjmiemy, że zmienne δ_i , $i = 1, \dots, n$, tworzą zbiór SOS typu 1, to warunek binarności zmiennych nie jest konieczny, bowiem wówczas ograniczenie (58) i warunki nieujemności zmiennych $\delta_i \geq 0$, $i = 1, \dots, n$, gwarantują całkowitoliczbowość zmiennych δ_i . Takie sformułowanie może prowadzić do skrócenia czasu rozwiązywania zadania.

Zbiory SOS typu 1 są również wygodnym narzędziem do modelowania sytuacji, gdy chcemy użyć zmiennej dyskretnej w , przyjmującej wartości ze zbioru $\{w_1, w_2, \dots, w_l\}$. Wartości te mogą być w praktycznym problemie, na przykład, rozważanymi poziomami inwestycji, wartościami bloków emisji akcji, itp. Zmienna w może być utworzona poprzez zadeklarowanie zmiennych ciągłych $\delta_i \geq 0$, $i = 1, \dots, n$, należących do zbioru SOS typu 1, oraz włączenie do zadania dwóch ograniczeń

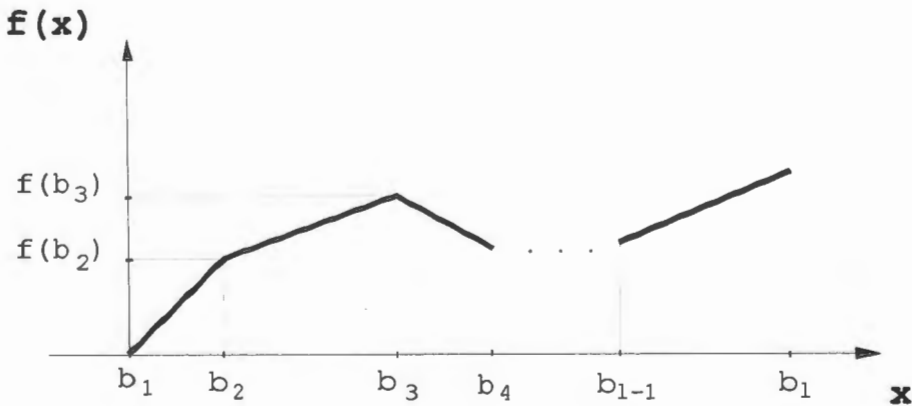
$$w - w_1\delta_1 - w_2\delta_2 - \dots - w_l\delta_l = 0,$$

$$\delta_1 + \delta_2 + \dots + \delta_l = 1.$$

Zauważmy, że podobnie jak poprzednio, ze względu na użycie zbioru SOS typu 1, nie jest konieczne nakładanie warunków binarności na zmienne δ_i .

Najczęstsze użycie zbiorów SOS typu 2 wiąże się z modelowaniem zależności nieliniowych, które są aproksymowane funkcjami odcinkowo-liniowymi. Poniższy przykład ilustruje taką sytuację.

Przykład 6. Załóżmy, że chcemy użyć w ograniczeniach lub w funkcji celu zadania, funkcji odcinkowo-liniowej $f(x)$ zmiennej ciągłej $x \in [b_1, b_l]$. Przykład takiej funkcji jest pokazany na poniższym rysunku.



Rys.1. Przykład funkcji odcinkowo-liniowej.

Wprowadźmy zmienne ciągłe $\delta_i \geq 0$, $i=1, \dots, l$, i przyjmijmy, że tworzą one zbiór SOS typu 2. Wówczas nieliniowa zależność $f(x)$ dla $x \in [b_1, b_l]$ może być modelowana poprzez następujący układ ograniczeń liniowych:

$$f(x) = f(b_1)\delta_1 + f(b_2)\delta_2 + \dots + f(b_l)\delta_l \quad (59)$$

$$x = b_1\delta_1 + b_2\delta_2 + \dots + b_l\delta_l \quad (60)$$

$$\delta_1 + \delta_2 + \dots + \delta_l = 1. \quad (61)$$

Aby się o tym przekonać załóżmy, że x przyjmuje wartość z przedziału $[b_i, b_{i+1}]$, $i = 1, \dots, l-1$. Wówczas z warunku, że zmienne δ_i , $i = 1, \dots, l$, tworzą zbiór SOS typu 2 oraz z warunków (60), (61) wynika, że $x = b_i\delta_i + b_{i+1}\delta_{i+1}$, $\delta_i + \delta_{i+1} = 1$, a zatem wartość x jest wyrażona jako kombinacja wypukła wartości b_i oraz b_{i+1} . Zgodnie z odcinkowo-liniową zależnością wartość $f(x)$ powinna być wówczas kombinacją wypukłą wartości $f(b_i)$ oraz $f(b_{i+1})$. Zauważmy, że tak istotnie jest, bowiem z (59) wynika że wówczas $f(x) = f(b_i)\delta_i + f(b_{i+1})\delta_{i+1}$.

W przypadku, gdy nie dysponujemy pakietem z możliwością deklarowania zbiorów SOS typu 2, możemy modelować odcinkowo-liniowe funkcje poprzez wprowadzenie dodatkowych zmiennych binarnych $z_i \in \{0,1\}$, $i = 1, \dots, l-1$, oraz układu dodatkowych ograniczeń. Ich celem jest zapisanie warunku, że co najwyżej dwie kolejne zmienne ciągłe δ_i , $i = 1, \dots, l$, są dodatnie. Można sprawdzić, że taki warunek jest modelowany przez następujący układ ograniczeń:

$$\begin{aligned} \delta_1 - z_1 &\leq 0 \\ \delta_2 - z_1 - z_2 &\leq 0 \\ \delta_3 - z_2 - z_3 &\leq 0 \\ &\vdots \\ \delta_{l-1} - z_{l-2} - z_{l-1} &\leq 0 \\ \delta_l - z_{l-1} &\leq 0 \\ z_1 + z_2 + \dots + z_{l-1} &= 1. \end{aligned} \quad (62)$$

Aby zatem móc korzystać z odcinkowo-liniowej funkcji $f(x)$ bez użycia zbiorów SOS typu 2, należy zadeklarować zmienne ciągłe

$\delta_i \geq 0$, $i = 1, \dots, l$, zmienne binarne $z_i \in \{0, 1\}$, $i = 1, \dots, l-1$, oraz dołączyć do zadania układ ograniczeń (59), (60), (61), (62).

□

4.1.6. Modelowanie ograniczeń dyzjunkcyjnych

W dotychczas rozpatrywanych zadaniach optymalizacyjnych zawsze mieliśmy do czynienia z sytuacją, w której żądaliśmy, aby wszystkie ograniczenia były jednocześnie spełnione. W niektórych przypadkach chcemy jednak modelować sytuację bardziej złożoną. Załóżmy mianowicie, że w zadaniu optymalizacyjnym mamy do czynienia z pewną liczbą K podzbiorów ograniczeń i dla dopuszczalności rozwiązania wystarcza, aby były spełnione ograniczenia z dowolnego podzbioru. Mówimy wówczas o *dyzjunkcyjnym układzie ograniczeń*. W najprostszym przypadku możemy mieć do czynienia z parą pojedynczych ograniczeń, np. $(a')^T x \leq b'$, $(a'')^T x \leq b''$, gdzie a' , $a'' \in \mathbb{R}^n$, b' , $b'' \in \mathbb{R}$, i dla dopuszczalności rozwiązania x wystarcza, aby było spełnione jedno z nich (albo oba równocześnie). Powyższe sytuacje mogą być w łatwy sposób modelowane z użyciem dodatkowych zmiennych binarnych.

Rozważmy ogólny przypadek układu ograniczeń dyzjunkcyjnych. Załóżmy, że mamy do czynienia z K podzbiarami ograniczeń i niech zdanie P_i , $i = 1, \dots, K$, oznacza warunek „Ograniczenia z podzbioru i są spełnione”. Wprowadźmy K zmiennych binarnych $\delta_i \in \{0, 1\}$, $i = 1, \dots, K$, potrzebnych do zapisania następujących implikacji:

$$\delta_i = 1 \rightarrow P_i, \quad i = 1, \dots, K. \quad (63)$$

Powyższe warunki mogą być w łatwy sposób modelowane zgodnie z podejściem opisanym w punkcie 4.1.3. Rozważmy bowiem i -tą grupę ograniczeń. Dla zapisania warunku $\delta_i = 1 \rightarrow P_i$ należy dla każdego ograniczenia z tej grupy wprowadzić warunek (33) albo (41) w przypadku, gdy jest to odpowiednie ograniczenie nierównościowe, albo

oba te warunki jednocześnie w przypadku, gdy jest to ograniczenie równościowe.

Mając wprowadzone zmienne binarne δ_i , $i = 1, \dots, K$, można już w łatwy sposób modelować warunek dyzjunkcji ograniczeń, który w ogólnym przypadku jest formułowany w następującej postaci dla całkowitego k , spełniającego warunki $1 \leq k < K$:

Co najmniej k spośród K podzbiorów ograniczeń musi być spełnionych po to, aby rozwiązanie było dopuszczalne.

Warunek powyższy jest modelowany poprzez wprowadzenie następującego ograniczenia:

$$\delta_1 + \dots + \delta_K \geq k. \quad (64)$$

Poniższy bardzo prosty przykład ilustruje użycie ograniczeń dyzjunkcyjnych w zadaniu konstrukcji portfela papierów wartościowych.

Przykład 6. Załóżmy, że w zbiorze N wszystkich papierów wartościowych, które mogą być wprowadzane do portfela, wyróżniamy cztery rozłączne podzbiory papierów $A_1, A_2, B_1, B_2 \subseteq N$. (Podzbiory te mogą na przykład odpowiadać różnym typom instrumentów finansowych, akcjom spółek z różnych sektorów, obligacjom o różnych klasach ryzyka, itp.). Portfel może być zaakceptowany w dwóch przypadkach:

- i. Udział papierów z podzbioru A_1 jest nie mniejszy niż 60% i udział papierów z podzbioru A_2 jest nie większy niż 15%;
- ii. Udział papierów z podzbioru B_1 jest nie większy niż 50% i udział papierów z podzbioru B_2 jest nie mniejszy niż 20%.

Niech x_i , $i \in N$, będzie zmienną ciągłą oznaczającą udział papieru i w portfelu. Oprócz naturalnego warunku

$$\sum_{i \in N} x_i = 1 \quad (65)$$

układ ograniczeń zadania musi zawierać dodatkowe ograniczenia modelujące warunki dyzjunkcyjne. Mamy tu do czynienia z dwiema grupami ograniczeń odpowiadających warunkom (i) oraz (ii), z których przynajmniej jedna musi być spełniona. Warunek (i) jest równoważny parze ograniczeń

$$\sum_{i \in A_1} x_i \geq 0,6, \quad (66)$$

$$\sum_{i \in A_2} x_i \leq 0,15. \quad (67)$$

Natomiast warunek (ii) możemy zapisać w postaci pary nierówności

$$\sum_{i \in B_1} x_i \leq 0,5, \quad (68)$$

$$\sum_{i \in B_2} x_i \geq 0,2. \quad (69)$$

Należy modelować dyzjunkcję „ P_1 lub P_2 ”, gdzie zdanie P_1 oznacza „Nierówności (66) i (67) są spełnione”, natomiast zdanie P_2 oznacza „Nierówności (68) i (69) są spełnione”. W tym celu wprowadzamy dwie zmienne binarne $\delta_1, \delta_2 \in \{0,1\}$, spełniające implikacje

$$\delta_1 = 1 \rightarrow P_1, \quad (70)$$

$$\delta_2 = 1 \rightarrow P_2. \quad (71)$$

Zgodnie z podejściem opisanym w punkcie 4.1.3 dla zapisania warunku (70), który jest równoważny parze implikacji

$$\delta_1 = 1 \rightarrow \sum_{i \in A_1} x_i \geq 0,6, \quad \delta_1 = 1 \rightarrow \sum_{i \in A_2} x_i \leq 0,15,$$

należy dołączyć do zadania następujące dwa ograniczenia:

$$\sum_{i \in A_1} x_i - 0,6\delta_1 \geq 0, \quad (72)$$

$$\sum_{i \in A_2} x_i + 0,85\delta_1 \leq 1. \quad (73)$$

Ograniczenie (72) wynika z (41) przy przyjęciu $m = -0,6$, natomiast (73) odpowiada nierówności (33) przy $M = 0,85$.

Podobnie można zapisać warunek (71) poprzez wprowadzenie pary ograniczeń

$$\sum_{i \in B_1} x_i + 0,5\delta_2 \leq 1, \quad (74)$$

$$\sum_{i \in B_2} x_i - 0,2\delta_2 \geq 0. \quad (75)$$

Dla zapisania dyzjunkcji „ P_1 lub P_2 ” potrzebne jest dodatkowe ograniczenie

$$\delta_1 + \delta_2 \geq 1. \quad (76)$$

Zatem pełny układ ograniczeń dla problemu optymalizacyjnego opisanego w niniejszym przykładzie zawiera następujące 6 nierówności: (65), (72), (73), (74), (75), (76).

□

4.1.7. Linearyzacja wyrażeń ze zmiennymi binarnymi

Istnieje kilka źródeł pojawiania się wyrażeń nieliniowych w zadaniach optymalizacyjnych ze zmiennymi binarnymi. Źródłem naturalnym są nieliniowe zależności, które muszą być uwzględnione w funkcji celu i ograniczeniach, związane na przykład z postacią funkcji użyteczności, efektem skali, itp. W niektórych przypadkach można się tych nieliniowości pozbyć wprowadzając dodatkowe zmienne i przekształcając zadanie (patrz np. punkty 3.1, 4.1.5). Drugim powodem

pojawiania się wyrażeń nieliniowych może być modelowanie warunków logicznych lub zabiegi związane z przekształcaniem zadania. Większość takich nieliniowości może być z zadania usunięta. Odbywa się to kosztem wprowadzania dodatkowych zmiennych i ograniczeń. W niniejszym punkcie rozważymy dwie typowe sytuacje, w których może być dokonana taka linearyzacja.

Założmy, że w funkcji celu lub ograniczeniu zadania występuje iloczyn dwóch zmiennych binarnych $\delta_1 \cdot \delta_2$. Wyrażenie takie może się pojawić w zadaniu w wyniku modelowania koniunkcji zdań P_1, P_2 , którym odpowiadają zmienne binarne δ_1, δ_2 . Iloczyn taki może być zastąpiony zmienną binarną δ po wprowadzeniu następujących trzech dodatkowych ograniczeń:

$$\begin{aligned} -\delta_1 + \delta &\leq 0 \\ -\delta_2 + \delta &\leq 0 \\ \delta_1 + \delta_2 - \delta &\leq 1. \end{aligned} \tag{77}$$

Łatwo sprawdzić, że zachodzi równoważność $\delta_1 \cdot \delta_2 = 1 \Leftrightarrow \delta = 1$. Wprowadzając sukcesywnie dodatkowe zmienne i ograniczenia, można w ten sposób linearyzować iloczyn dowolnej liczby zmiennych binarnych.

Założmy teraz, że mamy do czynienia z wyrażeniem $\delta \cdot x$, które jest iloczynem zmiennej binarnej δ i ograniczonej zmiennej ciągłej x . Wyrażenie takie można zastąpić dodatkową zmienną ciągłą y , wprowadzając następujący układ ograniczeń:

$$\begin{aligned} y - M\delta &\leq 0 \\ -x + y &\leq 0 \\ x - y + M\delta &\leq M, \end{aligned} \tag{78}$$

gdzie M jest oszacowaniem górnym zmiennej ciągłej x .

Łatwo sprawdzić, że powyższy układ nierówności gwarantuje, że $\delta \cdot x = y$.

Poniższy przykład pokazuje sytuację, w której w wyniku przekształcenia zadania pojawiają się nieliniowe wyrażenia w postaci iloczynów zmiennej ciągłej i zmiennej dyskretnej, oraz ilustruje sposób linearyzacji tych wyrażeń.

Przykład. 7. Rozważmy ponownie zadanie konstrukcji portfela stycznego, omawiane w Przykładzie 3. Przypomnijmy, że problem polega na znalezieniu portfela, dla którego minimalizowany jest iloraz wartości ryzyka przez zwrot. Wymaga to rozwiązania zadania z ilorazową funkcją celu w postaci:

$$\begin{aligned} \min \quad & \frac{R(x)}{\sum_{i=1}^n \bar{r}_i x_i} \\ & \sum_{i=1}^n x_i = 1 \\ & x_i \geq 0, \quad i = 1, \dots, n, \end{aligned} \quad (79)$$

gdzie wielkości występujące w zadaniu są zdefiniowane identycznie jak w Przykładzie 3. Załóżmy, że dodatkowo chcemy kontrolować, tak jak to robiono w Przykładzie 4, zarówno liczbę jak i minimalne udziały różnych akcji w portfelu. Wymaga to wprowadzenia (patrz Przykład 4) dodatkowych zmiennych binarnych z_i , $i = 1, \dots, n$, będących zmiennymi wskaźnikowymi dla zmiennych x_i , oraz dołączenia ograniczeń gwarantujących, że liczba różnych akcji w portfelu mieści się między k_1 a k_2 , oraz że minimalne udziały akcji wybranych do portfela są równe m . Tak wzbogacone zadanie (79) ma teraz postać

$$\begin{aligned}
 & \min \frac{R(x)}{\sum_{i=1}^n \bar{r}_i x_i} \\
 & \sum_{i=1}^n x_i = 1 \\
 & x_i - z_i \leq 0, \quad i = 1, \dots, n \\
 & x_i - m z_i \geq 0, \quad i = 1, \dots, n \\
 & k_1 \leq \sum_{i=1}^n z_i \leq k_2 \\
 & x_i \geq 0, \quad i = 1, \dots, n \\
 & z_i \in \{0, 1\}, \quad i = 1, \dots, n.
 \end{aligned} \tag{80}$$

Zadanie (80) jest zadaniem programowania liniowego mieszanego z ilorazową funkcją celu. W Przykładzie 3 pokazano, jak takie zadanie może być przekształcone do zadania z liniową funkcją celu poprzez usunięcie nieliniowości związanych z funkcją $R(x)$, a następnie wprowadzenie dodatkowej zmiennej ciągłej v , zamiany zmiennych według zależności

$$x_i \cdot v = w_i, \quad i = 1, \dots, n,$$

i dołączenie ograniczenia

$$\sum_{i=1}^n \bar{r}_i w_i = 1.$$

Jednakże zastosowanie tych zabiegów do zadania (80) rodzi pewne komplikacje. Można bowiem sprawdzić, że w wyniku omówionego wyżej postępowania (patrz Przykład 3) otrzymujemy następujące zadanie:

$$\begin{aligned}
 \min \quad & \frac{1}{L} \sum_{j=1}^L s_j \\
 & \sum_{i=1}^n (\bar{r}_i - r_{ij}) w_i - s_j \leq 0, \quad j = 1, \dots, L \\
 & \sum_{i=1}^n w_i - v = 0 \\
 & \sum_{i=1}^n \bar{r}_i w_i = 1 \\
 & w_i - z_i v \leq 0, \quad i = 1, \dots, n \\
 & w_i - m z_i v \geq 0, \quad i = 1, \dots, n \\
 & k_1 \leq \sum_{i=1}^n z_i \leq k_2 \\
 & w_i \geq 0, \quad i = 1, \dots, n \\
 & z_i \in \{0, 1\}, \quad i = 1, \dots, n \\
 & v \geq 0,
 \end{aligned} \tag{81}$$

przy czym skład portfela jest wyznaczony przez $x_i = \frac{w_i}{v}$, $i = 1, \dots, n$.

Zadanie (81) ma wprowadzić liniową funkcję celu, ale w wyniku przekształcenia $w_i = x_i \cdot v$ pojawiły się w ograniczeniach składniki $z_i \cdot v$, będące iloczynami zmiennej binarnej z_i przez zmienną ciągłą v . Chcąc rozwiązywać takie zadania przy pomocy pakietu przeznaczony do zadań programowania liniowego mieszanego, należy linearyzować wyrażenia $z_i \cdot v$, $i = 1, \dots, n$. Wymaga to wprowadzenia dodatkowych zmiennych ciągłych q_i , $i = 1, \dots, n$, i zastąpienia dla $i = 1, \dots, n$, każdej pary ograniczeń

$$w_i - z_i v \leq 0$$

$$w_i - m z_i v \geq 0$$

następującym układem ograniczeń:

$$\begin{aligned}
 w_i - q_i &\leq 0 \\
 w_i - mq_i &\geq 0 \\
 q_i - Mz_i &\leq 0 \\
 -v + q_i &\leq 0 \\
 v - q_i + Mz_i &\leq M,
 \end{aligned}
 \tag{82}$$

gdzie M jest oszacowaniem górnym zmiennej v . Dobór odpowiednio dużej wartości M ułatwia fakt, że v jest równa odwrotności zwrotu z portfela, a zatem jako M należy wziąć liczbę nie mniejszą niż odwrotność minimalnego zwrotu z portfela.

W Dodatku 5.4 zamieszczone są wydruki sformułowania oraz rozwiązania omawianego tu zadania dla przykładowych danych.

□

4.2. Metody rozwiązywania zadań programowania liniowego mieszanego

W literaturze wyróżnia się kilka grup metod, które są stosowane do rozwiązywania ogólnych zadań programowania liniowego mieszanego (PLM) lub szczególnych podklas tych zadań (patrz np. Garfinkel, Nemhauser (1978), Sysło, Deo, Kowalik (1993)). W praktyce jednak tylko jedna grupa metod jest stosowana w większości komercyjnych pakietów obliczeniowych dla zadań (PLM). Są to tak zwane metody podziału i oszacowań (patrz np. Garfinkel, Nemhauser (1978), Sysło, Deo, Kowalik (1993), Zorychta, Ogryczak (1981)). Pozostałe grupy metod mają obecnie głównie znaczenie historyczne, jak na przykład klasyczne metody odcięć, lub są stosowane do węższych klas zadań (PLM), jak na przykład metody przeglądu pośredniego lub metody programowania pseudoboolowskiego (patrz np. Garfinkel, Nemhauser (1978)). Z tego względu w niniejszym punkcie podamy jedynie podstawowe idee metody podziału i oszacowań dla zadań

(PLM) oraz przedstawimy ogólny opis komercyjnego pakietu CPLEX 3.0, zawierającego implementację tej metody.

4.2.1. Metoda podziału i oszacowań dla zadań programowania liniowego mieszanego

Metoda podziału i oszacowań jest ogólnym schematem rozwiązywania zadań programowania liniowego mieszanego. Wypiszmy raz jeszcze to zadanie w postaci macierzowej, przyjmując dla ustalenia uwagi, że mamy do czynienia z zadaniem maksymalizacji:

$$\begin{aligned} \max c^T x \\ Ax \leq b \\ l \leq x \leq u \\ x_i \in Z \text{ dla } i \in I \subseteq \{1, \dots, n\}. \end{aligned} \quad (83)$$

Zadanie, w którym pominięto ograniczenia całkowitoliczbowości, to znaczy zadanie

$$\begin{aligned} \max c^T x \\ Ax \leq b \\ l \leq x \leq u, \end{aligned} \quad (84)$$

nosi nazwę *relaksacji ciągłej zadania* (83). Jest to zadanie programowania liniowego ciągłego, które jest obliczeniowo znacznie prostsze niż zadanie (83). Idea metody podziału i oszacowań polega na rozwiązywaniu zadania (83) poprzez rozwiązanie pewnej liczby zadań programowania liniowego ciągłego typu (84) z odpowiednio dobranymi wektorami ograniczeń dolnych l i ograniczeń górnych u .

Zasadniczym elementem procedury podziału i oszacowań jest tak zwana *lista zadań oczekujących* (L). Początkowo lista ta zawiera jedynie zadanie wyjściowe (83). Dodatkowymi elementami są: rozwiązanie najlepsze x^* i wartość rozwiązania najlepszego v^* . Na po-

czątku obliczeń rozwiązanie najlepsze nie jest określone, natomiast jako wartość v^* przyjmuje się $-\infty$.

Procedura rozpoczyna się od wyboru zadania (P) z listy (L) i rozwiązania dla niego relaksacji ciągłej. Mogą zajść następujące przypadki:

- i. Rozwiązanie relaksacji nie istnieje, albo istnieje, ale jego wartość optymalna v^R spełnia warunek $v^R \leq v^*$. Wówczas zadanie jest usuwane z listy (L).
- ii. Rozwiązanie x^R relaksacji liniowej zadania (P) istnieje i spełnia warunek $v^R > v^*$. Wówczas badane jest spełnienie warunków całkowitoliczbowości:

1° Jeśli dla wszystkich indeksów ze zbioru I składowe wektora x^R są całkowitoliczbowe, to oznacza to, że zostało znalezione rozwiązanie dopuszczalne zadania (83), lepsze niż aktualnie znane rozwiązanie x^R . Wówczas rozwiązanie x^R staje się rozwiązaniem najlepszym i zostają dokonane podstawienia $x^* := x^R$, $v^* := v^R$, a zadanie (P) jest usuwane z listy (L).

2° Jeśli istnieje indeks $i \in I$, dla którego rozwiązanie x^R nie jest całkowitoliczbowe, to zadanie (P) jest usuwane z listy, a na jego miejsce na liście umieszczane są dwa zadania: zadanie (P'), które powstaje z zadanie (P) poprzez dołączenie ograniczenia $x_i \leq \lceil x_i^R \rceil$ (gdzie $\lceil x_i^R \rceil$ oznacza największą liczbę całkowitą, taką że $\lceil x_i^R \rceil \leq x_i^R$), oraz zadanie (P''), które powstaje z (P) poprzez dołączenie ograniczenia $x_i \geq \lceil x_i^R \rceil + 1$.

Powyższy krok procedury jest powtarzany tak długo, aż lista (L) stanie się pusta. Wówczas x^* jest rozwiązaniem optymalnego zadania (83), a jego wartość optymalna jest równa v^* .

Powyższy opis jest jedynie najbardziej ogólnym schematem postępowania w procedurze podziału i oszacowań. Bardzo istotnymi elementami, które muszą być określone w konkretnej realizacji metody, są zasady wyboru zadania (P) z listy (L) oraz wyboru zmiennej

i , dla której następuje opisany wyżej „podział” zadania (P) na zadania (P') i (P''). Konkretnie realizacje zawierają dużą liczbę szczegółowych rozwiązań opisanych wyżej kroków, których celem jest uzyskanie sprawnej obliczeniowo procedury (patrz np. *Using...* (1994), Zorychta, Ogryczak (1981)). Ze względu na to, że w przypadku zadań (PLM) trudno jest podać zestaw parametrów, który gwarantowałby największą efektywność równocześnie dla wszystkich możliwych zadań, część z tych parametrów może być w pakietach obliczeniowych dobierana przez użytkownika (patrz np. Zorychta, Ogryczak (1981)).

4.2.2. Opis pakietu CPLEX 3.0

Pakiet CPLEX 3.0 (patrz Zorychta, Ogryczak (1981)) jest znanym komercyjnym pakietem obliczeniowym służącym do rozwiązywania zadań programowania liniowego mieszanego w postaci

$$\begin{aligned} \max \text{ (lub min) } & \sum_{i=1}^n c_i x_i \\ & \sum_{i=1}^n a_{ij} x_i \sim b_j, \quad j = 1, \dots, m \\ & l_i \leq x_i \leq u_i, \quad i = 1, \dots, n \\ & x_i \in \mathbb{Z} \text{ dla } i \in I \subseteq \{1, \dots, n\}, \end{aligned} \quad (85)$$

gdzie \sim jest jednym z symboli $\geq, \leq, =$.

Danymi zadania są współczynniki funkcji celu $c_i, i = 1, \dots, n$, współczynniki ograniczeń $a_{ij}, i = 1, \dots, n, j = 1, \dots, m$, współczynniki prawych stron ograniczeń $b_j, j = 1, \dots, m$, oraz ograniczenia dolne i górne zmiennych zadania $l_i, u_i, i = 1, \dots, n$. Wielkości te mogą być dowolnymi liczbami rzeczywistymi. Dodatkowo ograniczenia zmiennych mogą przyjmować wartości $+\infty, -\infty$. Wielkościami wyznaczanymi przez pakiet są zmienne zadania $x_i, i = 1, \dots, n$. Zmienne x_i ,

$i \in I$, są zmiennymi całkowitoliczbowymi, pozostałe zmienne noszą nazwę zmiennych ciągłych.

W zależności od postaci zadania (85), do jego rozwiązania mogą być użyte różne programy przewidziane w pakiecie. Jeśli zadanie jest zadaniem programowania liniowego ciągłego, to znaczy $I = \emptyset$, wówczas dostępne są dwie grupy algorytmów: algorytmy oparte na metodzie simplex (patrz np. Nemhauser i in. (1989), *Using ...* (1994), Zorychta, Ogryczak (1981)) (OPTIMIZE, TRANOPT, NETOPT) oraz tak zwany algorytm barierowy (BAROPT) (patrz Nemhauser i in. (1989)). Algorytmy OPTIMIZE i TRANOPT są realizacjami odpowiednio metody prymalnej i dualnej simplex, natomiast algorytm NETOPT jest specjalizowaną wersją metody simplex dla zadań o strukturze sieciowej (patrz Nemhauser i in. (1989), *Using ...* (1994)). Wybór algorytmu jest dokonywany przez użytkownika.

W przypadku, gdy zadanie zawiera zmienne całkowitoliczbowe, tzn. $I \neq \emptyset$, wówczas zadanie jest rozwiązywane algorytmem, będącym implementacją metody podziału i oszacowań, w którym relaksacje ciągłe zadań rozwiązywane są algorytmem simplex.

1. Wprowadzanie zadania

Zadanie może być wprowadzone do pakietu na jeden z dwóch sposobów: w trybie interaktywnym albo przez wczytanie odpowiednio przygotowanego pliku z danymi zadania. Tryb interaktywny jest przewidziany dla niewielkich zadań lub do dokonywania zmian we wprowadzonym już zadaniu. Polega on na wpisywaniu zadania bezpośrednio z klawiatury z zachowaniem ustalonego formatu danych wejściowych. Większe zadania są wprowadzone poprzez wczytanie wcześniej przygotowanego pliku tekstowego zapisanego w odpowiednim formacie. W pakiecie przewidziane są dwa podstawowe formaty pliku wejściowego. Pierwszym jest tak zwany format MPS (patrz np. *Using ...* (1994)), będący tradycyjnie używanym standardem zapisu danych dla zadań programowania liniowego. Drugim jest for-

mat LP, w którym zapis zadania jest zbliżony do naturalnego zapisu algebraicznego (85). Po wprowadzeniu zadania może ono być wyświetlane na ekranie lub modyfikowane. W pakiecie dostępne są liczne komendy, pozwalające na weryfikację lub zmiany dowolnych fragmentów zapisu zadania, dodawanie i usuwanie ograniczeń, zmiennych, itp.

W pakiecie nie formułuje się żadnych ograniczeń na wielkość wprowadzanego lub rozwiązywanego zadania.

2. Rozwiązywanie zadania i analiza rozwiązania

Zadanie, które zostało poprawnie wprowadzone do pamięci, może być następnie rozwiązywane poprzez wybór jednego z wymienionych wyżej algorytmów. Użytkownik może również ustawiać liczne parametry tych algorytmów, decydujące o ich numerycznej efektywności. Przebieg rozwiązywania zadania jest wyświetlany na ekranie i zapamiętywany w pliku wyjściowym przeznaczonym na raport z działania pakietu. Może to być standardowy plik `cplex.log`, lub wskazany przez użytkownika plik, przeznaczony dla konkretnego zadania. Proces rozwiązywania może być przerywany, a następnie wznawiany przez użytkownika. Po uzyskaniu rozwiązania optymalnego wyświetlana jest informacja o wartości optymalnej zadania, czasie rozwiązywania i liczbie iteracji simpleksowych, a w przypadku zadań ze zmiennymi całkowitoliczbowymi, dodatkowo informacja o liczbie wierzchołków w drzewie rozwiązań metody podziału i oszacowań.

Uzyskane rozwiązanie może być następnie wyświetlone lub zapisane we wskazanym pliku. Istnieją też możliwości zapamiętania dodatkowych informacji o uzyskanym rozwiązaniu, pozwalające na szybsze rozwiązanie zadania w przypadku jego modyfikacji.

W przypadku zadań programowania liniowego ciągłego pakiet umożliwia również typową analizę pooptymalizacyjną uzyskanego rozwiązania. Podawane są następujące wielkości: tzw. koszty zredukowane dla zmiennych, koszty dualne dla ograniczeń oraz zakresy

zmian pojedynczych współczynników wektora funkcji celu i wektora prawych stron ograniczeń, które nie wymagają zmiany bazy optymalnej.

5. Dodatek. Przykłady obliczeń

W niniejszym punkcie przedstawione są dane i wyniki obliczeń z użyciem pakietu CPLEX 3.0 dla przykładowych zadań konstrukcji optymalnych portfeli, które były rozważane w poprzednich punktach.

Tabela 1. Stopy zwrotu z akcji

miesiąc/numer akcji	1	2	3	4	5
1	11.594	1.515	-1.579	-9.360	-0.408
2	22.078	5.299	9.947	7.578	0.000
3	-8.511	-8.000	9.446	-16.327	0.694
4	-3.488	0.395	5.164	-8.571	-3.862
5	-9.639	-9.764	-3.571	-18.243	-10.949
6	-12.000	6.195	-0.943	-11.570	-8.197
7	-10.606	-0.417	-7.619	-8.441	-6.786
8	-1.695	-6.276	11.856	-8.220	-13.725
9	-12.069	-9.091	-3.286	-10.227	-2.273
10	27.451	15.000	11.186	-3.797	2.791
11	-21.538	-4.465	6.550	-4.021	-8.140
12	-19.608	-6.132	8.750	12.676	-5.063
13	48.780	27.156	4.215	12.500	35.200
14	19.672	-2.372	7.355	15.271	12.121
15	1.370	15.190	13.889	18.000	-0.901
16	4.054	15.751	17.073	-0.847	18.548
17	-15.584	-9.810	1.042	23.884	-10.958
18	3.077	4.270	20.833	4.196	14.912
19	-4.478	-6.485	-7.543	-4.027	-5.058
20	4.688	11.679	3.263	-6.333	4.918
21	-11.940	-7.285	3.189	-18.939	-3.125
22	-1.695	-3.214	-7.506	-2.856	27.742
23	15.517	0.369	-3.580	5.882	16.026
24	5.970	-1.866	-7.250	-0.926	-2.762

Tabela 1 zawiera zaczerpnięte z *Finance ...* (1994) wartości stóp zwrotu dla pięciu spółek i dwudziestu czterech chwil obserwacji, od-

IBS *Seria*

Wspomaganie decyzji inwestycyjnych

Roman Kulikowski,
Marek Libura,
Leon Słomiński

44006

W książce omawiane są zagadnienia z obszaru analizy finansowej i teorii portfela inwestycyjnego z wykorzystaniem komputerowej metodologii wspomagającej podejmowanie decyzji.

Książka może być przedmiotem zainteresowania zarówno decydentów, podejmujących decyzje finansowe, jak i inwestorów giełdowych i doradców finansowych oraz studentów i doktorantów.

Monografia pozwoli głębiej i pełniej zrozumieć złożoną problematykę finansów i inwestycji, z uwzględnieniem różnych form ryzyka i podejmować w działalności praktycznej decyzje optymalne.

Rozważane są zasady konstruowania modeli matematycznych opisujących rynki kapitałowe – kształtowanie się cen oraz oczekiwanych zwrotów nakładów inwestycyjnych – jak również modeli działalności inwestora w postaci tzw. funkcji użyteczności.

ISBN 83-85847-09-X

W celu uzyskania bliższych informacji i zakupu dodatkowych egzemplarzy prosimy o kontakt z Instytutem Badań Systemowych PAN
ul. Newelska 6, 01-447 Warszawa
tel. 37-35-78 w. 241 e-mail: kotuszew@ibspan.waw.pl