ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Ноябрь—декабрь 2008. Том 15, № 6. С. 58–62

УДК 519.857

ДИНАМИЧЕСКОЕ ПРОГРАММИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ МНОЖЕСТВ ПАРЕТО

В. И. Струченков

Аннотация. Рассмотрена задача об оптимальном распределении ресурса. Для её решения предложена схема динамического программирования. Вместо рекуррентных уравнений используется пошаговое вычисление множества точек, оптимальных по Парето, на плоскости значений целевой функции и ресурса. Это позволяет экономить машинную память и сокращает время счёта. Эффективность подхода демонстрируется на примерах.

Ключевые слова: динамическое программирование, множество Парето.

1. Постановка задачи

Рассмотрим задачу, которую содержательно можно представить как поиск наилучшего способа защиты поверхности от агрессивного воздействия внешней среды. Интенсивность этого воздействия различна на разных участках поверхности. Поверхность разбивается на n элементов, и для каждого из них может быть выбран один из имеющихся способов защиты. Через X_i обозначим конечное множество возможных способов защиты элемента i ($i = 1, \dots, n$). Каждый способ $x_i \in X_i$ характеризуется неотрицательной величиной $f_i(x_i)$ ущерба, наносимого элементу i при данном способе защиты, и экономическим эффектом $g_i(x_i)$ от использования способа защиты x_i элемента i. Ущерб связан с остаточным воздействием внешней среды при неполной защите элемента, а экономический эффект представляет собой экономию затрат по сравнению с затратами на полную защиту элемента.

Требуется выбрать такие способы защиты элементов поверхности, при которых суммарный экономический эффект был бы максимален, а суммарный ущерб по всем элементам не превосходил заданной величины R.

В принятых обозначениях формальная постановка задачи звучит так: найти максимум суммы

$$\sum_{i=1}^{n} g_i(x_i) \tag{1}$$

при ограничениях

$$\sum_{i=1}^{n} f_i(x_i) \leqslant R, \quad x_i \in X_i, \ i = 1, \ldots, n. \tag{2}$$

Ущербы $f_i(x_i)$ и величину R считаем целыми неотрицательными числами, чего всегда можно добиться умножением на подходящий множитель. Предполагаем, что множество (2) допустимых решений непусто.

2. Схема динамического программирования

Задача (1), (2) представляет собой известную задачу оптимального распределения ресурса, для решения которой используется, в частности, метод динамического программирования [1, 2]. Приведём основные соотношения этого метода. Обозначим через $B_m(r)$ оптимум следующей задачи: найти максимум суммы $\sum\limits_{i=1}^{m} g_i(x_i)$ при ограничениях

$$\sum_{i=1}^{m} f_i(x_i) \leqslant r, \quad x_i \in X_i, \ i = 1, \ldots, m,$$

где m принимает значения $1, \ldots, n$, r — значения $0, 1, \ldots, R$. Очевидно, величина $B_n(R)$ равна оптимуму исходной задачи (1), (2). Её расчёт производится по рекуррентным уравнениям

$$B_m(r) = \max_{\{x_m \in X_m | f_m(x_m) \leqslant r\}} \{B_{m-1}(r - f_m(x_m)) + g_m(x_m)\}, \tag{3}$$

$$r = 0, \ldots, R, \ m = 1, \ldots, n.$$

Полагаем $B_0(r) = 0$, $r = 0, \ldots, R$, и $B_m(r) = -\infty$, если максимум в (3) берётся по пустому множеству.

При больших значениях R и n расчёт с использованием уравнений (3) требует значительного объёма памяти и времени счёта. Ниже предлагается подход, который позволяет существенно экономить вычислительные ресурсы.

На плоскости двух переменных введём отношение частичного порядка:

$$(a, b) \succ (c, d) \Longleftrightarrow a \leqslant c, \ b \geqslant d.$$

Пусть A — некоторое множество точек на плоскости. Точки из A, максимальные относительно частичного порядка, называют *оптимальными по Парето* или просто *паретовскими*.

Рассмотрим множество точек (F, G) вида

$$F = \sum_{i=1}^{m} f_i(x_i), \quad G = \sum_{i=1}^{m} g_i(x_i),$$

где вектор $(x_1, \ldots, x_m)$ пробегает все значения, удовлетворяющие условиям

$$\sum_{i=1}^{m} f_i(x_i) \leqslant R, \quad x_i \in X_i, \ i = 1, \ldots, m.$$

Совокупность паретовских точек этого множества обозначим через S_m. Из нескольких равных паретовских точек в множество S_m включается только одна. Обозначим через (F_{mk}, G_{mk}), $k = 1, \ldots, K_m$, точки множества S_m, нумеруя их по возрастанию координат, т. е.

$$F_{m1} < F_{m2} < \ldots < F_{mK_m}, \ G_{m1} < G_{m2} < \ldots < G_{mK_m}.$$

Нетрудно видеть, что $B_m(F_{mk}) = G_{mk}$, $k = 1, \ldots, K_m$. Функция $B_m(r)$ является неубывающей по аргументу r при данном m. Её график состоит из участков постоянства и точек роста, которые и составляют множество S_m. Таким образом, множество S_m содержит всю необходимую информацию о функции $B_m(r)$ в минимальном объёме.

Множества S_m, $m = 1, \ldots, n$, пересчитываются по шагам, аналогично уравнениям (3). На начальном шаге полагаем $S_0 = \{(0, 0)\}$. Опишем общий шаг. Пусть уже построено множество

$$S_{m-1} = \{(F_{m-1,k}, G_{m-1,k}) \mid k = 1, \ldots, K_{m-1}\}.$$

Рассмотрим множество точек (F, G) вида

$$F = F_{m-1,k} + f_m(x_m), \quad G = G_{m-1,k} + g_m(x_m),$$

где $k = 1, \ldots, K_{m-1}$, а переменная x_m пробегает все значения, удовлетворяющие условиям

$$F_{m-1,k} + f_m(x_m) \leqslant R, \quad x_m \in X_m.$$

Выделяя из этого множества паретовские точки и оставляя из равных точек только одну, получаем множество

$$S_m = \{(F_{mk}, G_{mk}) \mid k = 1, \ldots, K_m\}.$$

Этот процесс завершается построением множества

$$S_n = \{(F_{nk}, G_{nk}) \mid k = 1, \ldots, K_n\}.$$

Величина G_{nK_n} равна оптимуму исходной задачи.

Здесь использовано такое свойство решения: перспективные пары (F, G) должны образовывать множество Парето, а все остальные можно удалить. Это утверждение достаточно очевидно, поэтому его обоснование здесь не приводится.

В целях экономии вычислений можно предварительно выделить для каждого значения индекса $i = 1, \ldots, n$ паретовские точки множества $\{(f_i(x_i), g_i(x_i)) \mid x_i \in X_i\}$ и использовать в расчётах только их.

3. Примеры задач

Рассмотрим пример практической задачи (1), (2) со следующими характеристиками. Число элементов n равно 400. Числа $|X_i|$ способов защиты равны 13, 42 или 45 для различных элементов. Значения ущербов $f_i(x_i)$ лежат в пределах от 0 до 80000, а величина суммарного допустимого ущерба R равна 10^5.

Решение задачи по схеме динамического программирования, отличающейся от стандартной (3) тем, что на каждом шаге рассматриваются только реально достижимые значения ресурса («состояния системы») и из всех вариантов достижения каждого из состояний остаётся один наилучший, потребовало около часа машинного времени. С помощью предложенного подхода пересчёта паретовских множеств S_m задача было решена за 12 с. Максимальное число точек в множествах S_m составило 11954.

Рассмотрим задачу, приведённую в [2], о загрузке машины грузоподъёмностью 35 тонн выборкой из 6 предметов, веса которых равны 4, 7, 11, 12, 16 и 20 тонн, а стоимости задаются величинами 7, 10, 15, 20, 27 и 34 соответственно. Каждый предмет имеется только в одном экземпляре. Требуется выбрать те предметы, сумма весов которых не превышает 35 тонн, а суммарная стоимость максимальна. Очевидно, что, решая эту задачу полным перебором, нужно рассмотреть не более 64 вариантов. Решение этой задачи с помощью уравнений (3) требует на каждом шаге расчёта 35 значений функции $B_m(r)$. Паретовские множества S_m, $m = 1, \ldots, 6$, содержат 2, 4, 7, 13, 15 и 16 точек соответственно.

Отметим, что если веса предметов выражаются целыми числами не тонн, а килограммов, то по стандартному алгоритму [2] на каждом шаге

расчёта рекуррентных соотношений (3) требуется вычислить и запомнить 35000 значений функции $B_m(r)$. В то же время при полном переборе по-прежнему нужно рассмотреть не более 64 вариантов, и множества S_m при любых значениях весов и стоимостей предметов включают не более 64 точек.

ЛИТЕРАТУРА

1. **Беллман Р., Дрейфус С.** Прикладные задачи динамического программирования. — М.: Наука, 1965. — 458 с.
2. **Вентцель Е. С.** Исследование операций: задачи, принципы, методология. 3-е изд. — М.: Дрофа, 2004. — 208 с.

Струченков Валерий Иванович,
e-mail: str1942@ mail.ru

Статья поступила
27 мая 2008 г.
Переработанный вариант —
23 сентября 2008 г.