ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1984
Seria: AUTOMATYKA z. 74 Nr kol. 810

Andrzej Świerniak, Mariusz Słoma
Politechnika Śląska

ALGORYTM MINIMAKSOWEGO ROZDZIAŁU ZASOBÓW W SYSTEMIE

Streszczenie : W pracy przedstawiono algorytm określania optymalnego rozdziału zasobów w przypadku niepełnej informacji o dopływach dyskretnych zasobów do systemu . Zakładając, że znane jest ograniczenie na wielkość tych dopływów proponuje się minimaksowe kryterium optymalizacji oraz algorytm bazujący na standardowych procedurach programowania matematycznego.

1. Wstęp

Dynamiczny problem rozdziału zasobów w systemie można sformułować następująco :
Dane jest równanie ewolucji zasobów w systemie

$$x(n+1) = A\,x(n) - B\,u(n) + d(n) \qquad n = 0,1, \dots , N-1 \tag{1}$$

$$x(0) = x_0 > 0$$

gdzie: x jest k-wymiarowym wektorem zasobów, u - m-wymiarowym wektorem decyzji o przydziałach zasobów na poszczególne zadania /podsystemy/ , d - wektorem dopływów zasobów do systemu. Stałe $0 \leq a_{ii} \leq 1$, $0 \leq 1-b_{ij} \leq 1$ określają odpowiednio stopień deprecjacji zasobów i zużycia zasobów przydzielonych, zaś a_{ij} $/i \neq j/$ - określają strukturę przepływów między magazynami lub różnymi rodzajami zasobów .
Oznaczono przy tym $A = \{a_{ij}\}$, $B = \{b_{ij}\}$.
Określić rozdział zasobów w horyzoncie N znajdując ciąg decyzji u(n) , n = 0,1, ... , N-1 minimalizujący wskaźnik

$$I = \sum_{n=0}^{N-1} L_n\,(u(n)\,,\,x(n+1)) \tag{2}$$

przy ograniczeniach $0 \leq u_j \leq U_j(x)$, $0 \leq x_i \leq X_i$

$$i = 1,2, \dots , k \ ; \ j = 1,2, \dots , m$$

2. Gwarantowany wskaźnik jakości i zastosowanie programowania dynamicznego

Wyznaczanie decyzji optymalnych wymagałoby znajomości stanu układu, tzn. wektora zasobów w chwili n-tej oraz wszystkich dopływów w przyszłości, tzn. dla $i = n, n+1 \dots N-1$. Wymaganie to jest nierealne. W pracy będziemy zakładali, że potrafimy określić wektor $x(n)$ oraz oszacować maksymalną wielkość dopływu zasobów do systemu, tzn. :

$$\sum_{i=1}^{k} d_i(n) \leqslant w(n) \tag{3}$$

W związku z tym zadanie minimalizacji wskaźnika (2) należy zastąpić zadaniem wyznaczania strategii $\{h_n(x(n))\}$ takiej, że ciąg

$$\{h_o(x_o), \dots, h_{N-1}(x(N-1))\} = H_{N-1} = \{u(0), \dots, u(N-1)\} \tag{4}$$

minimalizuje gwarantowany wskaźnik jakości o postaci :

$$\tilde{I} = \sup_{D_N \in W} I \tag{5}$$

gdzie: przez D_N oznaczono ciąg niepewnych dopływów, tzn. :

$$D_N = \{d(0), \dots, d(N-1)\} \tag{6}$$

a W oznacza zbiór wynikający z ograniczeń (3).
Dodatkowo zakładać będziemy, że dopływy mogą przyjmować jedynie dyskretne wartości, ekstremum wskaźnika będzie zatem osiągane.
Rozwiązanie zadania minimalizacji wskaźnika (5) otrzymamy stosując programowanie dynamiczne.
Oznaczmy :

$$I^o = \inf_{H_N} \tilde{I} = \inf_{H_N} \sup_{D_N \in W} I = \inf_{u(0)} \sup_{|y_o, u(0)} \left\{ \inf_{u(1)} \sup_{|y_1, u(1)} \left(\dots \left(\inf_{u(N-1)} \sup_{|y_{N-1}, u(N-1)} I \right) \right) \right\}$$

gdzie; y_n oznacza wektor dostępnej informacji /tzn. przy założeniu dokładnej informacji o zasobach / :

$$y_n = \left(x_o^T, x^T(1) \dots x^T(n), u^T(0), \dots, u^T(n-1)\right) \qquad n = 0, 1, \dots, N-1$$

Ze względu na addytywną postać wskaźnika (2) możemy określić następujący ciąg rekurencyjny

$$E_N(x(N-1),\ u(N-1)) = \sup_{x(N)\in X_N \mid x(N-1),u(N-1)} \{ L_{N-1}(u(N-1),\ x(N))\} \tag{7}$$

$$Q_n(x(n)) = \inf_{u(n)} E_{n+1}(x(n),\ u(n)) \qquad n = 0,1,\ \dots\ ,N-1 \tag{8}$$

$$E_n(x(n-1),u(n-1)) = \sup_{x(n)\in X_n \mid x(n-1),u(n-1)} \{ Q_n(x(n)) + L_{n-1}(u(n-1),\ x(n))\}$$

$$n = 0,1,\ \dots\ ,N-1 \tag{9}$$

Zbiory X_n , po których dokonuje się maksymalizacji określone są jako :

$$X_{n+1|x(n),u(n)} = \left\{ x(n+1) :\ \sum_{i=1}^{k} \{ x_i(n) - \sum_{j=1}^{k} a_{ij}\, x_j(n) + \sum_{j=1}^{m} b_{ij}\, u_j(n)\} \le \le w(n) \right\} \tag{10}$$

Zauważmy, że

$$I^o = Q_o(x_o)$$

oraz sterowanie $u(n)$ zależy jedynie od aktualnego wektora zasobów $x(n)$. Stan $x(n)$ jest zatem informacją wystarczającą [1] w zadaniu minimalizacji gwarantowanego wskaźnika jakości i wśród strategii H_{N-1} istnieje strategia optymalna .

Założenie o osiąganiu ekstremów (zbiory x i u są wypukłe i zwarte a zbiór W skończony i zwarty) umożliwia zastąpienie relacji (7) - (10) równaniem :

$$Q_n(x(n)) = \min_{u(n)} \max_{d(n)|x(n),u(n)} \{ Q_{n+1}(x(n+1)) + L_n(u(n),\ x(n+1))\} \tag{11}$$

W każdym etapie procesu decyzyjnego należy zatem rozwiązać statyczny problem minimaksu dyskretnego dla ustalonych punktów siatki wektora zasobów $x(n)$. Przyjmując bowiem $x(n)$ jako ustalone i biorąc pod uwagę równanie ewolucji zasobów (1) mamy w każdym etapie problem znajdowania sterowania $u^o(n)$ takiego, że

$$\min_{u(n)} \max_{d(n)} F_n(u(n),\ d(n)) = \max_{d(n)} F_n(u^o(n),\ d(n)), \tag{12}$$

gdzie :

$$F_n(u(n), d(n)) = Q_{n+1}(A\, x(n) - B\, u(n) + d(n)) + + L_n(u(n),\ A\, x(n) - B\, u(n) + d(n)) \tag{13}$$

zaś $d(n)$ przyjmuje wartości dyskretne $d^j(n)$ $j = 1,2, \dots, K$, przy czym spełnione jest ograniczenie (3). W dalszym ciągu przy rozważaniach dotyczących statycznego zadania minimaksu opuszczać będziemy n i przyjmować, że organizacja siatek/wektor zasobów/jest rozstrzygnięta, a $x(n)$ ustalone.

3. Algorytm minimaksu dyskretnego

Rozwiązanie analityczne problemu (11) możliwe jest jedynie dla bardzo wąskiej klasy zagadnień [2]. Proponowany algorytm opiera się na metodzie kolejnych przybliżeń punktów ε-stacjonarnych [3], przy czym zgodnie z sugestią zawartą w [4] do wyznaczania kierunku ε-najszybszego spadku korzysta się ze standardowego algorytmu programowania kwadratowego.

Oznaczmy przez $G(u)$ funkcję maksimum, tzn.

$$G(u) = \max_j F(u, d^j) \tag{14}$$

u_ε nazywa się punktem ε-stacjonarnym funkcji maksimum [3] jeśli :

$$D(u_\varepsilon) = \min_{\|g\| = 1}\ \max_{j \in R_\varepsilon(u_\varepsilon)} \left\langle \frac{\partial F}{\partial u}(u_\varepsilon, d^j), g \right\rangle \geqslant 0, \tag{15}$$

gdzie:

$$R_\varepsilon(u) = \{ j : G(u) - F(u, d^j) \leqslant \varepsilon \}. \tag{16}$$

Kierunkiem ε-minimaksowego gradientu funkcji $G(u)$ w punkcie u nazywamy wektor jednostkowy $g_\varepsilon(u)$ taki, że :

$$\max_{j \in R_\varepsilon(u)} \left\langle \frac{\partial F}{\partial u}(u, d^j), g_\varepsilon(u) \right\rangle = D(u) \tag{17}$$

Oznaczmy przez $H_\varepsilon(u)$ zbiór wszystkich pochodnych występujących w (16), tzn.

$$H_\varepsilon(u) = \left\{ z = \frac{\partial F}{\partial u}(u, d^j),\ j \in R_\varepsilon(u) \right\} \tag{18}$$

a przez $L_\varepsilon(u)$ powłokę wypukłą tego zbioru, tzn.

$$L_\varepsilon(u) = \left\{ z = \sum_{j=1}^{p} a_j z_j : z_j \in H_\varepsilon(u), a_j \geq 0, \sum_{j=1}^{p} a_j = 1 \right\} \quad (19)$$

Wykorzystywany algorytm opiera się na następujących dwóch twierdzeniach [3]:

Tw.1. u_ε jest ε-stacjonarnym punktem funkcji $G(u)$ wtedy i tylko wtedy, gdy

$$0 \in L_\varepsilon(u_\varepsilon) \quad (20)$$

Tw.2. Jeżeli u nie jest ε-stacjonarnym punktem $G(u)$ to funkcja $G(u)$ posiada w punkcie u jedyny kierunek ε-minimaksowego gradientu $g_\varepsilon(u)$ dany jako:

$$g_\varepsilon(u) = -\frac{z}{\|z\|}, \quad (21)$$

gdzie: z jest punktem zbioru $L_\varepsilon(u)$ najbliższym początku układu współrzędnych.

Proponowany algorytm składa się z następujących etapów:

1° Przyjęcie warunków startowych, określenie ciągu $\{\varepsilon_j\}$ dodatniego zbieżnego do zera, $i = 0$, $j = 0$, $u^i = u^0$, $\varepsilon_j = \varepsilon_0$, określenie δ.

2° Wyznaczenie zbiorów $R_{\varepsilon_j}(u^i)$ oraz $H_{\varepsilon_j}(u^i)$ na podstawie definicji (16) i (18). Załóżmy, że są one p elementowe.

3° Sprawdzenie warunku $D(u^i) \geq 0$. Dokonuje się tego w oparciu o twierdzenie 1, sprowadzając problem do zadania programowania liniowego. Mianowicie określmy $p+1$ wymiarowy wektor $[a_1, a_2, \ldots, a_p, t]$ taki, że minimalizowane jest względem niego t przy ograniczeniach

$$\sum_{s=1}^{p} a_s z_s^r - t \leq 0 \qquad r = 1,2,\ldots,m$$

$$-\sum_{s=1}^{p} a_s z_s^r - t \leq 0 \qquad r = 1,2,\ldots,m$$

$$\sum_{s=1}^{p} a_s = 1, \quad a_s \geq 0, \quad t \geq 0,$$

gdzie: $z_s^r = \frac{\partial F}{\partial u_r}(d^s, u^i)$, $\quad s \in R_{\varepsilon_j}(u^i)$

Problem ten może być rozwiązany za pomocą algorytmu Simplex.

Jeżeli otrzymana stąd wartość t jest równa 0, to wówczas

$$0 \in L_{\varepsilon_j}(u^i)$$

i punkt $u^i_{\varepsilon_j} = u^i$ jest ε_j-stacjonarnym punktem funkcji $G(u)$. Jeśli tak nie jest przechodzimy do następnego kroku, czyli wyznaczania kierunku poszukiwań.

4° Wyznaczenie kierunku poszukiwań $g(u^i)$ wymaga znalezienia punktu $z \in L_{\varepsilon_j}$ spełniającego twierdzenie 2. Zadanie to można rozwiązać wykorzystując algorytm programowania kwadratowego. Zachodzi bowiem [4]

$$\min_{z \in L_{\varepsilon_j}(u^i)} z^T z = \min_a a^T Z^T Z a \tag{22}$$

przy ograniczeniach

$$\sum_{s=1}^{p} a_s = 1\,, \quad a_s \geqslant 0, \quad s = 1, \ldots, p \tag{23}$$

gdzie: Z jest macierzą utworzoną z wektorów z_s, $s = 1, \ldots, p$. Poszukiwany punkt z wyznaczony zostanie jako

$$z = \sum_{s=1}^{p} a_s z_s,$$

gdzie: a_s, $s = 1, \ldots, p$ są rozwiązaniami zadania programowania kwadratowego (22), (23). Proponujemy tu zastosowanie algorytmu Wolfe'a [5].

5° Mając wyznaczony kierunek $g(u^i)$ należy znaleźć h minimalizujące $G(u^i + h\,g(u^i))$. Jest to problem minimalizacji w kierunku, który można rozwiązać, np. metodą złotego podziału.
Wartość u w kolejnej iteracji otrzymuje się jako $u^{i+1} = u^i + h\,g(u^i)$ i przechodzimy do punktu 2°.
Jeśli w punkcie 3° okaże się, że u^i jest punktem ε_j-stacjonarnym wówczas sprawdzamy kryterium stopu:

$$\varepsilon_j < \delta$$

i gdy nie jest spełnione przechodzimy do punktu 2° z nowym ε_j.

Pewnym mankamentem proponowanego algorytmu jest konieczność różniczkowania funkcji celu, podczas gdy ze względu na siatki programowania dynamicznego funkcja ta jest zadana tabelarycznie. Trudność tę można pokonać przez aproksymację międzywęzłową i różniczkowanie numeryczne.

4. Przypadek dopływu zasobów ciągłych

Proponowany algorytm można rozszerzyć na przypadek nieznanych, lecz ograniczonych dopływów zasobów ciągłych. Można tego dokonać przez wprowadzenie siatek skończonych na zmienną d [3] bądź arbitralne przyjęcie pewnych wartości $d = d^j$ tworzących zbiór skończony. W [2] zaproponowano wykorzystanie tu algorytmu Salomona [6]. W algorytmie tym wykorzystuje się jedną z metod rozwiązania zadania programowania nieliniowego z ograniczeniami równościowymi do wyznaczania nowego elementu zbioru wartości d^j. Poszukuje się mianowicie w każdym kroku iteracyjnym elementu d^k maksymalizującego funkcję $F(u_\varepsilon^k, d)$, przy czym u_ε^k jest wynikiem uzyskanym z zastosowania algorytmu minimaksu dyskretnego przy poprzednio określonym zbiorze dyskretnych dopływów. W każdej pętli iteracyjnej wykorzystywany jest zatem algorytm z rozdziału 3, przy czym zbiór wartości d^j ulega zwiększeniu. Kryterium stopu jest brak poprawy w wartościach funkcji maksimum. Zauważmy, że dla rozwiązania całego problemu dynamicznego konieczne jest połączenie algorytmu minimaksu z metodą siatek programowania dynamicznego.

W pracy [2] przebadano algorytm na szeregu przykładach. W każdym punkcie siatki uzyskano dobrą zbieżność osiągając warunek stopu po trzech, czterech iteracjach algorytmu minimaksu statycznego ciągłego bądź dyskretnego. Mimo to, ze względu na konieczność stosowania metody siatek już przy modelu dwuwymiarowym i czterokrokowym horyzoncie sterowania czas obliczeń był znaczny.

Analityczne wyznaczenie strategii optymalnej jest w zasadzie możliwe jedynie w przypadku skalarnym. Poniżej przedstawimy taki przykład; ale nawet w tym przypadku uzyskanie rozwiązania analitycznego nie jest sprawą prostą.

Rozpatrzmy następujący problem rozdziału zasobów:
Dysponujemy zasobem początkowym w ilości $x_o = 10$ jednostek. Zasób ten można skierować w całości lub części na jeden z dwóch celów. Skierowanie y jednostek zasobu na I cel daje w ciągu okresu zysk $f_1 = c_1 - \frac{1}{2} y^2$, zaś skierowanie v jednostek na II cel w ciągu jednego okresu przynosi zysk $f_2 = c_2 - \frac{1}{2} v^2$, przy czym zasoby przeznaczone na I cel ulegają 20 procentowemu zużyciu. Maksymalny dopływ zasobu w ciągu jednego okresu wynosi $w = 1$ jednostka.

W celu określenia strategii optymalnej w ciągu 3 okresów $n = 0,1,2$ określmy równanie ewolucji zasobów:

$$x(n+1) = x(n) - 0.2\, y(n) + d(n) \qquad x(0) = x_o$$

$$d(n) \leq 1 \qquad\qquad 0 \leq y(n) \leq x(n)$$

oraz zysk całkowity:

$$V = \sum_{n=0}^{2} \left\{ \left[c_1 - \frac{1}{2}\ y^2(n)\right] + \left[c_2 - \frac{1}{2}\ v^2(n)\right] \right\} =$$

$$\sum_{n=0}^{2} \left\{ c_1 + c_2 - \frac{1}{2} y^2(n) - \frac{1}{2}\left(x(n) - y(n)\right)^2 \right\}$$

Problem maksymalizacji zysku zastąpmy ze względu na stałość c_1, c_2 problemem minimalizacji wskaźnika :

$$I = \frac{1}{2} \sum_{n=0}^{2} \left\{ y^2(n) + \left(x(n) - y(n)\right)^2 \right\} = \frac{1}{4} \sum_{n=0}^{2} \left\{ u^2(n) + x^2(n) \right\},$$

gdzie:

$$u(n) = 2\,y(n) - x(n)$$

Równanie ewolucji zasobów zmieni się do postaci :

$$x(n+1) = 0.9\,x(n) - 0.1\,u(n) + d(n)$$

Ze względu na niepełną informację o $d(n)$ zadanie optymalizacji rozwiążemy jako zadanie minimaksu, tzn. poszukiwać będziemy $u(n)$ minimalizującego

$$\tilde{I} = \max_{d(n) \leq 1} \sum_{n=0}^{2} \left\{ u^2(n) + x^2(n) \right\}$$

Ponieważ $x(0)$ jest dane, a jak łatwo zauważyć optymalne $u(2) = 0$ wskaźnik może być zapisany jako

$$\max_{d(n) \leq 1} \sum_{n=0}^{1} \left\{ u^2(n) + x^2(n+1) \right\}$$

przy ograniczeniach $\quad 0 \leq u(n) \leq x(n)$

Po wykorzystaniu warunków koniecznych i wystarczających minimaksu /ze względu na wypukłość funkcji maksimum/ mamy dla przyjętych danych

$$u(1) = \frac{9\,x(1) + 10}{101} < x(1)$$

oraz

$$u(0) = \frac{15{,}4\,x(0) + 27.2}{102} \cong 1.9 < x(0)$$

Już w przypadku dwuwymiarowym i kwadratowym wskaźnika jakości uzyskanie rozwiązania analitycznego jest niemożliwe. Należy wówczas zastosować algorytm numeryczny.

Przykładowo dla systemu, w którym zasoby ewoluują zgodnie z równaniami

$$x(n+1) = 0{,}9\,x(n) - u(n) + d_1(n) \qquad x(0) = 13.16$$

$$y(n+1) = 0.9\,y(n) - x(n) + d_2(n) \qquad y(0) = 21.41$$

przy ograniczeniach

$$0 \leqslant u(n) \leqslant x(n) \leqslant y(n)$$

$$d_1(n) + d_2(n) \leqslant 1$$

uzyskano strategię minimaksową dla wskaźnika jakości

$$I = \sum_{n=0}^{2} \left\{ x^2(n+1) + y^2(n+1) \right\}$$

w postaci decyzji :

$$u(0) = 7.67\ , \quad u(1) = 2.90\ , \quad u(2) = 1.56$$

i odpowiadających im wielkościach zasobów :

$$x(0) = 13.16\ , \quad x(1) = 4.87\ , \quad x(2) = 2.18\ , \quad x(3) = 1.11$$

$$y(0) = 21.41\ , \quad y(1) = 7.52\ , \quad y(2) = 3.31\ , \quad y(3) = 2.22$$

5. Uwagi końcowe

W pracy przedstawiono algorytm rozdziału zasobów w systemie, w którym dopływy zasobów są zmiennymi niepewnymi. W przeciwieństwie jednak do wielu prac, np. : [7], [8], [9], w których traktuje się te zmienne jako losowe, przyjęto odmienny model niepewności zakładający przynależność tych zmiennych do określonych zbiorów ograniczonych. Podstawowa wersja algorytmu dotyczy dopływu zasobów dyskretnych, przedstawiono jednak również wersję dla przypadku zasobów ciągłych. Kryterium jakości przyjmowane jest w postaci gwarantowanego wskaźnika, co wydaje się być naturalną konsekwencją postaci informacji o zmiennych niepewnych. Ta zaś wydaje się być informacją najczęściej dostępną w praktyce. Ogólność algorytmu umożliwia jego zastosowanie również w przypadku występowania innych parametrów niepewnych w modelu systemu i wskaźniku /np.: zapotrzebowań na zasoby/ jeśli one mają również charakter zbiorowy. W pracy nie rozpatrywano drugiego alternatywnego podejścia polegającego na aproksymacji zbiorów zmiennych niepewnych elipsoidami czy hiperwielościanami, co w niektórych prostych przypadkach prowadzi do analitycznych rezultatów, np. : [10] .

LITERATURA

[1] Świerniak A. : State-inequalities approach to control systems with uncertainty, IEE Proceedings, v.129 , pt.D , No 6, November 1982
[2] Słomp M.: Praca dyplomowa magisterska /nieopublikowana /, Gliwice,1983
[3] Diemianow W.F., Małoziemow W. : Wwiedienije w min-max, Izdat.Nauka, Moskwa, 1972
[4] Heller J.E., Cruz J.B. : An algorithm for min-max parameter optimization, Automatica, v.8 , No 3 , May 1972
[5] Findeisen W., Szymanowski J., Wierzbicki A. : Teoria i metody obliczeniowe optymalizacji, PWN, Warszawa , 1980
[6] Salomon D.M. : Minimax controller design, IEEE Trans.on Automatic Control, AC-13, No 4, August, 1968
[7] Aoki M., Toda M. : Parameter adaptive resource allocation problem for decentralized system, IEEE Trans. on Automatic Control, AC-20, No 2, April, 1975
[8] Gessing R. : Metoda dekompozycji i koordynacji statystycznie optymalnego statycznego rozdziału zasobów, Zeszyty Naukowe Politechniki Śląskiej, Automatyka, z.54, Gliwice, 1980
[9] Świerniak A. : Stochastycznie optymalne sterowanie rozdziałem zasobów w zdecentralizowanym systemie dynamicznym, Zeszyty Naukowe Politechniki Śląskiej, Automatyka, z.63, Gliwice, 1982
[10] Bertsekas D.P., Rhodes I.B. : Sufficiently informative functions and the minimax feedback control of uncertain dynamic systems, IEEE Trans. on Automatic Control , AC-18, No 2, April, 1973

Recenzent: Doc.dr hab.inż.Józef Grabowski

Wpłynęło do Redakcji do 30.03.1984r.

АЛГОРИТМ МИНИМАКСНОГО РАСПРЕДЕЛЕНИЯ РЕСУРСОВ В СИСТЕМЕ

Р е з ю м е

В работе представлен алгоритм определения оптимального распределения ресурсов в случае неполной информации о дискретном ресурсе на входе системы. Принимая, что известны ограничения на входной дискретный ресурс предлагается минимаксный оптимизационный критерий , а также алгоритм основан на стандартных процедурах математического программирования.

AN ALGORITHM FOR MIN - MAX RESOURCE ALLOCATION IN A SYSTEM

S u m m a r y

An algorithm for optimal resource allocation in the presence of uncertainty about discrete inflows to a system has been proposed. Assumption about given bounds on the unknown inflows implies min-max performance criterion. The algorithm is based on the standard mathematical programming procedures.