ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1990
Seria: AUTOMATYKA z.100 Nr kol.1082

Jarosław Sikorski
Instytut Badań Systemowych PAN

O WYZNACZANIU OGRANICZEŃ ZASTĘPCZYCH W METODZIE PODZIAŁU I OSZACOWAŃ[1]

Streszczenie. W pracy dokonano przeglądu użytecznych algorytmów rozwiązywania dualnego zadania ograniczenia zastępczego w celu wyznaczenia oszacowania pierwotnej wartości optymalnej dla zadań programowania całkowitoliczbowego. Na podstawie wyników eksperymentu obliczeniowego wskazano najefektywniejsze warianty przedstawionych schematów.

1. Wprowadzenie

Metoda podziału i oszacowań jest uniwersalnym i powszechnie stosowanym schematem postępowania w algorytmach rozwiązywania zadań programowania całkowitoliczbowego. W tej metodzie pierwotny zbiór rozwiązań dopuszczalnych dzielony jest na podzbiory, na których definiowane są tzw. podproblemy. Kolejne podziały wprowadzane są w taki sposób, że powstaje struktura zwana drzewem podproblemów; podproblem utożsamiany jest z wierzchołkiem tego drzewa. Metoda jest tym efektywniejsza, im mniejsze drzewo wystarcza do znalezienia pierwotnego rozwiązania optymalnego. Rozwój drzewa zależy od skuteczności tzw. kryteriów zamykania wierzchołków. Oparte są one na rozwiązywaniu relaksacji podproblemów i uzyskiwaniu tą drogą oszacowań dla wartości optymalnych w podproblemach. W zasadzie im lepsze jest to oszacowanie i im częściej rozwiązanie relaksacji jest pierwotnie dopuszczalne, tym szybciej zatrzymywany jest rozwój drzewa podproblemów.

Jak wykazano w wielu pracach (np. [1], [6], [7], [14], [18]), wykorzystanie relaksacji z ograniczeniem zastępczym prowadzi do oszacowań ściślejszych od uzyskiwanych na podstawie innych znanych relaksacji zadań całkowitoliczbowych. Relaksacja ta jest szczególnie przydatna w przypadku rozwiązywania metodą podziału i oszacowań zadań załadunku z wieloma ograniczeniami. Zadanie zastępcze jest wtedy klasycznym zadaniem

---

[1] Praca została wykonana w ramach RP.I.02 "Teoria sterowania i optymalizacji ciągłych układów dynamicznych i procesów dyskretnych", temat 4.5 "Wybrane metody rozwiązywania zadań programowania dyskretnego".

załadunku, dla którego opracowano skuteczne w praktyce algorytmy rozwiązywania (np. [2], [8]). Poszukiwane oszacowanie wartości optymalnej w podproblemie wyznaczane jest poprzez rozwiązywanie dokładne lub przybliżone odpowiedniego zadania dualnego. Zatem, aby można było w praktyce wykorzystać dobrą jakość oszacowań uzyskiwanych z zastosowaniem ograniczeń zastępczych, potrzebne są efektywne algorytmy rozwiązywania tego zadania dualnego.

2. Podstawowe sformułowania

Rozważmy pewien wierzchołek drzewa podproblemów. Niech będzie w nim badane binarne zadanie liniowe z wieloma ograniczeniami zwane w literaturze wielowymiarowym zadaniem załadunku:

$$v(P) = \max \{ cx : Ax \leq b, \; x_i = 0 \text{ albo } 1 \text{ ,dla } i=1,\dots,n \}, \qquad (P)$$

gdzie $c \in Z^n_+$, A jest macierzą mxn o elementach $a_{ki} \in Z_+$ i $b \in Z^m_+$ (Z_+ oznacza zbiór dodatnich liczb całkowitych). Najczęściej w praktyce $m \ll n$.

Dla danego mnożnika $\omega \in \mathbb{R}^m_+$ można zdefiniować relaksację zadania pierwotnego, która jest klasycznym zadaniem załadunku z jednym ograniczeniem.

$$h(\omega) = \max \{ cx : \omega(Ax-b) \leq 0 \text{ , } x_i \text{ j.w. } \}. \qquad (Z)$$

W ten sposób określona zostaje funkcja dualna $h: \mathbb{R}^m_+ \longrightarrow Z_+$. Pojedyncze ograniczenie w zadaniu (Z) nazywane jest zastępczym. Ze sposobu wprowadzenia relaksacji (Z) wynika, że $h(\omega) \geq v(P)$ dla każdego $\omega \in \mathbb{R}^m_+$.

Zadanie dualne ograniczenia zastępczego reprezentuje poszukiwanie takiego mnożnika ograniczenia zastępczego ω, dla którego odstęp pomiędzy wartościami h(ω) i v(P) jest minimalny. Ma ono postać:

$$v(D) = \min \{ h(\omega) : \omega \in \mathbb{R}^m_+ \}. \qquad (D)$$

Wartości optymalne w zadaniach (P) i (D) są zatem w relacji:

$$v(D) \geq v(P).$$

Oznacza ona, że rozwiązanie zadania dualnego ograniczenia zastępczego może być wykorzystywane jako oszacowanie od góry pierwotnej wartości optymalnej. Jak wykazano w [1], [7] i [14], jest ono lepsze od oszacowań wyznaczanych na podstawie ciągłej relaksacji lub zadania dualnego Lagrange'a.

Poniżej zebrano najistotniejsze właściwości zadania (D) [1], [6], [7].

(W1) $v(D) = v(P)$ wtedy i tylko wtedy, gdy istnieje takie $\hat{\omega} \in \mathbb{R}^m_+$ i takie $\hat{x}$ dopuszczalne dla (P), że $h(\hat{\omega}) = c\hat{x}$; wtedy $\hat{x}$ jest rozwiązaniem optymalnym dla (P) a $\hat{\omega}$ - dla (D)

(W2) Funkcja h jest półciągła z góry i quasi-wypukła; jest obszarami stała i ma charakter "schodkowy".

(W3) Wartość v(D) jest osiągana przez funkcję h na wypukłym lub otwartym stożku o wierzchołku w $0 \in \mathbb{R}^m$.

(W4) $h(\mu\omega) = h(\omega)$ dla $\mu > 0$, gdyż skalowanie mnożników ω nie zmienia zbioru dopuszczalnego w zadaniu (Z).

(W5) Jeśli $\tilde{x}$ jest rozwiązaniem optymalnym dla zadania (Z), to wektor $g \triangleq b - A\tilde{x} \in \mathbb{Z}^m_+$ jest quasi-subgradientem funkcji h w punkcie $\omega \in \mathbb{R}^m_+$; oznacza to, że dla wszystkich $\upsilon \in \mathbb{R}^m_*$ zachodzi: $\upsilon g \geq \omega g \Rightarrow h(\upsilon) \geq h(\omega)$.

Jak wynika z (W2), nie istnieje subgradient funkcji dualnej w każdym z punktów jej dziedziny. Można jednak wykorzystywać wektor o zbliżonych, choć słabszych, właściwościach (W5). (W3) uzasadnia użycie w definicji zadania (D) symbolu *min* w miejsce bardziej ogólnego *inf* (por. [14]). (W4) pozwala na modyfikowanie zadania dualnego poprzez arbitralne normalizacje mnożników, które mogą prowadzić do uzyskiwania ograniczonych zbiorów rozwiązań dopuszczalnych.

3. Algorytm quasi-subgradientowy

W metodach minimalizacji funkcji wypukłych podstawowe znaczenie ma pojęcie subgradientu. Dla funkcji quasi-wypukłych zdefiniowano pojęcie quasi-subgradientu, które uogólnia pojęcie subgradientu. Jeden z quasi-subgradientów funkcji dualnej ograniczenia zastępczego w danym punkcie można łatwo wyznaczyć na podstawie (W5), przy okazji rozwiązywania odpowiedniej relaksacji (Z).

Prezentowany w tym punkcie algorytm rozwiązywania zadania (D) naśladuje prosty schemat optymalizacji nieróżniczkowalnej, w którym kierunkiem poszukiwań jest subgradient. Zadanie dualne rozwiązywane jest w zmodyfikowanej postaci wykorzystującej właściwość (W4).

$$v(D) = \min \{ h(\omega) : \omega \in S^2 \}, \qquad (D')$$

$$\text{gdzie} \quad S^2 = \{ \omega \in \mathbb{R}^m_+ : \|\omega\|_{L_2} = 1 \}.$$

W algorytmie wygodnie jest wykorzystywać kierunki styczne do sfery jednostkowej w kolejno wyznaczanych punktach. Uzyskiwane są one przez rzutowanie wektora przeciwnego do quasi-subgradientu na odpowiednią hiperpłaszczyznę styczną, co dane jest wzorem:

$$d = -g + (g^T\omega)\omega ,$$

gdzie d oznacza wyznaczony kierunek, a g - quasi-subgradient funkcji h w punkcie ω. Jak wykazano w [15], wektor d jest także quasi-subgradientem funkcji dualnej w punkcie $\omega \in S^2$.

W poniższym schemacie algorytmu quasi-subgradientowego dla zadania (D') przyjęto, że $QK(\omega;\tilde{x})$ symbolizuje procedurę, która dla danego mnożnika ω podaje jedno z rozwiązań optymalnych relaksacji (Z) (oznaczane przez $\tilde{x}$).

ALGORYTM QS

begin

wybierz $\omega \in S^2$; $q := +\infty$; wybierz $j_{max} \geq 1$;

for $j := 1$ **to** j_{max} **do**

begin

$QK(\omega ; \tilde{x})$;

$q := \min\{c\tilde{x}, q\}$; $g := b - A\tilde{x}$; $d := -g + (g \cdot \omega)\omega$;

wybierz $t > 0$;

for $k := 1$ **to** m **do**

$\bar{\omega}_k := \max\{0 , \omega_k + t \cdot d_k / \|d\|_{L_2}\}$; $\omega := \bar{\omega}/\|\bar{\omega}\|_{L_2}$

end

end.

Na wstępie należy wybrać startowy mnożnik należący do sfery jednostkowej i ciąg współczynników kroku. Rozwiązywanie zadania (Z) dla aktualnego mnożnika pozwala wyznaczać wartość i quasi-subgradient funkcji dualnej zgodnie z (W5). Na tej podstawie wyznaczany jest kierunek poszukiwań d oraz oszacowanie dualnej wartości optymalnej q. Wszystkie mnożniki ω wyznaczane w Algorytmie QS należą do zbioru dopuszczalnego S^2.

Ciąg $\{q^j\}$ (j - numer iteracji w zewnętrznej pętli **for**) wyznaczany w wyżej opisanym algorytmie jest zbieżny do $v(0)$, jeśli ciąg współczynników kroku $\{t^j\}$ zbiega do zera i ma rozbieżny szereg. Wynika to z ogólniejszego twierdzenia podanego w [15] dla schematu, którego szczególnym przypadkiem jest Algorytm QS. Jednakże w praktyce, tak jak i dla prostego algorytmu subgradientowego, stosowanie tego typu ciągów współczynników kroku nie daje zadowalających rezultatów. Dlatego przy rozwiązywaniu zadania (D') w trakcie eksperymentu obliczeniowego zastosowano dwa inne rodzaje ciągów $\{t^j\}$ (por. [8]). Jeden geometryczny: $t^{j+1} = \gamma \cdot t^j$, gdzie należy wybrać $t^1 > 0$ oraz $1 > \gamma \gg 0$, a drugi "pierwiastkowy": $t^{j+1} = t^1 \cdot \sqrt{1 + (1-j)/j_{max}}$. Startowy mnożnik ograniczenia zastępczego wybierano na dwa sposoby: standardowo jako $\omega_k = 1/\sqrt{m}$ lub w oparciu o parametry zadania jako $\omega_k = \max\left[0 , \left[\sum_{i=1}^{n} a_{ki} - b_k\right] \Big/ \sum_{i=1}^{n} a_{ki}\right]$ (z dodatkowym unormowaniem), dla $k=1,\ldots,m$. Wprowadzano także modyfikację kierunku d poprzez przyjęcie $d_k = 0$, gdy $\omega_k = 0$ i $g_k > 0$. Miała ona na celu zachowywanie zerowych składowych mnożnika dla tych ograniczeń pierwotnych, które nie były naruszane przez rozwiązanie aktualnej relaksacji, mimo że nie zostały uwzględnione w definiującym ją ograniczeniu zastępczym. Powstało w ten sposób 8 wariantów algorytmu:

Alg.	kierunek d		startowe ω		współczynnik t	
	modyf.	stand.	param.	stand.	geom.	pierw.
QS1	×		×		×	
QS2	×		×			×
QS3	×			×	×	
QS4	×			×		×
QS5		×	×		×	
QS6		×	×			×
QS7		×		×	×	
QS8		×		×		×

W trakcie eksperymentu obliczenia w Algorytmie QS przerywano nie tylko, gdy w pętli for wykonano j_{max} iteracji, ale także, gdy liczba kolejnych iteracji bez zmniejszenia wartości q przekroczyła zadaną wartość.

W każdej iteracji przedstawionego schematu rozwiązywane jest zadanie zastępcze (Z) . Dla zadania pierwotnego będącego zadaniem załadunku z wieloma ograniczeniami jest ono klasycznym zadaniem załadunku. Opracowano dla niego skuteczne w praktyce algorytmy rozwiązywania (np. [2],[9]), co upoważnia do używania symbolicznie zapisanej procedury QK.

4. Algorytm heurystyczny

Rozważmy szczególny przypadek binarnego zadania załadunku (P), w którym występują tylko dwa ograniczenia pierwotne: m = 2. Zadanie zastępcze (Z) powstaje przez wprowadzenie dwuwymiarowego mnożnika $\omega = [\omega_1, \omega_2]$. Jeśli do rozwiązywania odpowiedniego zadania (D) zostanie zastosowany algorytm zanikającego wielościanu [1],[17], to nabierze on, w porównaniu z ogólnym przypadkiem zadania pierwotnego, korzystnych cech ułatwiających obliczenia numeryczne. Po pierwsze, łatwo będzie redukować opis kolejnych wielościanów, pozbywając się nieaktywnych prostych odcinających. Dzięki temu opis wielościanów nie będzie się zmieniał ilościowo w kolejnych iteracjach i pozostanie zawsze prosty. Po drugie, nie będzie potrzebne rozwiązywanie pomocniczych zadań liniowych [17].

Oznaczmy algorytm zanikającego wielościanu zastosowany do zadania (P) z dwoma ograniczeniami przez ZW2 i zapiszmy go symbolicznie w postaci procedury

$$ZW2(\, a^1,\ a^2,\ b_1,\ b_2\ ;\ \omega_1,\ \omega_2,\ \tilde{x},\ q\ ).$$

Parametrami wejściowymi są wektory współczynników dwóch ograniczeń pierwotnych – a^1 i a^2, oraz prawe strony tych ograniczeń – b_1 i b_2. Parametrami wyjściowymi są dwie składowe optymalnego mnożnika ograniczenia zastępczego – ω_1 i ω_2, jedno z optymalnych rozwiązań relaksacji utworzonej dla tego mnożnika – $\tilde{x}$ oraz przybliżenie wartości v(D) – q.

Przedstawiony poniżej algorytm rozwiązywania zadania dualnego (D) dla zadań pierwotnych z wieloma ograniczeniami wykorzystuje ideę, która po raz pierwszy została zaproponowana w [5]. Rozwinięto ją i wykorzystano w [4]. Polega ona na wielokrotnym rozwiązywaniu zadania dualnego utworzonego dla dwóch ograniczeń pierwotnych w celu uniknięcia trudności związanych z większą ich liczbą. Ograniczenie zastępcze wyznaczone w taki sposób nie jest najczęściej ograniczeniem optymalnym w sensie zadania (D).

Algorytm realizujący tę heurystyczną metodę przedstawiono poniżej. Wykorzystano w nim procedurę ZW2 dla wyznaczania optymalnego ograniczenia zastępczego w zadaniu z dwoma ograniczeniami pierwotnymi oraz dwie procedury wyboru jednego ograniczenia pierwotnego z podanej listy: Choose0 i Choose. Dla tych dwóch procedur parametrem wejściowym jest lista indeksów P, a parametrem wyjściowym - indeks wybranego ograniczenia k. Wiersze macierzy ograniczeń A oznaczono przez $a^1,\dots,a^m$, a odpowiadające im składowe wektora prawych stron przez $b_1,\dots,b_m$. Procedura QK jest taka sama jak w Algorytmie QS. Przyjęto dodatkowe oznaczenie $e^k \triangleq [0,\dots,1,\dots,0] \in \mathbb{R}^m$, gdzie k-ta składowa jest równa 1.

ALGORYTM W

<u>begin</u>

$P := \{1,\dots,m\}$; Choose0(P ; k) ; $\omega := e^k$;

QK(ω ; $\tilde{x}$) ; $q := c\tilde{x}$; $\bar{a} := a^k$; $\bar{b} := b_k$; $P := P \setminus \{k\}$;

<u>repeat</u>

Choose(P ; k) ;

<u>if</u> $a^k x > b_k$ <u>then</u>

<u>begin</u>

ZW2($\bar{a}$, a^k, $\bar{b}$, b_k ; $\bar{\upsilon}$, υ^k, x, q) ;

$\bar{a} := \bar{\upsilon}\cdot\bar{a} + \upsilon^k\cdot a^k$; $\bar{b} := \bar{\upsilon}\cdot\bar{b} + \upsilon^k\cdot b_k$; $\omega := \bar{\upsilon}\cdot\omega + \upsilon^k\cdot e^k$

<u>end</u> ;

$P := P \setminus \{k\}$

<u>until</u> $P = \emptyset$

<u>end.</u>

W każdej iteracji pętli <u>repeat</u> na liście P znajdują się te ograniczenia, które nie zostały jeszcze uwzględnione w utworzonym ograniczeniu zastępczym. W rezultacie działania procedury ZW2 tworzone jest następne ograniczenie zastępcze, w którym uwzględniono o jedno ograniczenie pierwotne więcej. Po zakończeniu obliczeń w Algorytmie W wyznaczone zostaje heurystyczne rozwiązanie zadania (D) podane przez mnożnik ograniczenia zastępczego ω i wartość q. Jest ona przybliżeniem wartości v(D) i spełnia nierówność:

$$q \geq v(D) \geq v(P).$$

Warianty Algorytmu W powstają przy wykorzystaniu różnych sposobów wyboru ograniczeń pierwotnych w procedurach Choose0 i Choose; w trakcie eksperymentu obliczeniowego badano cztery takie warianty.

ALGORYTM W1

W procedurze Choose0 jest rozwiązywane m klasycznych zadań załadunku dla każdego ograniczenia pierwotnego osobno. Uzyskane wartości optymalne są sortowane w porządku niemalejącym. W Choose0 wskazywane jest ograniczenie pierwsze w kolejności. Procedura Choose wskazuje w każdej iteracji następne w kolejności ograniczenie pierwotne.

ALGORYTM W2 [4]

W procedurze Choose0 jest rozwiązywane m klasycznych zadań załadunku dla każdego ograniczenia pierwotnego osobno. Uzyskane wartości optymalne są porównywane i wybierana jest najmniejsza. W Choose0 wskazywane jest odpowiadające jej ograniczenie pierwotne. W procedurze Choose wskazywane jest ograniczenie najbardziej naruszone przez rozwiązanie aktualnej relaksacji typu (Z) dla dwóch ograniczeń w zadaniu pierwotnym.

ALGORYTM W3

W procedurze Choose0 wyznaczane są naruszenia wszystkich ograniczeń pierwotnych przez trywialne rozwiązanie zadania (P) bez ograniczeń (x_i = 1, dla i=1,...,n). Uzyskane wartości naruszeń są sortowane w porządku nierosnącym. W Choose0 wskazywane jest ograniczenie pierwsze w kolejności. Procedura Choose wskazuje w każdej iteracji następne w kolejności ograniczenie pierwotne.

ALGORYTM W4

W procedurze Choose0 wyznaczane są naruszenia wszystkich ograniczeń pierwotnych przez trywialne rozwiązanie zadania (P) bez ograniczeń (x_i = 1, dla i=1,...,n). Uzyskane wartości naruszeń są porównywane i wybierana jest największa. W Choose0 wskazywane jest odpowiadające jej ograniczenie pierwotne. W procedurze Choose wskazywane jest ograniczenie najbardziej naruszone przez rozwiązanie aktualnej relaksacji typu (Z).

Algorytm heurystyczny W stosowany może być w zadaniach załadunku typu (P), w których m > 2. W przypadku gdy m = 2, należy do rozwiązywania odpowiednich zadań (D) stosować wprost Algorytm ZW2.

5. Eksperyment obliczeniowy

Obliczenia wykonane dla zadań testowych wybranych z literatury miały na celu sprawdzenie praktycznej przydatności przedstawionych algorytmów wyznaczania górnych oszacowań pierwotnej wartości optymalnej w zadaniach załadunku z wieloma ograniczeniami. Dla każdego z zadań testowych wyznaczono przybliżenie rozwiązania optymalnego w zadaniu dualnym (D)

Algorytmami H1÷H4 oraz rozwiązano zadanie (D') Algorytmami QS1÷QS8. Pozwoliło to na wskazanie najskuteczniejszych wariantów algorytmu quasi-subgradientowego i schematu heurystycznego oraz na porównanie rezultatów uzyskanych tymi dwoma metodami.

Zadania testowe o postaci (P) zostały wybrane z prac: [3] (oznaczono je FLEIS), [10] (oznaczono je PETE1÷PETE7), [11] (oznaczono je PLAT1 i PLAT2), [12] (oznaczono je SETO1 i SETO2), [13] (oznaczono je SHIO1÷SHI30) oraz [19] (oznaczono je WEIN1÷8). Poniżej podano ich rozmiary.

Poz. lit.	Nazwa	n	m
[19]	WEIN1÷6 WEIN7,8	28 105	2 2
[11]	PLAT1 PLAT2	28 35	4 4
[10]	PETE6 PETE7	39 50	5 5
[13]	SHI01÷05 SHI06÷09 SHI10÷13 SHI14÷17	30 40 50 60	5 5 5 5

Poz. lit.	Nazwa	n	m
[13]	SHI18÷21 SHI22÷25 SHI26÷30	70 80 90	5 5 5
[3]	FLEIS	20	10
[10]	PETE1 PETE2 PETE3 PETE4 PETE5	6 10 15 20 28	10 10 10 10 10
[12]	SETO1,2	60	30

Tab.1

Wszystkie obliczenia testowe przeprowadzono na mikrokomputerze zgodnym z IBM AT, pracującym z zegarem 10 MHz i wyposażonym w koprocesor arytmetyczny. Programy napisano w języku Pascal. Do rozwiązywania klasycznych zadań załadunku z jednym ograniczeniem w algorytmach QS i ZW2 (symboliczna procedura QK) wykorzystano procedurę RKnap z modułu SOLSACK [6]. W algorytmie QS przyjęto $j_{max} = 40$, $t^1 = 2.0$ i $\gamma = 0.85$ dla geometrycznie zbieżnego ciągu współczynników kroku oraz $j_{max} = m + 6$ i $t^1 = 2.0$ dla ciągu "pierwiastkowego". Ponadto zatrzymywano w nim obliczenia, gdy liczba iteracji bez poprawy wartości q przekraczała $5 + \lfloor m / 10 \rfloor$.

W poniższej tabeli przedstawiono wyniki eksperymentu. Zawiera ona wyznaczone w całej grupie zadań testowych średnie wartości wybranych parametrów: względnej wielkości odstępu oszacowania od pierwotnej wartości optymalnej $(100\% \cdot [q - v(P)]/v(P) - \bar{p}$, liczby rozwiązanych relaksacji (Z) - $\bar{z}$, całkowitego czasu obliczeń - $\bar{t}$ (w sek) oraz czasu zużytego na rozwiązywanie relaksacji - $\bar{s}$ (w sek.). Podano także różnicę czasów $\bar{t}$ i $\bar{s}$, która charakteryzuje nakład obliczeń związany z cechami algorytmu rozwiązywania zadania (D), a nie z jakością algorytmu rozwiązywania relaksacji (Z). Z powodów podanych w podsumowaniu zamieszczono dodatkowo wartości $\bar{t}'$ i $\bar{s}'$ wyznaczone z pominięciem zadań PETE6 i 7 oraz PLAT1 i 2.

Alg.	$\bar{p}$	$\bar{z}$	$\bar{t}$	$\bar{t}'$	$\bar{s}$	$\bar{s}'$	$\bar{t}-\bar{s}$
QS1	0.65	13	8.30	5.67	6.15	3.26	2.15
QS2	0.93	11	6.05	4.65	4.23	2.62	1.82
QS3	0.66	12	8.26	4.72	6.57	2.87	1.69
QS4	0.94	11	6.60	4.61	4.83	2.61	1.77
QS5	0.66	12	7.42	5.18	5.51	3.04	1.91
QS6	0.75	11	6.59	4.63	4.74	2.55	1.85
QS7	0.69	12	7.56	5.23	5.70	3.16	1.86
QS8	0.78	11	6.92	4.81	5.11	2.56	1.81
H1	0.85	23	82.42	6.23	81.55	5.44	0.87
H2	0.90	24	26.05	6.43	25.18	5.62	0.87
H3	0.93	13	79.31	3.59	78.58	2.94	0.73
H4	0.86	13	32.75	3.77	31.98	3.07	0.77

Tab.2

6. Podsumowanie

Na podstawie Tabeli 2 można stwierdzić, że geometrycznie zbieżny ciąg współczynników kroku w Algorytmie QS, przy przyjętych w teście wartościach parametrów wyjściowych, pozwala uzyskiwać oszacowania lepsze niż ciąg "pierwiastkowy". Modyfikacja kierunku poszukiwań nie poprawia skuteczności tego algorytmu. Zmiana sposobu wybierania startowego mnożnika ograniczenia zastępczego nie wpływa praktycznie na jakość uzyskiwanych rezultatów. Wśród badanych wariantów algorytmu quasi-subgradientowego skutecznością wyróżniają się QS3 i QS5.

Wśród algorytmów heurystycznych najlepsze oszacowania pierwotnej wartości optymalnej uzyskiwano Algorytmami H1 i H4. Oszacowania, które wyznaczano używając wariantu QS3 lub QS5, były lepsze od najlepszych oszacowań heurystycznych. Potwierdza to przybliżony charakter rozwiązań zadania dualnego, które wyznaczane są według schematu H. Algorytmy H3 i H4 charakteryzują się wyraźnie mniejszą liczbą rozwiązanych relaksacji. W przypadku zadań pierwotnych o mniejszej liczbie ograniczeń aktywnych w wyznaczonym ograniczeniu zastępczym (np. PETE1+5, FLEIS, SHIO1+30) liczba ta jest mniejsza od liczby relaksacji przyciętnie rozwiązywanych w algorytmie QS. Dla zadań o większej liczbie ograniczeń tworzących wynikowe ograniczenie zastępcze (np. PETE6 i 7, PLAT1 i 2, SETO1 i 2) relacja ta jest odwrotna i trzeba stwierdzić, że przedstawiona metoda heurystyczna jest mniej przydatna w takim przypadku. Biorąc pod uwagę jakość oszacowań i liczbę rozwiązanych relaksacji, można stwierdzić, że Algorytm H4 jest najlepszym wariantem metody heurystycznej. Jego czas $\bar{t}'$ należy do najmniejszych spośród wszystkich badanych algorytmów.

Należy jeszcze zwrócić uwagę na wystąpienie długich czasów rozwiązywania relaksacji budowanych w algorytmach heurystycznych dla zadań PETE6 i 7 oraz PLAT1 i 2 - stąd duże wartości $\bar{t}$ i $\bar{s}$ w Tabeli 2. Powodem jest pojawianie się w pierwszych iteracjach tych algorytmów zadań załadunku o silnie skorelowanych danych (por. [16]). Taki jest bowiem charakter najbardziej "napiętych" ograniczeń pierwotnych w wymienionych zadaniach. W metodzie heurystycznej ograniczenia zastępcze są początkowo budowane wyłącznie w oparciu o te najbardziej "napięte" ograniczenia. Natomiast przy wykorzystaniu Algorytmu QS to zjawisko nie występuje, bowiem zastosowane w nim mnożniki startowe powodują od początku działania algorytmu konstruowanie ograniczeń zastępczych, które uwzględniają wszystkie lub prawie wszystkie ograniczenia pierwotne.

LITERATURA

[1] Dyer M.E.: Calculating surrogate constraints. Mathematical Programming 19 (1980) 255-278.
[2] Fayard D., Plateau G.: An algorithm for the solution of the 0-1 knapsack problem. Computing 28 (1982) 269-267
[3] Fleischer J., Sigmap Newsletter 20 (1976).
[4] Gavish B., Pirkul H.: Efficient algorithms for solving multiconstraint zero-one knapsack problems to optimality. Mathematical Programming 31 (1985) 78-105.
[5] Glover F.: A multiphase-dual algorithm for the zero-one integer programming problem. Operations Research 13 (1965) 879-919.
[6] Glover F.: Surrogate constraints. Operations Research 16 (1968) 741-749.
[7] Greenberg H.J., Pierskalla W.P.: Surrogate mathematical programs. Operations Research 18 (1970) 924-939.
[8] Held M., Wolfe P., Crowder H.P.: A validation of subgradient optimisation. Mathematical Programming 6 (1974) 62-88.
[9] Martello S., Toth P.: A new algorithm for the 0-1 knapsack problem. Management Science 5 (1988) 633-644.
[10] Petersen C.C.: Computational experience with variants of the Balas algorithm applied to the selection of R and D projects. Management Science 13 (1967) 736-750.
[11] Plateau G.: Reduction de la taille des problemes lineaires en variables 0-1. Research Report 71 (1976) UST Lille 1.
[12] Senju S., Toyoda Y.: An approach to linear programming with 0-1 variables. Management Science 15 (1968) 196-207.
[13] Shih W.: A branch and bound method for the multiconstraint 0-1 knapsack problem. Journal of Operations Research Society 30 (1979).
[14] Sikorski J.: Dualne zadanie wyznaczania najlepszego ograniczenia zastępczego. Archiwum Automatyki i Telemechaniki 4 (1984) 471-481.
[15] Sikorski J.: Quasi-subgradient algorithms for calculating surrogate constraints. In: Malanowski K.,Mizukami K. (eds.) Lecture Notes in Control and Information Sciences v.82 Springer Verlag (1986) 203-236.
[16] Sikorski J.: Moduł SOLSACK. Procedury rozwiązywania binarnych zadań załadunku. Opracowanie wewnętrzne ZPM-5 (1989) IBS PAN.
[17] Sikorski J.: Szybki algorytm dla wyznaczania ograniczenia zastępczego w zadaniach załadunku z dwoma ograniczeniami. Opracowanie wewnętrzne ZPM-19 (1989) IBS PAN.
[18] Sikorski J.: Eksperyment obliczeniowy z heurystycznym algorytmem wyznaczania oszacowania wartości optymalnej w zadaniu załadunku z wieloma ograniczeniami. Opracowanie wewnętrzne ZPM-27 (1989) IBS PAN.
[19] Weingartner H.M.,Ness D.N.: Methods for the solution of the multidimensional 0-1 knapsack problem. Operations Research 15 (1967) 83-103.

Recenzent: Prof.dr hab.inż.J.Błażewicz

Wpłynęło do Redakcji do 1990-04-30.

ON CALCULATING EQUIVALENT CONSTRAINTS IN THE BRANCH-AND-BOUND METHOD

S u m m a r y

A survey of the algorithms of solving the equivalent dual problem which are useful for calculating upper bounds of the optimal value in discrete programming problems is presented. Results of the computational experiment performed to compare several variants of the algorithms are summarized.

О ВЫЧИСЛЕНИИ ЗАМЕНИМЫХ ОГРАНИЧЕНИЙ В МЕТОДЕ РАЗДЕЛЕНИЯ И СЦЕНКИ

Р е з ю м е

В работе проведен обзор пригодных алгоритмов решения двойственной задачи заменимого ограничения, чтобы определить оценку первоначального оптимального значения для задач целочислительного програмирования. На основании результатов расчетного эксперимента указаны самые эффективные варианты представленых схем.