Tadeusz Trzaskalik*

Strategie optymalne i prawie optymalne w dyskretnym stochastycznym programowaniu dynamicznym

Wstęp

W niniejszym artykule zajmujemy się wieloetapowymi, dyskretnymi procesami decyzyjnymi. Rozpatrujemy procesy podzielone na skończoną liczbę etapów. Decyzja nie jest podejmowana jednorazowo, lecz wielokrotnie, na początku każdego etapu. W zależności od dokonanych wyborów skutki wcześniejszych decyzji ograniczają lub – przeciwnie – rozszerzają możliwości decyzyjne w następnych etapach.

Dynamikę rozpatrywanych procesów można przedstawić następująco [Trzaskalik, 1986, 1990, 1998]. Na początku każdego z etapów proces może się znajdować w pewnym zbiorze stanów dopuszczalnych. Decydent podejmuje decyzję dopuszczalną, co skutkuje przejściem procesu do stanu początkowego następnego etapu. W procesach deterministycznych przejście to określone jest poprzez funkcję przejścia. Ciąg podejmowanych decyzji oceniany jest całościowo za pomocą wieloetapowej funkcji kryterium, będącej sumą kryteriów etapowych, charakteryzujących wybory dokonane w kolejnych etapach.

Kluczową rolę w zarządzaniu wieloetapowym procesem decyzyjnym odgrywa pojęcie strategii. Jest to funkcja, która każdemu dopuszczalnemu procesu przyporządkowuje decyzję dopuszczalną. Optymalne sterowanie wieloetapowym, dyskretnym procesem decyzyjnym wymaga znalezienia strategii optymalnej, dla której wieloetapowa funkcja kryterium przyjmuje – w zależności od charakteru rozpatrywanego problemu – wartość maksymalną (tak jak w rozpatrywanych w dalszej części niniejszej pracy procesach) lub minimalną. Do jej znalezienia stosujemy metodę programowania dynamicznego, wykorzystującą równania optymalności Bellmana [Bellman, 1957; Bellman, Dreyfus, 1967].

Niejednokrotnie jednoznaczne opisanie dynamiki procesu oraz oszacowanie skutków podejmowanych decyzji nie jest jednak możliwe. W wielu przypadkach staramy się uzyskać częściową wiedzę pozwalającą na oszacowanie rozkładów prawdopodobieństwa, opisujących

* Prof. dr hab., Katedra Badań Operacyjnych, Wydział Informatyki i Komunikacji, Uniwersytet Ekonomiczny w Katowicach, ul. 1 Maja 50, 40-287 Katowice, tadeusz.trzaskalik@ue.katowice.pl

zachowanie się procesu w zależności od podejmowanych decyzji, oraz wartości kryteriów etapowych związanych z tymi zmianami. Podejście to nazwane jest podejściem stochastycznym. Oceniając rozpatrywane strategie, wykorzystujemy tu pojęcie wartości oczekiwanej strategii, związane ze znanym decydentowi rozkładem prawdopodobieństwa w zbiorze stanów początkowych. Do określenia strategii optymalnej ponownie wykorzystujemy zasadę optymalności Bellmana, tym razem w wersji stochastycznej [Trzaskalik, 1986; Trzaskalik, Do Thien Hoa, 1999; Trzaskalik, Sitarz, 2007; Trzaskalik, Nowak, 2012].

Dla decydenta interesująca może być jednak nie tylko informacja o tym, jaką postać ma strategia optymalna, ale również znalezienie kolejnych strategii, dla których wartość oczekiwana jest bliska wartości optymalnej, z dokładnością określoną przez decydenta. Pojawia się pytanie o to, czy i w jaki sposób strategie te mogą być znajdowane.

Zagadnienie to w przypadku zadań deterministycznych podejmowane było we wcześniejszych pracach. Proponowane rozwiązania dotyczyły jednak poszukiwania odpowiednich realizacji procesu, które w przypadku zagadnień deterministycznych mogły być wyznaczone ze względu na charakter tych procesów [Elmaghraby, 1970; Trzaskalik 1990]. W przypadku zadań stochastycznych takie podejście nie jest możliwe.

Proponowana w niniejszej pracy metoda polega na znalezieniu zbioru strategii optymalnych, a następnie rozszerzaniu tego zbioru o kolejne strategie prawie optymalne, które mieszczą się w obszarze zainteresowań decydenta. Nowe strategie generujemy poprzez zmianę decyzji w jednym stanie dla strategii zaakceptowanej wcześniej. Dla każdej nowo określonej strategii sprawdzamy, czy nie była już wcześniej analizowana, a jeżeli nie, to wyliczamy jej wartość oczekiwaną i sprawdzamy, czy mieści się ona w obszarze zainteresowań decydenta. Proponowany dalej algorytm porządkuje to postępowanie. Jego szczegóły przedstawiono w dalszej części opracowania.

Zaproponowany sposób postępowania wzorowany jest na metodzie poszukiwania prawie optymalnych strategii w analizie drzewa decyzyjnego, zaproponowanej w pracy [Nowak, 2014]. Wcześniej podobne podejście, polegające na analizowaniu baz sąsiednich w stosunku do bazy aktualnie rozpatrywanej, zastosował R. Steuer w wielokryterialnej metodzie programowania liniowego ADBASE, która pozwala na znalezienie wszystkich baz sprawnych [Steuer, 2003].

Celem niniejszego artykułu jest prezentacja metody poszukiwania rozwiązań prawie optymalnych, polegającej na generowaniu kolejnych strategii przez zmianę decyzji w jednym stanie. Pozwala ona na ograniczenie przeglądu zbioru wszystkich strategii.

Artykuł składa się z 4 części. W części 1 przedstawione zostały: wykorzystywana dalej notacja, metoda obliczenia wartości oczekiwanej dla dowolnej strategii oraz równania optymalności, pozwalające na znalezienie w oparciu o zasadę optymalności Bellmana strategii optymalnej. W części 2 zdefiniowano pojęcie strategii prawie optymalnej i przedstawiono proponowany algorytm, pozwalający na wyznaczenie zbioru strategii prawie optymalnych z zadaną przez decydenta dokładnością. Część 3 ma charakter ilustracyjny. Przedstawiono w niej szczegółowo przykład liczbowy, ilustrujący działanie algorytmu. W części 4 znajduje się dyskusja i wnioski.

Praca wykonana w ramach projektu badawczego NCN DEC-2013/11/B/HS4/01471.

1. Strategia optymalna

Wykorzystamy następujące oznaczenia [Trzaskalik, 1986]:

T – liczba etapów rozpatrywanego procesu decyzyjnego,

y_t – stan procesu na początku etapu t (t=1,…,T),

$\mathbf{Y}_t$ – skończony zbiór stanów procesu na początku etapu t,

$\mathbf{Y}_{T+1}$ – skończony zbiór stanów końcowych procesu,

x_t – stan dopuszczalny na początku etapu t,

$\mathbf{X}_t(y_t)$ – skończony zbiór stanów dopuszczalnych na początku etapu t, jeżeli proces na początku tego etapu znajdował się w stanie $y_t \in \mathbf{Y}_t$,

$F_t(y_{t+1} | y_t, x_t)$ – wartość kryterium etapowego dla etapu t przy przejściu od stanu y_t do stanu y_{t+1}, gdy podjęta była decyzja $x_t \in \mathbf{X}_t(y_t)$,

$P_t(y_{t+1} | y_t, x_t)$ – prawdopodobieństwo przejścia procesu w etapie t ze stanu y_t do stanu y_{t+1}, gdy podjęta została decyzja $x_t \in \mathbf{X}_t(y_t)$.

$P(y_1)$ – rozkład prawdopodobieństwa dla stanów początkowych $y_1 \in \mathbf{Y}_1$.

Zachodzi związek:

$$\forall_{t\in\overline{1,T}} \forall_{y_t\in\mathbf{Y}_t} \forall_{x_t\in\mathbf{X}_t(y_y)} \sum_{y_{t+1}\in\mathbf{Y}_{t+1}} P_t(y_{t+1} \mid y_t, x_t) = 1 \tag{1}$$

gdzie:

$\{x\}$ – strategia, czyli funkcja, która każdemu stanowi $y_t \in \mathbf{Y}_t$ przyporządkowuje jednoznacznie decyzję $x_t \in \mathbf{X}_t(y_t)$,

$\{\mathbf{X}\}$ – zbiór wszystkich strategii rozpatrywanego procesu,

$\{x_{\overline{t,T}}\}$ – strategia skrócona, obejmująca etapy od t do T,.

Załóżmy, że wybraliśmy pewną strategię $\{\bar{x}\} \in \{\mathbf{X}\}$. Oczekiwaną wartość realizacji procesu dla strategii skróconej $\{\bar{x}_{\overline{T,T}}\}$ obliczamy następująco:

$$G_T(y_T, \{\bar{x}_{\overline{T,T}}\}) = \sum_{y_{T+1} \in \mathbf{Y}_{T+1}} F_T(y_{T+1} \mid y_T, \bar{x}_T) P_T(y_{T+1} \mid y_T, \bar{x}_T) \tag{2}$$

Oczekiwaną wartość realizacji procesu dla strategii skróconej $\{\bar{x}_{\overline{t,T}}\}$, gdy proces na początku etapu znajdował się w stanie $y_t \in \mathbf{Y}_t$ obliczamy ze wzoru:

$$G_t(y_t, \{\bar{x}_{\overline{t,T}}\}) = \sum_{y_{t+1} \in \mathbf{Y}_{t+1}} (F_t(y_{t+1} \mid y_t, \bar{x}_t) + G_{t+1}(y_{t+1}, \{\bar{x}_{\overline{t+1,T}}\})) P_t(y_{t+1} \mid y_t, \bar{x}_t) \tag{3}$$

Oczekiwaną wartość realizacji procesu dla ustalonej strategii $\{\bar{x}\}$ obliczamy ze wzoru:

$$G\{\bar{x}\} = \sum_{y_1 \in \mathbf{Y}_1} G_1(y_1, \{\bar{x}\}) P_1(y_1) \tag{4}$$

Wykorzystując zasadę optymalności Bellmana [Bellman, 1997], określamy strategię optymalną.

Algorytm 1

1. Dla każdego stanu $y_T \in \mathbf{Y}_T$ obliczamy wartości optymalne

$$G_T^*(y_T) = \max_{x_T \in \mathbf{X}_t(y_t)} \sum_{y_{T+1} \in \mathbf{Y}_{T+1}} F_T(y_{T+1} \mid y_T, x_T) P_T(y_{T+1} \mid y_T, x_T) \tag{5}$$

   i znajdujemy decyzję $x_t^*(y_t)$, dla której to maksimum jest osiągnięte. Decyzja ta jest częścią konstruowanej przez nas strategii optymalnej.

2. Dla etapu t, $t \in \overline{T-1,1}$ i każdego stanu $y_t \in \mathbf{Y}_t$ obliczamy wartości optymalne

$$G_t^*(y) = \max_{x_t \in \mathbf{X}_t(y_t)} \sum_{y_{t+1} \in \mathbf{Y}_{t+1}} (F_t(y_{t+1} \mid y_t, x_t) + G_{t+1}^*(y_{t+1})) P_t(y_{t+1} \mid y_t, x_t) \tag{6}$$

   i znajdujemy decyzję $x_t^*(y_t)$, dla której to maksimum jest osiągnięte. Decyzja ta jest częścią konstruowanej przez nas strategii optymalnej.

3. Optymalną oczekiwaną wartość realizacji procesu obliczamy ze wzoru:

$$G\{x^*\} = \sum_{y_1 \in \mathbf{Y}_1} G_1^*(y_1, \{x^*\}) P_1(y_1) \tag{7}$$

2. Wyznaczanie strategii prawie optymalnych

Strategię $\{x^p\}$ nazywamy strategią prawie optymalną, jeżeli jej oczekiwana wartość różni się od oczekiwanej wartości strategii optymalnej $\{x^*\}$ co najwyżej o zadaną wartość ε, czyli

$$G\{x^*\} - G\{x^p\} \leq \varepsilon \tag{8}$$

Oznacza to, że decydenta interesują strategie, dla których

$$\mathrm{G}\{\mathrm{x}^{\mathrm{p}}\} \geq \mathrm{Z} \tag{9}$$

przy czym

$$Z = G\{x^*\} - \varepsilon \tag{10}$$

Przypuścimy, że decydent zainteresowany jest znalezieniem strategii prawie optymalnych i określił wartość ε. Przyjmujemy następujące oznaczenia:
LS – lista strategii optymalnych i prawie optymalnych,
LSB – lista strategii do przebadania, czyli takich, które mogą być modyfikowane w celu wyznaczenia kolejnych strategii prawie optymalnych.
LSC – lista strategii rozpatrywanych w trakcie działania algorytmu.

Algorytm 2

1. Przyjmij**: LS**:= ∅, **LSB**:= ∅, **LSC** = ∅.
2. Wykorzystując **Algorytm 1**, wyznacz zbiór strategii {**X***}, dla których rozpatrywane kryterium osiąga wartość optymalną.
3. Zapisz strategie ze zbioru {**X***} do zbiorów **LS**, **LSB i LSC**:
   LS:= **LSC** ∪ {**X***}.
   LSB := **LBS** ∪ {**X***}.
   LSC:= **LSC** ∪ {**X***}.
4. Jeżeli **LSB** = ∅, przejdź do kroku 11.
5. Wybierz kolejną strategię {x} ze zbioru **LSB** i usuń ją z tego zbioru:
   LSB:= **LSB** \ {x}.
6. Wyznacz wszystkie strategie zmodyfikowane, które różnią się od strategii {x} decyzją podejmowaną w jednym stanie i zapisz je w zbiorze **M**{x}.
7. Sprawdź, czy w zbiorze **M**{x} znajdują się strategie, które są również w zbiorach **LS**, **LSB** oraz **LSC**. Usuń powtarzające się strategie ze zbioru **M**{x}.
   M{x} = **M**{x} \ (**M**{x}∩**LS**) \ (**M**{x}∩**LSB**) \ (**M**{x}∩**LSC**)
8. Sprawdź, czy **M**{x} ≠ ∅. Jeżeli nie, przejdź do kroku 4.
9. Dla kolejnych strategii $\{x^m\} \in$ **M**{x}:

a) wykorzystując wzory (2) i (3), oblicz wartość oczekiwaną rozpatrywanej strategii $\{x^m\}$,
b) zapisz strategię $\{x^m\}$ w zbiorze **LSC**:
LSC:= **LSC** $\cup$ $\{x^m\}$,
c) jeżeli wartość oczekiwana rozpatrywanej strategii jest nie niższa niż Z to zapisz strategię $\{x^m\}$ do zbiorów **LS** oraz **LSB**:
LS:= **LS** $\cup$ $\{x^n\}$,
LSB:= **LSB** $\cup$ $\{x^n\}$.

10. Przejdź do kroku 4.
11. Koniec procedury.

3. Ilustracja działania Algorytmu 2

Rozpatrujemy trzyetapowy proces decyzyjny. Zbiory stanów na początku kolejnych etapów są następujące:

$$\mathbf{Y}_1 = \{1,2\} \; \mathbf{Y}_2 = \{3,4\} \qquad \mathbf{Y}_3 = \{5,6\}$$

Zbiór stanów końcowych procesu ma postać:

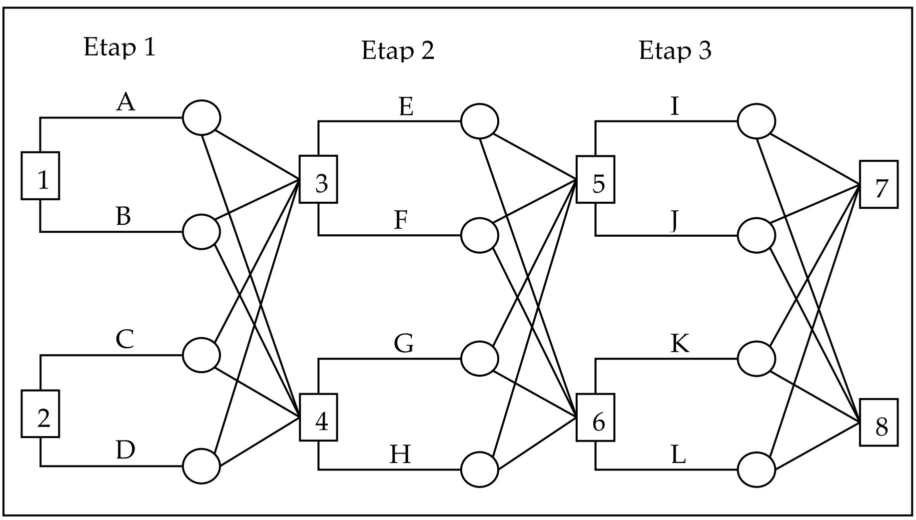
$$\mathbf{Y}_4 = \{7,8\}$$

Zbiory decyzji dopuszczalnych są następujące:

$$\mathbf{X}_1(1) = \{A,B\} \quad \mathbf{X}_1(2) = \{C,D\}$$
$$\mathbf{X}_2(3) = \{E,F\} \quad \mathbf{X}_2(4) = \{G,H\}$$
$$\mathbf{X}_3(5) = \{I,J\} \quad \mathbf{X}_3\{6\} = \{K,L\}$$

Graf procesu przedstawiony jest na rysunku 1.

Rysunek 1. Graf procesu



Źródło: Opracowanie własne.

Przykładowo, jeżeli proces znajduje się w stanie 1 i podjęta została decyzja A, wtedy prawdopodobieństwo przejścia procesu do stanu 3 wynosi P(3|1,A), natomiast prawdopodobieństwo przejścia do stanu 4 jest równe P(4|1,A). Odpowiadające tym sytuacjom wartości funkcji kryterium wynoszą odpowiednio F(3|1,A) oraz F(4|1,A). Wszystkie wartości prawdopodobieństw przejść oraz etapowych funkcji korzyści dane są w tablicy 1.

Tablica 1. Wartości prawdopodobieństw przejść i funkcji korzyści

Etap	$(y_{t+1} \mid y_t, x_t)$	P(·)	F(·)	Etap	$(y_{t+1} \mid y_t, x_t)$	P(·)	F(·)
1	(3 | 1,A)	0,4	15	2	(5 | 4,G)	0,5	15
1	(4 | 1,A)	0,6	17	2	(6 | 4,G)	0,5	18
1	(3 | 1,B)	0,7	18	2	(5 | 4,H)	0,3	13
1	(4 | 1,B)	0,3	19	2	(6 | 4,H)	0,7	22
1	(3 | 2,C)	0,4	15	3	(7 | 5,I)	0,2	30
1	(4 | 2,C)	0,6	17	3	(8 | 5,I)	0,8	12
1	(3 | 2,D)	0,7	18	3	(7 | 5,J)	0,9	22
1	(4 | 2,D)	0,3	19	3	(8 | 5,J)	0,1	28
2	(5 | 3,E)	0,5	15	3	(7 | 6,K)	0,2	30
2	(6 | 3,E)	0,5	18	3	(8 | 6,K)	0,8	12
2	(5 | 3,F)	0,3	13	3	(7 | 6,L)	0,9	22
2	(6 | 3,F)	0,7	22	3	(8 | 6,L)	0,1	28

Źródło: Opracowanie własne.

Ze względu na niewielkie rozmiary tego ilustracyjnego zadania istniejące strategie możemy dla lepszej przejrzystości wypisać i ponumerować od 1 do 64. Numerację tę przedstawia tablica 2.

Tablica 2. Lista strategii

Nr	Decyzje	Nr	Decyzje	Nr	Decyzje	Nr	Decyzje
1	(A,C,E,G.I,K)	17	(A,D,E,G.I,K)	33	(B,C,E,G.I,K)	49	(B,D,E,G.I,K)
2	(A,C,E,G.I,L)	18	(A,D,E,G.I,L)	34	(B,C,E,G.I,K)	50	(B,D,E,G.I,L)
3	(A,C,E,G.J,K)	19	(A,D,E,G.J,K)	35	(B,C,E,G.J,K)	51	(B,D,E,G.J,K)
4	(A,C,E,G.J,L)	20	(A,D,E,G.J,L)	36	(B,C,E,G.J,L)	52	(B,D,E,G.J,L)
5	(A,C,E,H.I,K)	21	(A,D,E,H.I,K)	37	(B,C,E,H.I,K)	53	(B,D,E,H.I,K)
6	(A,C,E,H.I,L)	22	(A,D,E,H.I,L)	38	(B,C,E,H.I,L)	54	(B,D,E,H.I,L)
7	(A,C,E,H.J,K)	23	(A,D,E,H.J,K)	39	(B,C,E,H.J,K)	55	(B,D,E,H.J,K)

Nr	Decyzje	Nr	Decyzje	Nr	Decyzje	Nr	Decyzje
8	(A,C,E,H.J,L)	24	(A,D,E,H.J,L)	40	(B,C,E,H.J,L)	56	(B,D,E,H.J,L)
9	(A,C,F,G.I,K)	25	(A,D,F,G.I,K)	41	(B,C,F,G.I,K)	57	(B,D,F,G.I,K)
10	(A,C,F,G.I,L)	26	(A,D,F,G.I,L)	42	(B,C,F,G.I,L)	58	(B,D,F,G.I,L)
11	(A,C,F,G.J,K)	27	(A,D,F,G.J,K)	43	(B,C,F,G.J,K)	59	(B,D,F,G.J,K)
12	(A,C,F,G.J,L)	28	(A,D,F,G.J,L)	44	(B,C,F,G.J,L)	60	(B,D,F,G.J,L)
13	(A,C,F,H.I,K)	29	(A,D,F,H.I,K)	45	(B,C,F,H.I,K)	61	(B,D,F,H.I,K)
14	(A,C,F,H.I,L)	30	(A,D,F,H.I,L)	46	(B,C,F,H.I,L)	62	(B,D,F,H.I,L)
15	(A,C,F,H.J,K)	31	(A,D,F,H.J,K)	47	(B,C,F,H.J,K)	63	(B,D,F,H.J,K)
16	(A,C,F,H.J,L)	32	(A,D,F,H.J,L)	48	(B,C,F,H.J,L)	64	(B,D,F,H.J,L)

Źródło: Opracowanie własne.

Przypuśćmy, że decydent określił, że interesują go strategie optymalne oraz prawie optymalne, dla których oczekiwana wartość może być mniejsza od oczekiwanej wartości optymalnej nie więcej niż o 1,5%. Wykorzystując Algorytm 2, wyznaczamy strategie optymalne i prawie optymalne w następujący sposób:

1. Przyjmujemy:
   LS:= ∅, **LSB**:= ∅, **LSC** := ∅.
2. Wykorzystując **Algorytm 1**, znajdujemy zbiór strategii optymalnych:
   $\{\mathbf{X}^*\} = \{\{x^{64}\}\}$
   przy czym $\{x^{64}\}$ = (B, D, F, H, J, L) (patrz tablica 1). Mamy
   $G^*\{x^{64}\}$ = 60.2. Ponieważ wartości oczekiwane dla strategii prawie optymalnych nie mogą się odchylać od oczekiwanej wartości optymalnej więcej niż o 1,5%, mamy: ε= 0,903, Z = 59,297.
3. Dodajemy strategię optymalną do zbiorów **LS** i **LSB**. Mamy:
   LS := **LS** ∪ $\{\mathbf{X}^*\} = \{ \{x^{64}\} \}$
   LSB := **LSB** ∪ $\{\mathbf{X}^*\} = \{ \{x^{64}\} \}$.
4. Ponieważ **LSB** ≠ ∅ przechodzimy do kroku 5.
5. Wybieramy strategię $\{x^{64}\}$ ze zbioru **LSB**, usuwamy ją z tego zbioru i dodajemy do zbioru **LSC**:
   LSB:= **LSB** \ $\{x^{64}\}$ = ∅
   LSC = **LSC** ∪ $\{x^{64}\} = \{ \{x^{64}\} \}$.
6. Określamy wszystkie strategie zmodyfikowane, różniące się od strategii $\{x^{64}\}$ decyzją w jednym stanie. Otrzymujemy następujące strategie:
   $\{x^{63}\}$ = {B, D, F, H, J, K} $\{x^{56}\}$ = {B, D, E, H, J, L}

$\{x^{62}\} = \{B, D, F, H, I, L\}$ $\{x^{48}\} = \{B, C, F, H, J, L\}$

$\{x^{60}\} = \{B, D, F, G, J, L\}$ $\{x^{32}\} = \{A, D, F, H, J, L\}$

i umieszczamy je w zbiorze $\mathbf{M}\{x^{64}\}$:

$\mathbf{M}\{x^{64}\} := \{ \{x^{63}\}, \{x^{62}\}, \{x^{60}\}, \{x^{56}\}, \{x^{48}\}, \{x^{32}\} \}$.

7. Sprawdzamy, czy zbiór $\mathbf{M}\{x^{64}\}$ zawiera strategie, które znajdują się również w zbiorach **LS**, **LSB** i **LSC**. Otrzymujemy:
   $\mathbf{M}\{x^{64}\} \cap \mathbf{LS} = \varnothing$
   $\mathbf{M}\{x^{64}\} \cap \mathbf{LSB} = \varnothing$
   $\mathbf{M}\{x^{64}\} \cap \mathbf{LSC} = \varnothing$
   stąd
   $\mathbf{M}\{x^{64}\} \setminus (\mathbf{M}\{x^{64}\} \cap \mathbf{LS}) \setminus (\mathbf{M}\{x^{64}\} \cap \mathbf{LSB}) \setminus (\mathbf{M}\{x^{64}\} \cap \mathbf{LSC}) = \mathbf{M}\{x^{64}\}$
8. Mamy $\mathbf{M}\{x^{64}\} \neq \varnothing$.
9. Rozpatrujemy kolejne strategie $\{x^m\}$ ze zbioru $\mathbf{M}\{x^{64}\}$.
   Strategia $\{x^{63}\}$
   a) obliczamy $G\{x^{63}\} = 55{,}3$
   b) dodajemy strategię $\{x^{63}\}$ do zbioru **LSC**:
      $\mathbf{LSC} := \mathbf{LSC} \cup \{x^{63}\} = \{\{x^{64}\}, \{x^{63}\} \}$
   c) ponieważ $G\{x^{63}\} < 59{,}297$, strategii $\{x^{63}\}$ nie dodajemy ani do zbioru **LS**, ani do zbioru **LSB**.

   Strategia$\{x^{62}\}$
   a) obliczamy $G\{x^{62}\} = 58{,}1$
   b) dodajemy strategię $\{x^{62}\}$ do zbioru **LSC**:
      $\mathbf{LSC} := \mathbf{LSC} \cup \{x^{62}\} = \{ \{x^{64}\}, \{x^{63}\}, \{x^{62}\} \}$
   c) ponieważ $G\{x^{62}\} < 59{,}297$, strategii $\{x^{62}\}$ nie dodajemy ani do zbioru **LS**, ani do zbioru **LSB**.

   Strategia$\{x^{60}\}$
   a) obliczamy $G\{x^{60}\} = 59{,}36$
   b) dodajemy strategię $\{x^{60}\}$ do zbioru LSC:
      $\mathbf{LSC} := \mathbf{LSC} \cup \{x^{60}\} = \{ \{x^{64}\}, \{x^{63}\}, \{x^{62}\}, \{x^{60}\} \}$
   c) ponieważ $G\{x^{63}\} > 59{,}297$, strategię $\{x^{60}\}$ dodajemy zarówno do zbioru **LS**, jak i do zbioru **LSB**:
      $\mathbf{LS} := \mathbf{LS} \cup \{x^{60}\} = \{ \{x^{64}\}, \{x^{60}\} \}$
      $\mathbf{LSB} := \mathbf{LSB} \cup \{x^{60}\} = \{ \{x^{60}\} \}$

   Strategia$\{x^{56}\}$
   a) obliczamy $G\{x^{56}\} = 58{,}24$
   b) dodajemy strategię $\{x^{56}\}$ do zbioru **LSC**:
      $\mathbf{LSC} := \mathbf{LSC} \cup \{x^{56}\} = \{ \{x^{64}\}, \{x^{63}\}, \{x^{62}\}, \{x^{60}\}, \{x^{56}\} \}$

c) ponieważ G$\{x^{56}\}$ < 59,297, strategii $\{x^{56}\}$ nie dodajemy ani do zbioru **LS**, ani do zbioru **LSB**.

Strategia$\{x^{48}\}$

a) obliczamy G$\{x^{48}\}$ = 58,94

b) dodajemy strategię $\{x^{48}\}$ do zbioru **LSC**:
LSC := **LSC** ∪ $\{x^{48}\}$ = { $\{x^{64}\}$, $\{x^{63}\}$, $\{x^{62}\}$, $\{x^{60}\}$, $\{x^{56}\}$ $\{x^{48}\}$ }

c) ponieważ G$\{x^{56}\}$ < 59,297, strategii $\{x^{56}\}$ nie dodajemy ani do zbioru **LS**, ani do zbioru **LSB**:

Strategia$\{x^{32}\}$

a) obliczamy G$\{x^{32}\}$ = 59,36

b) dodajemy strategię $\{x^{32}\}$ do zbioru **LSC**:
LSC := **LSC** ∪ $\{x^{32}\}$ = { $\{x^{64}\}$, $\{x^{63}\}$, $\{x^{62}\}$, $\{x^{60}\}$, $\{x^{56}\}$ $\{x^{48}\}$ $\{x^{32}\}$ }

c) ponieważ G$\{x^{32}\}$ > 59,297, strategię $\{x^{32}\}$.dodajemy zarówno do zbioru **LS**, jak i do zbioru **LSB**:
LS := **LS** ∪ $\{x^{32}\}$ = {$\{x^{64}\}$, $\{x^{32}\}$, $\{x^{60}\}$}
LSB := **LSB** ∪ $\{x^{32}\}$ = {$\{x^{60}\}$, $\{x^{32}\}$ }

10. Przechodzimy do kroku 4.
4. Ponieważ **LSB** ≠ ∅, przechodzimy do kroku 5.
5. Ze zbioru **LSB** wybieramy strategię $\{x^{60}\}$ = (B, D, F, G, J, L), usuwamy ją z tego zbioru:
LSB:= **LSB** \ $\{x^{60}\}$ = { $\{x^{32}\}$
6. Określamy wszystkie strategie zmodyfikowane, różniące się od strategii $\{x^{60}\}$ decyzją w jednym stanie. Otrzymujemy następujące strategie:
$\{x^{59}\}$ = {B, D, F, G, J, K} $\{x^{52}\}$ = (B, D, E, G, J, L),
$\{x^{58}\}$ = (B, D, F, G, I, L), $\{x^{44}\}$ = (B, C, F, G, J, L),
$\{x^{64}\}$ = (B, D, F, H, J, L), } $\{x^{28}\}$ = (A, D, F, G, J, L),
i umieszczamy je w zbiorze **M**$\{x^{60}\}$:
M$\{x^{60}\}$:= { $\{x^{59}\}$, $\{x^{58}\}$, $\{x^{64}\}$, $\{x^{52}\}$, $\{x^{44}\}$, $\{x^{28}\}$ }.
7. Sprawdzamy, czy zbiór **M**$\{x^{60}\}$ zawiera strategie, które znajdują się również w zbiorach **LS**, **LSB** i **LSC**. Otrzymujemy:
M$\{x^{60}\}$ ∩ **LS** = ∅
M$\{x^{60}\}$ ∩ **LSB** = ∅
M$\{x^{60}\}$ ∩ **LSC** = $\{x^{64}\}$,
stąd
M$\{x^{60}\}$:= **M**$\{x^{60}\}$ \ (**M**$\{x^{60}\}$∩**LS**) \ (**M**$\{x^{60}\}$∩**LSB**) \ (**M**$\{x^{60}\}$∩**LSC**) =.
{ $\{x^{59}\}$, $\{x^{58}\}$, $\{x^{52}\}$, $\{x^{44}\}$, $\{x^{28}\}$ }..
8. Mamy **M**$\{x^{60}\}$ ≠ ∅.

9. Rozpatrujemy kolejne strategie $\{x^m\}$ ze zbioru **M**$\{x^{60}\}$. Ponieważ
   G$\{x^{59}\}$ = 54,88 < 59,297 G$\{x^{44}\}$ = 57,596 < 59,297
   G$\{x^{58}\}$ = 56,84 < 59,297 G$\{x^{28}\}$ = 58,184 < 59,297
   G$\{x^{52}\}$ = 57,4 < 59,297
   żadnej ze strategii ze zbioru_**M**$\{x^{32}\}$ nie dołączamy do zbioru **LS**, ani do zbioru **LSB**, stąd:
   LS = $\{\{x^{64}\}, \{x^{32}\}, \{x^{60}\}\}$
   LSB = $\{ \{x^{32}\} \}$
   Po wykonaniu kolejnych operacji otrzymujemy:
   LSC = $\{ \{x^{64}\}, \{x^{63}\}, \{x^{62}\}, \{x^{60}\}, \{x^{56}\} \{x^{48}\}, \{x^{32}\},\{x^{59}\}, \{x^{58}\}, \{x^{52}\}, \{x^{44}\}, \{x^{28}\}\}$
10. Przechodzimy do kroku 4.
4. Ponieważ **LSB** $\neq \varnothing$, przechodzimy do kroku 5.
5. Ze zbioru **LSB** wybieramy strategię $\{x^{32}\}$ = (A, D, F, H, J, L), usuwamy ją z tego zbioru:
   LSB:= **LSB** $\setminus \{x^{32}\} = \varnothing$
6. Określamy wszystkie strategie zmodyfikowane, różniące się od strategii $\{x^{32}\}$ decyzją w jednym stanie. Otrzymujemy następujące strategie:
   $\{x^{31}\}$ = {A, D, F, H, J, K} $\{x^{24}\}$ = {A, D, E, H, J, L}
   $\{x^{30}\}$ = {A, D, F, H, I, L} $\{x^{16}\}$ = {A, C, F, H, J, L}
   $\{x^{28}\}$ = {A, D, F, G, J, L} $\{x^{64}\}$ = {B, D, F, H, J, L}
   i umieszczamy je w zbiorze **M**$\{x^{32}\}$:
   M$\{x^{32}\} := \{ \{x^{31}\}, \{x^{30}\}, \{x^{28}\}, \{x^{24}\}, \{x^{16}\}, \{x^{64}\} \}$.
7. Sprawdzamy, czy zbiór **M**$\{x^{32}\}$ zawiera strategie, które znajdują się również w zbiorach **LS**, **LSB** i **LSC**. Otrzymujemy:
   M$\{x^{32}\} \cap$ **LS** = $\varnothing$
   M$\{x^{32}\} \cap$ **LSB** = $\varnothing$
   M$\{x^{32}\} \cap$ **LSC** = $\{ \{x^{64}\}, \{x^{28}\} \}$
   stąd
   M$\{x^{32}\}$:= **M**$\{x^{32}\} \setminus$ (**M**$\{x^{32}\}\cap$**LS**) $\setminus$ (**M**$\{x^{32}\}\cap$**LSB**) $\setminus$ (**M**$\{x^{32}\}\cap$**LSC**) =
   $\{ \{x^{31}\}, \{x^{30}\}, \{x^{28}\}, \{x^{24}\}, \{x^{16}\}, \}$.
8. Mamy **M**$\{x^{32}\} \neq \varnothing$.
9. Rozpatrujemy kolejne strategie $\{x^m\}$ ze zbioru **M**$\{x^{32}\}$. Ponieważ
   G$\{x^{31}\}$ = 54,46 < 59,297 G$\{x^{24}\}$ = 57,736 < 59,297
   G$\{x^{30}\}$ = 57,26 < 59,297 G$\{x^{16}\}$ = 58,1 < 59,297
   żadnej ze strategii ze zbioru_**M**$\{x^{32}\}$ nie dołączamy do zbioru **LS**, ani do zbioru **LSB,** czyli
   LS = $\{\{x^{64}\}, \{x^{60}\}, \{\{x^{32}\} \}$

LSB = ∅

Do zbioru **LSC** dodajemy kolejne rozpatrywane strategie ze zbioru **M**{x^{32}}, stąd:

LSC = { {x^{16}}, {x^{24}}, {x^{28}}, {x^{30}}, {x^{31}}, {x^{32}}, {x^{48}}, {x^{56}}, {x^{60}}, {x^{62}}, {x^{63}}, {x^{64}}, },

10. Przechodzimy do kroku 4.
4. Ponieważ **LSB** = ∅, przechodzimy do kroku 11.
11. Koniec procedury.

Zakończenie

W artykule zdefiniowano pojęcie strategii prawie optymalnych dla dyskretnych, stochastycznych procesów wieloetapowych oraz zaproponowano algorytm pozwalający na znalezienie strategii optymalnych i prawie optymalnych. W rozpatrywanym przykładzie oprócz strategii optymalnej znaleziono dwie strategie prawie optymalne, różniące się od strategii optymalnej nie więcej niż o 1,5%.

Algorytm poddawany jest obecnie testom komputerowym. Czas potrzebny na jego realizację zależny jest od liczby strategii, dla których wartość oczekiwana mieści się w zadanym przez decydenta przedziale tolerancji. Jeżeli przedział ten obejmuje zbyt dużo strategii, obliczenia mogą nie zostać przeprowadzone do końca. Dysponujemy wówczas jednak tymi strategiami, które udało się wyznaczyć do momentu przerwania obliczeń.

Zagadnienie wyznaczania strategii optymalnych i prawie optymalnych znajduje zastosowanie w podejściu hierarchicznym wielokryterialnego programowania dynamicznego. Dalsze prace nad wykorzystaniem tej metody skierowane będą w tym kierunku.

Zakres praktycznych zastosowań proponowanego algorytmu obejmuje wszystkie te problemy, w których występuje sekwencja powiązanych ze sobą decyzji. Z sytuacją taką mamy na przykład do czynienia w zarządzaniu portfelem projektów. Skład portfela podlega ciągłym zmianom. Zakończenie określonego przedsięwzięcia umożliwia wykorzystanie zwolnionych zasobów, przy czym realizowane projekty są często powiązane. Algorytm może być również przydatny w problemach z zakresu zarządzania zdolnością produkcyjną. Ustalając strategię zwiększania możliwości wytwórczych, firma kieruje się głównie finansową oceną inwestycji. Jednocześnie jednak bierze pod uwagę inne, trudno kwantyfikowalne czynniki. Strategia optymalna pod względem finansowym może okazać się mniej korzystna ze względu na inne kryteria. Decydenci mogą zatem być zainteresowani poszukiwaniem takiego rozwiązania, które co prawda jest oceniane nieco gorzej z punktu widzenia finansowego, to jednak jest zdecydowanie bardziej atrakcyjne z punktu widzenia innych ważnych celów organizacji.

Literatura

1. Bellman R. (1957), *Dynamic Programming*, Princeton University Press.
2. Bellman R., Dreyfus S. (1967), *Programowania dynamiczne. Zastosowania*, PWE, Warszawa.
3. Elmaghraby S. E. (1970), *The Theory of Networks and Management Science*, Part 1 „Management Science", Vol. 17.
4. Nowak M. (2014), *Wykorzystanie podejścia quasi-hierarchicznego w wielokryterialnym drzewie* decyzyjnym, w: *Analiza i wspomaganie decyzji*, D. Kopańska-Bródka (red.), „Studia Ekonomiczne. Zeszyty Naukowe Wydziałowe", nr 208, Uniwersytet Ekonomiczny w Katowicach.
5. Nowak M., Trzaskalik T. (2012), *Interactive procedure for a multiobjective stochastic discrete dynamic problem,* „Journal of Global Optimization", Vol. 57, No. 2.
6. Steuer R. E. (2003), *ADBASE: A Multiple Objective Linear Programming Solver for All Efficient Extreme Points and All Efficient Unbounded Edges*, Terry College of Business, University of Georgia, Athens, Georgia.
7. Trzaskalik T. (1998), *Multiobjective Analysis in Dynamic Environmnent*, The Karol Adamiecki University of Economics in Katowice Press, Katowice.
8. Trzaskalik T. (1990), *Wielokryterialne dyskretne programowanie dynamiczne. Teoria i zastosowania w praktyce gospodarczej*, Akademia Ekonomiczna w Katowicach, Katowice.
9. Trzaskalik T. (1986), *Wybrane problemy programowania dynamicznego*, Akademia Ekonomiczna w Katowicach, Katowicach.
10. Trzaskalik T., Do Thien Hoa (1999), *Wielokryterialne, wieloetapowe procesy decyzyjne w warunkach niepewności*, w: *Modelowanie preferencji a ryzyko'99*, część 2, Trzaskalik T. (red.), Akademia Ekonomiczna im. K. Adamieckiego, Katowice.
11. Trzaskalik T., Sitarz S. (2007), *Discrete dynamic programming with outcomes in random variable structures*, „European Journal of Operational Research", Vol. 177.

Streszczenie

W pracy rozpatrujemy wieloetapowe, dyskretne, stochastyczne procesy decyzyjne. Dla decydenta interesujące może być nie tylko znalezienie strategii optymalnej, ale również kolejnych strategii, dla których wartość oczekiwana jest bliska wartości oczekiwanej strategii optymalnej, z dokładnością określoną przez decydenta.

Celem artykułu jest zaproponowanie algorytmu pozwalającego na znalezienie strategii optymalnych i prawie optymalnych. Strategie optymalne znajdujemy, wykorzystując zasadę optymalności Bellmana. Proponowana w niniejszej pracy metoda polega na znalezieniu zbioru strategii optymalnych, a następnie rozszerzaniu tego zbioru o kolejne strategie prawie optymalne, które mieszczą się w obszarze zainteresowań decydenta. Nowe strategie generujemy poprzez zmianę decyzji w jednym stanie dla strategii zaakceptowanej wcześniej. Zaproponowany algorytm ilustrowany jest prostym przykładem liczbowym, wyjaśniającym jego działanie.

Słowa kluczowe

programowanie dynamiczne, strategia, modele stochastyczne

Optimal and Near Optimal Strategies in Discrete Stochastic Dynamic Programming (Summary)

In the paper multistage, discrete stochastic decision processes are considered. For the decision maker it may be interesting not only to find the optimal strategy, but also another strategies , for which their expected values are close to the expected value of the optimal strategy , with the accuracy determined by the decision maker .

The aim of the paper is to propose the algorithm that allows to find optimal and near optimal strategies. We find the optimal strategies using Bellman's principle of optimality. The method proposed in the paper relies on finding the set of optimal strategies and expanding it to the next near optimal strategies that are of interest to the decision maker. New strategies are generated by changing decision in one state only for the strategies approved earlier. The proposed algorithm is illustrated with a simple numerical example explaining of how it works.

Keywords

dynamic programming, strategy, stochastic models