Аппроксимационные схемы

Определение Алгоритм A называется *ε-аппроксимационной схемой*, если для любого $\varepsilon \in]0,1[$ и любых исходных данных I задачи верно

$$z^A(I) \geq (1-\varepsilon) z^*(I)$$

Другими словами, алгоритм A является $(1-\varepsilon)$-приближенным алгоритмом для всех $\varepsilon \in]0,1[$.

Алгоритм H^ε

1. $l := \min \{\lceil 1/\varepsilon \rceil - 2, n\}$, $z^A := 0$

2. Для всех подмножеств $L \subset J$, $|L| \leq l-1$

   if $(\sum_{j \in L} w_j \leq c)$ and $(\sum_{j \in L} p_j > z^A)$ then $z^A := \sum_{j \in L} p_j$

3. Для всех подмножеств $L \subset J,\ |L| = l$

if $(\sum_{j \in L} w_j \le c)$ then

- Применить алгоритм A^{MG} к задаче с множеством предметов $\{j \mid p_j \le \min\{p_i,\ i \in L\}\} \setminus L$ и вместимостью рюкзака $c - \sum_{j \in L} w_j$
- if $\sum_{j \in L} p_j + z^{MG} > z^A$ then $z^A := \sum_{j \in L} p_j + z^{MG}$.

Теорема Алгоритм H^{ε} является ε-аппроксимационной схемой.

Доказательство. Если оптимальное решение z^* содержит не более l предметов, то $z^A = z^*$ и утверждение верно.

Пусть в оптимальном решении более чем l предметов. Выберем в нем подмножество L^* из l предметов с наибольшими весами p_j. Рассмотрим подзадачу S с множеством предметов $\{ j \mid p_j \le \min \{p_i,\ i \in L^*\} \} \setminus L^*$ и вместимостью рюкзака $c - \sum_{j \in L^*} w_j$. Оптимальное решение этой подзадачи обозначим через z_S^*. Тогда $z^* = z_S^* + \sum_{j \in L^*} p_j$. Приближенное решение, полученное алгоритмом A^{MG}, обозначим через z_S^{MG}.

По определению $z^A \ge \sum_{j \in L^*} p_j + z_S^{MG}$ и, кроме того, $z_S^{MG} \ge \frac{1}{2} z_S^*$.

Рассмотрим два случая:

1. $\sum_{j\in L^*} p_j \ge \frac{l}{l+2} z^*$. Тогда $z^A \ge \sum_{j\in L^*} p_j + z_S^{MG} \ge \sum_{j\in L^*} p_j + \frac{1}{2} z_S^* =$

$$\sum_{j\in L^*} p_j + \frac{1}{2}(z^* - \sum_{j\in L^*} p_j) = \frac{1}{2}(z^* + \sum_{j\in L^*} p_j) \ge \frac{1}{2}(z^* + \frac{l}{l+2} z^*) = \frac{l+1}{l+2} z^*$$

2. $\sum_{j\in L^*} p_j < \frac{l}{l+2} z^*$. Тогда в L^* найдется предмет с $p_j < \frac{1}{l+2} z^*$. По определению в подзадаче S все предметы имеют вес не более $\frac{1}{l+2} z^*$. Применяя свойство 2 для *LP*-релаксаций, получаем

$$z^* = \sum_{j\in L^*} p_j + z_S^* \le \sum_{j\in L^*} p_j + z_S^{MG} + \frac{1}{l+2} z^* \le z^A + \frac{1}{l+2} z^*.$$

Итак, в обоих случаях получаем $z^A \ge \frac{l+1}{l+2} z^*$. Величина $\frac{l+1}{l+2}$ растет с ростом l и $\frac{l+1}{l+2} \ge \frac{1/\varepsilon - 1}{1/\varepsilon} = 1 - \varepsilon$. ■

$$T = O(n\, n^l) = \mathrm{O}(n^{l+1}). \quad \Pi = O(n).$$

Пример Положим $n = \left\lceil \frac{1}{\varepsilon} \right\rceil + 1$, $c = \left\lceil \frac{1}{\varepsilon} \right\rceil M$ и

$$p_1 = 2, \quad p_2 = p_3 = \ldots = p_n = M,$$

$$w_1 = 1, \quad w_2 = w_3 = \ldots = w_n = M.$$

Оптимум $z^* = M(l+2)$, $l = n-3$, $z^A = (l+1)M + 2$ и

$$z^A / z^* \to (l+1)/(l+2) \quad \text{при} \quad \text{M} \to \infty.$$

Определение ε-аппроксимационная схема A называется *полиномиальной*, если ее трудоемкость полиномиально зависит от длины записи исходных данных задачи.

$T_H = O(n^{l+1}) = O(n^{1/\varepsilon - 1})$ — полиномиальная зависимость при заданном ε. Если $\varepsilon = 0.1$, то $T_H = O(n^9)$, то есть алгоритм H^ε является полиномиальной ε-аппроксимационной схемой для задачи о рюкзаке.

Определение ε-аппроксимационная схема A называется *полностью полиномиальной*, если ее трудоемкость полиномиально зависит от длины записи исходных данных задачи и величины $1/\varepsilon$.

Теорема Для задачи о рюкзаке существует полностью полиномиальная ε-аппроксимационная схема.

Доказательство Для примера $I = \{p_1,\ldots, p_n, w_1,\ldots, w_n, c\}$ построим новый пример $\bar{I}$, в котором $\bar{c} = c, \bar{w}_j = w_j, \bar{p}_j = \left\lfloor \frac{p_j}{K} \right\rfloor$ для некоторой константы $K > 0$, которую определим позже. Для примера $\bar{I}$ применим алгоритм ДП и найдем оптимальный набор предметов $\bar{X}$. Скорее всего он будет отличаться от оптимального набора X^* для примера I. Оценим разность между полученным значением z^A и оптимальным z^*:

$$z^A = \sum_{j\in\bar{X}} p_j \geq \sum_{j\in\bar{X}} K\left\lfloor \frac{p_j}{K} \right\rfloor \geq K \sum_{j\in X^*} \left\lfloor \frac{p_j}{K} \right\rfloor \geq \sum_{j\in X^*} K(\frac{p_j}{K} - 1) =$$

$$= \sum_{j\in X^*} (p_j - K) = z^* - |X^*|K.$$

Второе неравенство в этой цепочке следует из оптимальности $\bar{X}$ для $\bar{I}$.

Мы хотим получить $\dfrac{z^* - z^A}{z^*} \le \dfrac{|X^*|K}{z^*} \le \varepsilon.$

Следовательно, $K \le \dfrac{\varepsilon \cdot z^*}{|X^*|}$. Так как $n \ge |X^*|$ и $z^* \ge p_{\max}$, то

$$\frac{\varepsilon \cdot z^*}{|X^*|} \ge \frac{\varepsilon \cdot z^*}{n} \ge \frac{\varepsilon \cdot p_{\max}}{n}$$

и, полагая $K = \varepsilon \cdot p_{\max}/n$, получим нужное значение для параметра K.

Трудоемкость алгоритма определяется трудоемкостью ДП. Если вместо исходной задачи решать обратную к ней, то $T = O(Un)$, где U — верхняя оценка на оптимальное значение целевой функции $\bar{z}^* = \sum_{j \in \bar{X}} \bar{p}_j$.

Очевидно, что $\bar{z}^* \le n\bar{p}_{\max}$, но $\bar{p}_{\max} \le \frac{p_{\max}}{K} = \frac{n}{\varepsilon}$, то есть $\bar{z}^* \le n^2/\varepsilon$.

Полагая $U = n^2/\varepsilon$, получаем $T = O(n^3 \cdot \frac{1}{\varepsilon})$, $\Pi = O(Un) = O(n^3 \cdot \frac{1}{\varepsilon})$. ■

Задача о ближайшем соседе

Дано: функция $f(x, y) \geq 0$ — затраты на обслуживание отрезка дороги от x до y, $0 \leq x \leq y \leq M$, x, y — целочисленные точки, n — число отрезков.

Найти: оптимальное разбиение сегмента $[0,M]$ на n отрезков.

Математическая модель:

$$\min \sum_{k=1}^{n} f(x_{k-1}, x_k)$$

$$0 = x_0 \leq \ldots \leq x_n = M$$

Алгоритм динамического программирования

S_k(y) — минимальные затраты на обслуживание k отрезков для сегмента $[0, y]$.

Рекуррентные соотношения:

$$S_1(y) = f(0,y), \quad y = 1,\ldots, M$$

$$S_k(y) = \min_{1 \le x \le y} \{S_{k-1}(x) + f(x,y)\}, \quad y = 1,\ldots, M, \quad k = 2,\ldots, n.$$

$$T = O(nM^2) \quad \Pi = O(nM)$$

Оптимизация числа отрезков

Для каждого $n = 1,\ldots, M$ найти $S_n(M)$ и выбрать наименьшее значение

$T = O(M^3), \quad П = O(M^2).$

Модифицированный вариант

$\widetilde{S}(y)$ — минимальные затраты на обслуживание сегмента $[0, y]$.

Рекуррентные соотношения:

$$\widetilde{S}(0) = 0,$$

$$\widetilde{S}(y) = \min_{0 \le x \le y-1} \{\widetilde{S}(x) + f(x, y)\}, \quad y = 1,\ldots, M.$$

$$T = O(M^2), \quad П = O(M).$$

Задача замены оборудования

Приведение затрат к начальному моменту

Пусть χ — банковский процент, или коэффициент эффективности капиталовложений (годовая норма дисконта).

Если S_1 — капитал в начальный год, то по истечении года эта сумма станет равной $S_2 = S_1 \cdot (1 + \chi)$, а в конце t-го года $S_t = S_1 \cdot (1 + \chi)^{t-1}$.

Если в год t хотим потратить сумму S_t, то в начальный год должны иметь $S_1 = \dfrac{S_t}{(1+\chi)^{t-1}}$. Затраты S_t в год t, будучи приведенными к начальному моменту $t = 1$, равны

$$\widetilde{S} = \alpha^{t-1} \cdot S_t,$$

где $\alpha = \dfrac{1}{1+\chi}$ — *коэффициент дисконтирования*, $0 < \alpha \le 1$.

Если в течении T лет производились траты S_1, S_2,…, S_t то суммарные приведенные затраты вычисляются по формуле:

$$\widetilde{S} = \sum_{t=1}^{T} \alpha^{t-1} S_t .$$

Пример Рассмотрим распределение капитала в 8 млн. руб. в течение 8 лет при банковском проценте $\chi = 0.1$ $(\alpha = 0.91)$ и трех стратегиях:

1) все траты в 1-й год;
2) равномерные траты;
3) все траты в последний год.

Стратегия	Год								Суммарные приведенные затраты
	1	2	3	4	5	6	7	8	
1	8	–	–	–	–	–	–	–	8.0
2	1	1	1	1	1	1	1	1	5.9
3	–	–	–	–	–	–	–	8	4.2

Постановка задачи

g — начальная стоимость оборудования;

c_t — стоимость эксплуатации оборудования в t год, $c_{t+1} \geq c_t$;

Система функционирует бесконечно, оборудование периодически заменяется.

T — период замены оборудования;

S_T — суммарные затраты при фиксированном периоде замены T :

$$S_T = g + \sum_{t=1}^{T} \alpha^{t-1} c_t + \alpha^T g + \sum_{t=1}^{T} \alpha^{T+t-1} c_t + \alpha^{2T} g + \sum_{t=1}^{T} \alpha^{2T+t-1} c_t + \dots .$$

за первые T лет за вторые T лет ……

Благодаря дисконтированию затрат ($\alpha < 1$) величина S_T конечна:

$$S_T = (g + \sum_{t=1}^{T} \alpha^{t-1} c_t)(1 + \alpha^T + \alpha^{2T} + \dots) = (g + \sum_{t=1}^{T} \alpha^{t-1} c_t)/(1 - \alpha^T) < \infty .$$

Задача отыскания оптимального периода замены оборудования: $S_T \to \min\limits_{T>0}$

Применение динамического программирования

Рассмотрим систему, функционирующую в течение T лет, причем решение о замене оборудования принимается каждый год.

Дано: $\{1,\ldots, m\}$ — набор типов оборудования;

g_t^i — стоимость оборудования i-го типа, купленного в год t;

$c_t^i(\tau)$ — стоимость годовых эксплуатационных затрат на оборудование i-го типа, купленного в год t и проработавшего τ лет;

$\Phi_t^i(\tau)$ — остаточная стоимость оборудования i-го типа возраста τ, купленного в год t;

n — максимально допустимый возраст оборудования;

$i_0,\ \tau_0$ — тип и возраст оборудования в начале функционирования;

Пусть $S_t^i(\tau)$ — минимальные суммарные затраты в интервале $[t,\ T]$, приведенные к началу t-го года, при условии, что в начале t-го года было оборудование типа i возраста τ. Требуется найти $S_1^{i_0}(\tau_0)$.

Рекуррентные соотношения:

$$S_T^i(\tau) = \min \begin{cases} c_{T-\tau}^i(\tau+1) - \alpha \Phi_{T-\tau}^i(\tau+1), & \text{если замены нет,} \\ \min\limits_{1 \le k \le m} [g_T^k + c_T^k(1) - \alpha \Phi_T^k(1)] - \Phi_{T-\tau}^i(\tau), & \text{в случае замены,} \end{cases}$$

$$1 \le \tau \le n, \quad 1 \le i \le m,$$

$$S_t^i(\tau) = \min \begin{cases} c_{t-\tau}^i(\tau+1) + \alpha S_{t+1}^i(\tau+1), & \text{если продолжаем эксплуатировать оборудование} \\ \min\limits_{1 \le k \le m} [g_t^k + c_t^k(1) + \alpha S_{t+1}^k(1)] - \Phi_{t-\tau}^i(\tau), & \text{если заменяем оборудование} \end{cases}$$

$$1 \le \tau \le n, \quad 1 \le i \le m, \quad 1 \le t < T.$$

Алгоритм может быть реализован с трудоемкостью $T = O(m^2 nT)$, и памятью $П = O(mnT)$.