Schemat programowania dynamicznego *(ang. dynamic programming)*

Jest jedną z metod rozwiązywania problemów optymalizacyjnych.

Jej twórcą (1957) był amerykański matematyk *Richard Ernest Bellman*.

Schemat ten określa ogólne podejście polegające na przekształceniu zadania optymalizacyjnego w wieloetapowy proces decyzyjny, w którym stan na każdym etapie zależy od decyzji wybieranej ze zbioru decyzji dopuszczalnych.

Przyjmijmy natępujące oznaczenia:

- d_i - decyzja podejmowana na i-tym etapie procesu,
- S_i - zbiór stanów procesu na kolejnych etapach,
  - s_i - stan przed podjęciem decyzji d_i,
  - s_{i+1} - stan po podjęciu decyzji d_i,
- D_i - zbiór decyzji dopuszczalnych na i-tym etapie procesu,

Przebieg wieloetapowego procesu podejmownaia decyzji można zapisać w postaci trasformacji

$$s_k = T(s_{k-1}, d_{k-1})$$

stąd

$$s_{i+1} = T(s_i, d_i)$$

Z procesem podejmowania decyzji związana jest skalarna funkcja celu służąca do oceny ciągu decyzji $d_1, d_2, \ldots, d_N$

$$F = (s_1, s_2, \ldots, s_N; d_1, d_2, \ldots, d_N)$$

Rozwiązaniem rozważanego problemu jest znalezienie strategii optymalnej, czyli takiego ciągu decyzji $d_1, d_2, \ldots, d_N$, dla którego funkcja F osiaga ekstremum (minimum bądź maksimum).

Jeżeli S_1 jest znany, to przebieg wieloetapowego procesu decyzjnego jest wyznaczony przez ciąg decyzji dopuszczalnych $d_1, d_2, \ldots, d_N$ nazywanych **strategią**.

Wyznaczanie strategii optymalnej w ogólnym przypadku wymaga rozwiązania zadania optymalizacji nieliniowej

$$\min_{d_1,d_2,\ldots,d_N} F(s_1, s_2, \ldots, s_N; d_1, d_2, \ldots, d_N)$$

$$s_k = T(s_{k-1}, d_{k-1}), k = 2, \ldots, N,$$

gdzie S_1 jest dane.

Stąd zysk z zastosowania schematu programowania dynamicznego osiągany jest tylko dla pewnej podklasy problemów posiadających **własność Markowa**, która mówi, że

przebieg procesu począwszy od etapu k dalej zależy tylko od stanu s_k

Z własności Markowa wynika też **zasada optymalności Bellmana**, mówiąca że

strategia optymalna ma tę własność, że niezależnie od stanu początkowego i decyzji początkowej pozostałe decyzje muszą tworzyć strategię optymalną z punktu widzenia stanu winikłego z pierwszej decyzji.

Rozważmy przykład maksymalizacji funkcji addytywnej.

$$F(s_1, s_2, \ldots, s_N; d_1, d_1, \ldots, s_N) = \sum_{i=1}^{N} f_i(s_i, d_i)$$

Zgodnie z zasadą optymalności Bellmana mamy

$$\max_{d_1, d_2, \ldots, d_N} (f_1(s_1, d_1) + \ldots + f_N(s_N, d_N)) =$$

$$\max_{d_1} (f_1(s_1, d_1) + \max_{d_k, \ldots, d_N} (f_2(s_2, d_2) + \ldots + f_N(s_N, d_N)))$$

Oznaczając

$$g_k(s_k) = \max_{d_k, \ldots, d_N} (f_k(s_k, d_k) + \ldots + f_N(s_N, d_N))$$

otrzymamy *równania Bellmana*

$$g_N(s_N) = \max_{d_N} f_N(s_N, d_N)$$

$$g_k(s_k) = \max_{d_k} f_k(s_k, d_k) + g_{k+1}(s_{k+1}))$$

Wynik można zapisać w postaci *równania rekurencyjnego Bellmana*

$$g_k(s_k) = \max_{d_k} f_k(s_k, d_k) + g_{k+1}(T(s_k, d_k)))$$

Wykorzystanie równań Bellmana można przedstawić w postaci następującego algorytmu:

krok 1. etap n wyznacz

$$g_N(s_N) := \max_{d_N \in D_N} \left\{ f_N(s_N, d_N) \right\}$$

kroki 2. . . . , $n-1$**.** etapy $n-k$ wyznacz

$$g_k(s_k) := \max_{d_k \in D_k} \left\{ f_k(s_k, d_k) + g_{k+1}(s_{k+1}) \right\}$$

krok n. etap 1 wyznacz

$$g_1(s_1) := \max_{d_1 \in D_1} \left\{ f_1(s_1, d_1) + g_2(s_2) \right\}$$

Niemniej jednak, do wyboru decyzji optymalnej wykorzystuje się odpowiednio skontruowaną **funkcję dominacji**. Wtedy dycyzja optymalna to taka, dla której wartości funkcji dominacji przyjmuje minimum bądź maksimum.

Pytanie 1. Czy zatem, dla każdego problemu można skonstruować algorytm programowania dynamicznego?

Niemniej jednak, do wyboru decyzji optymalnej wykorzystuje się odpowiednio skontruowanej **funkcji dominacji**. Wtedy dycyzja optymalna to taka, dla której wartości funkcji dominacji przyjmuje minimum bądź maksimum.

Pytanie 1. Czy zatem, dla każdego problemu można skonstruować algorytm programowania dynamicznego?

Niestety nie.

Niemniej jednak, do wyboru decyzji optymalnej wykorzystuje się odpowiednio skontruowanej **funkcji dominacji**. Wtedy dycyzja optymalna to taka, dla której wartości funkcji dominacji przyjmuje minimum bądź maksimum.

Pytanie 1. Czy zatem, dla każdego problemu można skonstruować algorytm programowania dynamicznego?

Niestety nie.

Pytanie 2. Czym powinien się charakteryzować problem, aby można było zastosować programowanie dynamiczne?

Niemniej jednak, do wyboru decyzji optymalnej wykorzystuje się odpowiednio skontruowanej **funkcji dominacji**. Wtedy dycyzja optymalna to taka, dla której wartości funkcji dominacji przyjmuje minimum bądź maksimum.

Pytanie 1. Czy zatem, dla każdego problemu można skonstruować algorytm programowania dynamicznego?

Niestety nie.

Pytanie 2. Czym powinien się charakteryzować problem, aby można było zastosować programowanie dynamiczne?

Powinien posiadać tzw. **optymalność podstruktury**, żeby na każdym etapie procesu podejmując pewną decyzję można uzyskać kolejny podproblem.

Inaczej mówiąc. Problem musi dać się zdekomponować na podproblemy tak, żeby można było zastosować do podjęcia dezycji tę samą funkcję dominacji.

Przykład 1. Problem plecakowy. Sposób znany.

patrz: Tablica

Przykład 1. Problem plecakowy. Sposób nowy.

patrz: Tablica

Przykład 2. Problem podziału. Sposób znany (w skrócie).

patrz: Tablica

Przykład 2. Problem podziału. Sposób nowy (też w skrócie).

patrz: Tablica

Przykład 3. Pan Marek mieszkający w miejscowości A_1 wyrusza na wakacje do miejscowości A_9. Sieć połączeń przedstawiono na rysunku.

Zadanie Wyznaczyć najkrótszą trasę pomiędzy A_1 i A_9.

patrz: Tablica

Pytanie 1. Jak wyznaczyć etapy procesu decyzyjnego?

Pytanie 2. Jakie są zbiory S_i , $(i = 1, 2, 3, 4, 5)$ stanów (z których można wyruszyć)?

Pytanie 3. Jak wyglądają zbiory decyzji dopuszczalnych D_i (dopuszczalnych połączeń pomiędzy miastami A_i, A_j)?

Pytanie 4. Jak określona jest długość trasy (za pomocą $f_i(s_i, d_i)$)?

Przykład 4. / Zadanie 1.

Znaleźć maksimum funkcji

$$f(x, y) = -x + 6y$$

jeśli spełnia określone ograniczenia

$$2x + y + \leq 150$$

$$x + 5y \leq 200$$

$$x \geq 0, y \geq 0$$

Przykład 5. / Zadanie 2.

Wyznaczyć takie nieujemne liczby x, y, z, których iloczyn jest maksymalny a

$$x + 2y + z = 30$$