УДК 519.854

**В.А. МИХАЙЛЮК**

# РЕОПТИМИЗАЦИЯ ЗАДАЧИ О ПОКРЫТИИ МНОЖЕСТВАМИ

**Ключевые слова:** *реоптимизация, r-приближенный алгоритм.*

В настоящее время при решении NP-трудных задач возникла новая вычислительная парадигма — реоптимизация [1–7]. Наиболее известный подход к решению таких задач состоит в разработке и использовании приближенных алгоритмов, которые за полиномиальное время дают решение, близкое к оптимальному. Качество приближенного решения (обычно измеряемого как отношение значений приближенного и оптимального решений) должно быть гарантировано. Направление исследований, связанное с разработкой лучших приближенных алгоритмов, и классификация проблем, основанная на качестве приближения, которое достигается за полиномиальное время, занимают важное место в теоретических исследованиях информатики за последние два десятилетия.

Понятие реоптимизации состоит в следующем. Пусть П — некоторая NP-трудная (возможно, NP-полная) проблема, $I$ — начальный экземпляр проблемы П, оптимальное решение которого известно. Предлагается новый экземпляр $I'$ задачи П, полученный некоторыми «незначительными» изменениями экземпляра $I$. Как можно эффективно использовать знания об оптимальном решении $I$ для вычисления точного или приближенного решения экземпляра $I'$?

Цель реоптимизации при применении приближенных методов — использование знаний о решении начального экземпляра $I$ для: либо достижения лучшего качества приближения (аппроксимационного отношения) $I'$; либо создания более эффективного (быстрого по времени) алгоритма определения оптимального или близкого к нему решения $I'$; либо выполнения первого и второго условий.

Известны следующие результаты по реоптимизации дискретных задач оптимизации. При вставке элементарной дизъюнкции реоптимизация Max Weighted Sat (взвешенная задача о выполнимости на максимум) аппроксимируема с отношением 0,81, хотя Max Weighted Sat аппроксимируема с отношением 0,77 [7]. При вставке вершины в граф реоптимизация Min Vertex Cover (минимальное вершинное покрытие графа) аппроксимируема с отношением 1,5, Min Vertex Cover — с отношением 2 [7]. При вставке вершины (терминальной или нет) реоптимизация Min Steiner Tree (минимальное дерево Штейнера) аппроксимируема с отношением 1,5, Min Steiner Tree — с отношением $1+\ln(3)/2\approx 1,55$ [4].

Следует отметить цикл работ по реоптимизации задачи о коммивояжере (TSP — Travelling Salesman Problem) [1–3, 5]. Например, задача Minimum Metric TSP (Min TSP — задача о коммивояжере на минимум с метрическими расстояниями) аппроксимируема с отношением 1,5, ее реоптимизация при вставке нового узла — с отношением 1,34, реоптимизация этой задачи при изменении расстояний — с отношением 1, 4 [7]. Для общей задачи о коммивояжере (Min TSP) неизвестны оценки аппроксимации как для нее самой, так и для различных версий реоптимизации.

## ПРЕДВАРИТЕЛЬНЫЕ СВЕДЕНИЯ О ПРИБЛИЖЕННЫХ АЛГОРИТМАХ И АППРОКСИМАЦИОННЫХ КЛАССАХ ЗАДАЧ ДИСКРЕТНОЙ ОПТИМИЗАЦИИ

Приведем некоторые необходимые понятия для дальнейшего изложения [7].

**Определение 1** [7]. NP-оптимизационная (NPO) проблема П определяется как четверка $(E, \mathrm{Sol}, m, \mathrm{opt})$ такая, что:

- $E$ — множество экземпляров П, распознаваемое за полиномиальное время;
- для данного $I \in E$ $\mathrm{Sol}(I)$ — множество допустимых решений $I$; для каждого $S \in \mathrm{Sol}(I)$ $|S|$ (размер $S$) полиномиален по $|I|$ (размер $I$); для любого данного $I$ и любого $S$ (полиномиального по $|I|$) за полиномиальное время можно определить $S \in \mathrm{Sol}(I)$;
- для данных $I \in E$ и $S \in \mathrm{Sol}(I)$ $m(I, S)$ — значение (числовое) $S$; $m$ полиномиально вычислима и называется целевой функцией;
- $\mathrm{opt} \in \{\min, \max\}$ — тип оптимизационной проблемы.

Для данной NPO-проблемы $\Pi = \{E, \mathrm{Sol}, m, \mathrm{opt}\}$ оптимальное решение экземпляра $I$ с П обозначим $S^*(I)$, его величину — $m(I, S^*(I)) - \mathrm{opt}(I)$.

**Определение 2** [7]. Для данной NPO-проблемы $\Pi = (E, \mathrm{Sol}, m, \mathrm{opt})$ приближенный (аппроксимационный) алгоритм $A$ — это алгоритм, который для данного экземпляра $I$ с П выдает допустимое решение $S \in \mathrm{Sol}(I)$.

Если $A$ выполняется за полиномиальное время от $|I|$, то $A$ называется полиномиальным приближенным алгоритмом для П.

Качество приближенного алгоритма оценивается как отношение $\rho_A(I)$ (аппроксимационное отношение) между значением приближенного решения $m(I, A(I))$ и значением оптимального решения $\mathrm{opt}(I)$. Таким образом, для минимизационных проблем аппроксимационное отношение находится в пределе $[1, \infty)$, для максимизационных — в $[0, 1]$.

Относительно качества приближенных алгоритмов NPO-проблемы классифицируют следующим образом.

**Определение 3** [7]. NPO-проблема П принадлежит к классу $APX$, если существует полиномиально приближенный алгоритм $A$ и рациональное число $r$ такое, что для данного $I$ с П $\rho_A(I) \le r$ (соответственно $\rho_A(I) \ge r$), если П — минимизационная (соответственно максимизационная) проблема. В этом случае $A$ называется $r$-приближенным (аппроксимационным) алгоритмом (проблема П — $r$-аппроксимируема алгоритмом $A$).

Примерами комбинаторных оптимизационных проблем, принадлежащих к классу $APX$, являются Max Weighted Sat, Min Vertex Cover, Min TSP. Для отдельных проблем из $APX$ можно ввести более сильную форму аппроксимационности.

Для таких проблем для любого рационального $r > 1$ (или $r \in (0,1)$ для максимизационных проблем) существует алгоритм $A_r$ и подходящий полином $p$ такой, что $A_r$ — $r$-аппроксимационный (приближенный) алгоритм со временем, измеряемым как $p$ от $|I|$. Семейство алгоритмов $A_r$ (параметризованное с помощью $r$) называется полиномиальной приближенной схемой (PTAS — Polynomial Time Approximation Scheme).

**Определение 4** [7]. NPO-проблема П принадлежит к классу $PTAS$, если для любого рационального $r > 1$ (соответственно $r \in (0,1)$) и любого экземпляра $I$ с П существует такая полиномиальная приближенная схема $A_r$, что $\rho_{A_r}(I) \le r$ (соответственно $\rho_{A_r}(I) \ge r$) для П — минимизационной (максимизационной) проблемы.

Заметим, что в определении $PTAS$ время алгоритма $A_r$ полиномиально относительно размера входа, однако оно может быть экспоненциально относительно $r-1$. Лучшая ситуация возникает, когда время выполнения полиномиально как по размеру входа, так и по $r-1$ (или $1-r$ для максимизационных проблем). В этом случае алгоритм называется полностью полиномиальной приближенной схемой (FPTAS — Fully Polynomial Time Approximation Scheme).

**Определение 5** [7]. NPO-проблема принадлежит к классу *FPTAS*, если она допускает полностью полиномиальную приближенную схему.

Если $P \neq NP$, то имеет место включение $FPTAS \subset PTAS \subset APX \subset NPO$.

## РЕОПТИМИЗАЦИЯ ЗАДАЧИ О ПОКРЫТИИ МНОЖЕСТВАМИ ПРИ ИЗМЕНЕНИИ МАТРИЦЫ ОГРАНИЧЕНИЙ

Задачу о покрытии множествами (задача $\Pi(A,c)$) рассмотрим в следующей постановке:

$$\min \left\{ f(x) = \sum_{i=1}^{n} c_i x_i \;\middle|\; x \in Q(A) \right\},$$

$$Q(A) = \left\{ x \in B^n \;\middle|\; \sum_{j=1}^{n} a_{ij} x_j \geq 1,\; i = 1, \ldots, m \right\},$$

где $B^n = \{0,1\}^n$, $A = \{a_{ij}\}$, $i = 1, \ldots, m$, $j = 1, \ldots, n$, $A - (m,n)$ — булева матрица $(m \leq n)$, $c_i > 0$, $i = 1, \ldots, n$.

Будем пользоваться следующими результатами.

Известно [8], что жадный алгоритм решает задачу $\Pi(A,c)$ с оценкой точности $\sum_{t=1}^{c(A)} \frac{1}{t} \leq \ln m + 1$, где $c(A)$ — максимальное число единиц в столбце матрицы *A*, т.е. жадный алгоритм — $(\ln m + 1)$-приближенный (аппроксимационный) алгоритм для решения $\Pi(A,c)$.

Введем следующие обозначения для измененных экземпляров задачи $\Pi(A,c)$: $\Pi(A_j^+, c)$ (соответственно $\Pi(A_j^-, c)$) — задача $\Pi(A,c)$ с заменой в столбце *j* матрицы *A* произвольного 0 на 1 (1 на 0); $\Pi(A_j^+(p), c)$ (соответственно $\Pi(A_j^-(p), c)$) — задача $\Pi(A,c)$ с заменой в столбце *j* матрицы *A* произвольного числа $p (1 \leq p < m)$ 0 на 1 (1 на 0); $REOPT(\cdot)$ — соответствующий реоптимизационный алгоритм, исходящий из оптимального решения задачи $\Pi(A,c)$.

Допустимое решение $S = \{j_1, \ldots, j_k\}$ задачи $\Pi(A,c)$ интерпретируем как совокупность столбцов $\{j_1, \ldots, j_k\}$ (подмножеств множества $\{1, \ldots, m\}$), которые составляют покрытие матрицы *A* (т.е. $S = \{j_1, \ldots, j_k\}$ соответствует допустимый вектор $x = (x_1, \ldots, x_n) \in Q(A)$ такой, что $x_{j_1} = x_{j_2} = \ldots = x_{j_k} = 1$; остальные компоненты равны 0). Под весом $c(S)$ решения *S* будем понимать $c(S) = c_{j_1} + \ldots + c_{j_k}$.

**Теорема 1.** Существует $REOPT(\Pi(A_j^+, c))$, являющийся $\left(2 - \frac{1}{\ln m + 1}\right)$-приближенным (аппроксимационным) алгоритмом.

**Доказательство.** Применим технику доказательства реоптимизации Min Vertex Cover (задачи о минимальном вершинном покрытии) при вставке новой вершины из [7]. Построим $REOPT(\Pi(A_j^+, c))$.

Пусть $S^*$ — оптимальное решение задачи $\Pi(A,c)$, $S_{I'}^*$ — оптимальное решение $\Pi(A_j^+, c)$, $S^* \cup \{j\}$ — допустимое решение $\Pi(A_j^+, c)$.

Если $S_{I'}^*$ содержит *j*, то $S^* \cup \{j\}$ — решение.

Предположим, что $S^*_{I'}$ не содержит $j$, тогда $S^* \cup \{j\}$ — допустимое решение $\Pi(A_j^+, c)$ и, поскольку $c(S^*) \le c(S^*_{I'})$, имеем

$$c(S^* \cup \{j\}) \le c(S^*_{I'}) + c_j. \tag{1}$$

Построим допустимое решение $S_1$ задачи $\Pi(A_j^+, c)$:

— применим $\rho$-приближенный (аппроксимационный) алгоритм для задачи $\Pi(A_j^+, c)$ с исключенным столбцом (множеством) $j$ (соответствующее $x_j = 0$ в $\Pi(A, c)$);

— к полученному решению добавим столбец (множество) $j$.

Таким образом, имеем

$$c(S_1) \le \rho(c(S^*_{I'}) - c_j) + c_j = \rho c(S^*_{I'}) - (\rho - 1) c_j \tag{2}$$

(как и выше, $S^*_{I'}$ не содержит $j$). Среди решений $S^* \cup \{j\}$ и $S_1$ выберем ($S$) наилучшее (с наименьшим значением веса). Домножив (1) на $\rho - 1$ и прибавив к (2), получим

$$(\rho - 1) c(S^* \cup \{j\}) + c(S_1) \le (\rho - 1) c(S^*_{I'}) + \rho c(S^*_{I'}) = (2\rho - 1) c(S^*_{I'}).$$

Поскольку $(\rho - 1) c(S^* \cup \{j\}) + c(S_1) \ge (\rho - 1 + 1) \min\{c(S^* \cup \{j\}), c(S_1)\} = \rho c(S)$, имеем

$$\rho c(S) \le (2\rho - 1) c(S^*_{I'}).$$

Получено приближенное решение $S$ задачи $\Pi(A_j^+, c)$, для которого

$$c(S) \le \frac{2\rho - 1}{\rho} c(S^*_{I'}) = \left(2 - \frac{1}{\rho}\right) c(S^*_{I'}).$$

Положив $\rho = \ln m + 1$ (т.е. применив при формировании $S_1$ жадный алгоритм), в результате получим доказательство теоремы.

**Теорема 2.** Для $1 < p < m$ существует $REOPT(\Pi(A_j^+(p), c))$, являющийся $\left(2 - \frac{1}{\ln m + 1}\right)$-приближенным (аппроксимационным) алгоритмом.

**Доказательство** полностью аналогично доказательству теоремы 1.

**Теорема 3.** Существует $REOPT(\Pi(A_j^-, c))$, являющийся $\left(2 - \frac{1}{\ln m + 1}\right)$-приближенным (аппроксимационным) алгоритмом.

**Доказательство** проведем по аналогии с предыдущим (с некоторыми модификациями). Построим соответствующий $REOPT(\Pi(A_j^-, c))$.

Пусть $S^*$ — оптимальное решение задачи $\Pi(A, c)$, $S^*_{I'}$ — оптимальное решение задачи $\Pi(A_j^-, c)$.

Если $S^*$ не содержит $j$, то $S^*_{I'} = S^*$. Если $S^*$ содержит $j$, то $S^*$ может не быть допустимым решением $\Pi(A_j^-, c)$. Построим в этом случае подходящее приближение решения задачи $\Pi(A_j^-, c)$.

Для фиксированного $j$ положим $i_0 = \arg\min\{c_i : a_{ij} = 1\}$ (т.е. $i_0$ такое, $1 \le i \le n$, что $c_{i_0}$ минимально и $a_{ij} = 1$, $1 \le i \le n$). Тогда в любом случае $S^* \cup \{i_0\}$ — допустимое решение $\Pi(A_j^-, c)$. Поскольку $c(S^*) \le c(S^*_{I'})$, имеем

$$c(S^* \cup \{i_0\}) \le c(S^*_{I'}) + c_{i_0}. \tag{3}$$

Построим допустимое решение $S_1$ задачи $\Pi(A_j^-, c)$:

— применим $\rho$-приближенный (аппроксимационный) алгоритм для задачи $\Pi(A_j^-, c)$ с исключенным столбцом (множеством) $i_0$ (соответствующее $x_{i_0} = 0$ в $\Pi(A, c)$);

— к полученному решению добавим столбец (множество) $i_0$.

Таким образом, имеем

$$c(S_1) \le \rho(c(S_{I'}^*) - c_{i_0}) + c_{i_0} = \rho c(S_{I'}^*) - (\rho - 1)c_{i_0}. \tag{4}$$

Среди решений $S^* \cup \{i_0\}$ и $S_1$ выберем $(S)$ наилучшее (с наименьшим значением веса). Домножив (3) на $\rho - 1$ и прибавив к (4), получим

$$(\rho-1)c(S^* \cup \{i_0\}) + c(S_1) \le (\rho-1)c(S_{I'}^*) + \rho c(S_{I'}^*) = (2\rho - 1)c(S_{I'}^*).$$

Поскольку

$$(\rho-1)c(S^* \cup \{i_0\}) + c(S_1) \ge (\rho - 1 + 1)\min\{c(S^* \cup \{i_0\}), c(S_1)\} = \rho c(S),$$

имеем $\rho c(S) \le (2\rho - 1)c(S_{I'}^*)$.

Получено приближенное решение $S$ задачи $\Pi(A_j^-, c)$, для которого

$$c(S) \le \frac{2\rho - 1}{\rho} c(S_{I'}^*) = \left(2 - \frac{1}{\rho}\right) c(S_{I'}^*).$$

Положив $\rho = \ln m + 1$ (т.е. применив при формировании $S_1$ жадный алгоритм), получим доказательство теоремы.

**Теорема 4.** Для $1 < p < m$ существует $REOPT(\Pi(A_j^-(p), c))$, являющийся $\left(2 - \frac{1}{\ln m + 1}\right)$-приближенным (аппроксимационным) алгоритмом.

Доказательство аналогично доказательству теоремы 3.

СПИСОК ЛИТЕРАТУРЫ


1. Ausiello G., Escoffier B., Monnot J., and Paschos V.Th. Reoptimization of minimum and maximum traveling salesman's tours // Algorithmic theory. SWAT 2006; Lect. Notes Comput. Sci. — Berlin: Springer, 2006. — **4059**. — P. 196–207.
2. On the approximability of TSP on local modifications of optimal solved instances / H.J. Bockenhauer, L. Forlizzi, J. Hromkovic, et al. // Algorithmic Oper. Res. — 2007. — **2**(2). — P. 83–93.
3. Bockenhauer H.J., Hromkovic J., Momke T., and Widmayer P. On the hardness of reoptimization // Proc. of the 34th Intern. Conf. on Current Trends in Theory and Practice of Computer Science (SOF — SEM 2008); Lect. Notes Comput. Sci. — Berlin: Springer, 2008. — **4910**. — P. 50–65.
4. Escoffier B., Milanic M., and Paschos V.Th. Simple and fast reoptimizations for the Steiner tree problem // Algorithmic Oper. Res. — 2009. — **4**(2). — P. 86–94.
5. Archetti C., Bertazzi L., and Speranza M.G. Reoptimizing the travelling salesman problem // Networks. — 2003. — **42**(3). — P. 154–159.
6. Archetti C., Bertazzi L., and Speranza M.G. Reoptimizing the 0-1 knapsack problem. — Manuscript, 2008.
7. Ausiello G., Bonifaci V., and Escoffier B. Complexity and approximation in reoptimization // Computability in Context: Computation and Logic in the Real World. — Imperial College Press, members of the 2007 Computability Europe conference CiE 2007: Logic and Computation in the Real World (June, 2007). — P. 24–33.
8. Chvatal V.A. A greedy heuristic for the set covering problem // Math. Oper. Res. — 1979. — **4**, N 3. — P. 233–235.


*Поступила 10.12.2009*