УДК 519.854

В.А. МИХАЙЛЮК

К ВОПРОСУ О СУЩЕСТВОВАНИИ ПОЛИНОМИАЛЬНО ПРИБЛИЖЕННЫХ СХЕМ ДЛЯ РЕОПТИМИЗАЦИИ ДИСКРЕТНЫХ ЗАДАЧ ОПТИМИЗАЦИИ

Ключевые слова: *реоптимизация, r-приближенный алгоритм, полиномиально приближенные схемы.*

Понятие реоптимизации [1–7] состоит в следующем. Пусть П — некоторая NP-трудная (возможно, NP-полная) проблема, I — начальный экземпляр проблемы Π, оптимальное решение которого известно. Предлагается новый экземпляр I' задачи Π, полученный некоторыми «незначительными» изменениями экземпляра I. Возникает ворос: как можно эффективно использовать знания об оптимальном решении I для вычисления точного или приближенного решения экземпляра I'? Цель реоптимизации при использовании приближенных методов — применение знаний о решении начального экземпляра I при условии: либо достижения лучшего качества приближения (аппроксимационного отношения) I'; либо создания более эффективного (по времени) алгоритма определения оптимального или близкого к нему решения I'; либо выполнения первого и второго пунктов.

Исследования по реоптимизации различных задач дискретной оптимизации проведены в работах [1–7].

ПРЕДВАРИТЕЛЬНЫЕ СВЕДЕНИЯ О ПРИБЛИЖЕННЫХ АЛГОРИТМАХ И АППРОКСИМАЦИОННЫХ КЛАССАХ ЗАДАЧ ДИСКРЕТНОЙ ОПТИМИЗАЦИИ

Приведем некоторые необходимые понятия для дальнейшего изложения, введенные в [7].

Определение 1. 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 вычислимая величина, называемая целевой функцией;
- $\mathrm{opt} \in \{\min, \max\}$ — тип оптимизационной проблемы.

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

Определение 2. Для 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 относительно $|I|$, то A называется полиномиальным приближенным алгоритмом для П.

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

Для отдельных проблем из APX можно ввести более сильную форму аппроксимационности. Для любого рационального $r > 1$ (или $r \in (0,1)$ для максимизационных проблем) существует алгоритм A_r и требуемый 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 полиномиально приближенной схемой (Polynomial Time Approximation Scheme — PTAS).

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

Замечание 1. Определение 4 эквивалентно следующему. NPO-проблема П принадлежит классу PTAS, если для любого рационального $\varepsilon > 0$ и любого экземпляра I с П существует такая полиномиально приближенная схема A_ε, что $\rho_{A_\varepsilon}(I) \le 1+\varepsilon$ (соответственно $\rho_{A_\varepsilon}(I) \ge 1-\varepsilon$) для П-минимизационной (П-максимизационной) проблемы.

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

Определение 5. NPO-проблема принадлежит классу FPTAS, если она допускает полностью полиномиальную приближенную схему.

Если $P \ne \mathrm{NP}$, то имеет место включение $\mathrm{FPTAS} \subset \mathrm{PTAS} \subset \mathrm{APX} \subset \mathrm{NPO}$.

Известны следующие данные о существовании приближенных полиномиальных схем для реоптимизации дискретных задач оптимизации [10]. В предположении $P \ne \mathrm{NP}$ не существует FPTAS для задачи о коммивояжере на минимум с неравенством треугольника при увеличении (уменьшении) значения веса единственного ребра. Если $P \ne \mathrm{NP}$, то существует PTAS для задачи о коммивояжере на максимум с неравенством треугольника также при увеличении (уменьшении) значения веса единственного ребра, однако для этой же задачи не существует FPTAS. Не существует FPTAS для задачи «минимальное дерево Штейнера» с неравенством треугольника при уменьшении веса единственного ребра, если $P \ne \mathrm{NP}$.

ПОЛИНОМИАЛЬНО ПРИБЛИЖЕННЫЕ СХЕМЫ ДЛЯ «НАСЛЕДСТВЕННЫХ» ПРОБЛЕМ

Пусть $G = (V, E)$ — граф с множеством вершин V и множеством ребер E. Будем считать некоторое свойство на графах «наследственным» при выполнении следующих условий. Если $G = (V, E)$ удовлетворяет такому свойству, то для любого $V' \subseteq V$ подграф $G[V']$, индуцированный V', удовлетворяет этому же свойству, например, независимость, двудольность, планарность — три «наследственных» свойства.

Определение 6. Пусть $G=(V,E,w)$ — взвешенный вершинный граф. Назовем Hered (наследство) класс проблем нахождения для графа $G=(V,E)$ подмножества вершин S таких, что:

а) $G[S]$ удовлетворяет «наследственному» свойству;

б) значение $w(S)=\sum_{v\in S} w(v)$ максимально.

Например, Max Weighted Independent Set (максимальное взвешенное независимое множество), Max Weighted Bipartite Subgraph (максимальный взвешенный двудольный подграф), Max Weighted Planar Subgraph (максимальный взвешенный планарный подграф) — три известные проблемы Hered, которые соответствуют трем «наследственным» свойствам.

Теорема 1 [7]**.** Пусть Π — проблема из Hered. При вставке вершины реоптимизация Π аппроксимируема с отношением $1/2$ (в константное время).

Рассмотрим невзвешенные проблемы, т.е. когда все вершины имеют вес единица. Пусть I — начальный экземпляр Π, I' — окончательный экземпляр Π (вставляется новая вершина v), S^* — оптимальное решение на I, $S^*_{I'}$ — оптимальное решение на I'. Если вставляется только одна вершина, начальное оптимальное решение имеет абсолютную ошибку, не большую единицы в окончательном экземпляре, т.е. имеет место

$$|S^*|\geq|S^*_{I'}|-1\,.$$

Справедлива следующая теорема.

Теорема 2 [7]**.** При вставке новой вершины реоптимизация любой невзвешенной проблемы из Hered предполагает существование PTAS.

Доказательство. Пусть $\varepsilon>0$ фиксировано, положим $k=[1/\varepsilon]$. Рассмотрим следующий алгоритм.

1. Протестировать все подмножества V размера не больше k и пусть S_1 — наибольшее из них такое, что $G[S_1]$ удовлетворяет «наследственному» свойству.

2. Вывести такое решение S из S_1 и S^*, которое имеет большее число вершин. Тогда, если $S^*_{I'}$ имеет размер, не больший $1/\varepsilon$, найдем его на шаге 1. Иначе, $|S^*_{I'}|\geq\dfrac{1}{\varepsilon}$ и $\dfrac{|S^*|}{|S^*_{I'}|}\geq\dfrac{|S^*_{I'}|-1}{|S^*_{I'}|}\geq 1-\varepsilon$.

Алгоритм полиномиален при $\varepsilon=\text{const}$.

В данном случае существование PTAS следует из двух свойств: абсолютная ошибка не больше единицы; простота рассматриваемых проблем. Проблема называется простой [11], если для фиксированной константы k определить, что оптимальное решение имеет значение не больше k (максимизация), можно за полиномиальное время. Этот результат корректен для Min Vertex Cover (минимальное вершинное покрытие) и не правомерен для Minimum Edge Coloring (минимальная реберная раскраска).

ПОЛИНОМИАЛЬНО ПРИБЛИЖЕННЫЕ СХЕМЫ И РЕОПТИМИЗАЦИЯ NPO-ПРОБЛЕМ

Представляют интерес вопросы доказательства несуществования PTAS для NPO-проблем (NPO-проблемы — оптимизационные версии NP-полных проблем распознавания свойств) в предположении $P\neq$ NP. Наиболее простой техникой таких доказательств является техника «щели» (gap technique), впервые использованная в середине 1970-х годов (например, в [12]). Ее идея состоит в доказательстве NP-трудности с помощью полиномиальных сводимостей, что способствует образованию «щели» между целевыми значениями нет-экземпляров и целевыми значениями да-экземпляров. Наиболее полно это будет выражено при установлении критерия несуществования PTAS для реоптимизации NPO-проблем.

Пусть Π — NPO-проблема и согласно определению 1 $I \in E$ (I — экземпляр Π), $S \in \mathrm{Sol}(I)$ — допустимое решение, $m(I, S)$ — значение (числовое) целевой функции, opt = min (для определенности), $S^*(I)$ — оптимальное решение Π с I, $\mathrm{opt}(I)$ — числовое значение $m(I, S^*(I))$.

Пусть I' — некоторое «незначительное» изменение экземпляра I.

Реоптимизационную проблему, соответствующую I', обозначим $\Pi(I', m(I, S(I)), S^*(I)) = \Pi(I', I)$. Реоптимизационный алгоритм для $\Pi(I', I)$ состоит в решении задачи Π с I' и целевой функцией $m(I, S(I))$ с использованием оптимального решения I ($S^*(I)$). Допустим, что для минимизационной NPO-проблемы Π все допустимые решения всех экземпляров имеют целые неотрицательные значения целевой функции.

Теорема 3. Пусть X — NP-трудная проблема распознавания свойств, $\Pi(I', I)$ — минимизационная реоптимизационная проблема и T — полиномиально вычислимое преобразование множества экземпляров X во множество экземпляров $\Pi(I', I)$ такое ($a < b$ — фиксированные неотрицательные целые), что:

1) каждый да-экземпляр X отображается на экземпляр $\Pi(I', I)$ со значением целевой функции, не большим a;

2) каждый нет-экземпляр X отображается на экземпляр $\Pi(I', I)$ со значением целевой функции, не меньшим b.

Тогда для реоптимизации $\Pi(I', I)$ не существует полиномиального ρ-приближенного алгоритма с коэффициентом $\rho < \frac{b}{a}$, если $P \neq$ NP. Это означает, что для реоптимизации $\Pi(I', I)$ не существует PTAS, если $P \neq$ NP.

Доказательство. Допустим, что существует полиномиальный ρ-приближенный алгоритм с коэффициентом $\rho < \frac{b}{a}$. Пусть I — произвольный экземпляр проблемы X; применим полиномиальное преобразование T. Для экземпляра $I' = T(I)$ используем полиномиальный приближенный алгоритм (для проблемы $\Pi(I', I)$). Если I является да-экземпляром, то значение функции цели $T(I)$ не больше a и приближенный алгоритм даст решение со значением, строго меньшим, чем $\left(\frac{b}{a}\right)a = b$. Если I — нет-экземпляр, то значение функции цели $T(I)$ не меньше b и приближенный алгоритм не может дать решение со значением, меньшим оптимального b. Таким образом, можно различить за полиномиальное время да-экземпляры (приближенное значение $T(I) < b$) и нет-экземпляры (приближенное значение $T(I) \geq b$) NP-трудной проблемы X. Это означает существование полиномиального алгоритма для распознавания NP-трудной задачи, следовательно, $P =$ NP.

ЗАДАЧА О ПОКРЫТИИ МНОЖЕСТВАМИ

Рассмотрим NPO-задачу о покрытии множествами в следующей постановке (задача $\Pi(A, c)$):

$$\min\left\{ f(x) = \sum_{i=1}^{n} c_i x_i \mid x \in Q(A) \right\}, \qquad Q(A) = \left\{ x \in B^n \mid \sum_{j=1}^{n} a_{ij} x_j \geq 1,\ i = 1, \ldots, m \right\}.$$

Здесь $B^n = \{0,1\}^n$; $A = \{a_{ij}\}$, $i = 1, \dots, m$, $j = 1, \dots, n$; $A - (m, n)$ — булева матрица $(m \le n)$; $c_i > 0$, $i = 1, \dots, n$.

Матрица A не содержит нулевых строк и строк с одной единицей (условие **A**).

Если $c_i > 0$ — целые, $i = 1, \dots, n$, то задачу $\Pi(A, c)$ назовем взвешенной; если $c_i = 1$, $i = 1, \dots, n$, то задачу $\Pi(A, 1)$ назовем невзвешенной (1 — n-мерный вектор, состоящий из единиц).

Будем пользоваться следующими результатами.

Известно [8], что жадный алгоритм решает задачу $\Pi(A, c)$ с оценкой точности

$$\sum_{t=1}^{c(A)} \frac{1}{t} \le \ln m + 1,$$

где $c(A)$ — максимальное число единиц в столбце матрицы A, т.е. жадный алгоритм — это $(\ln m + 1)$-приближенный (аппроксимационный) алгоритм для решения $\Pi(A, c)$.

В [9] показано, что если $\mathrm{NP} \not\subset \mathrm{TIME}(n^{O(\log\log n)})$, то для произвольного $\varepsilon > 0$ не существует полиномиального алгоритма, который может аппроксимировать задачу о покрытии множествами с коэффициентом $(1 - \varepsilon) \ln m$.

Введем следующие обозначения для измененных экземпляров задачи $\Pi(A, c)$: $\Pi(A_j^+, c, A)$ (соответственно $\Pi(A_j^-, c, A)$) обозначим задачу $\Pi(A, c)$ с заменой в столбце j матрицы A произвольного 0 на 1 (1 на 0); $\Pi(A_j^+, 1, A)$ (соответственно $\Pi(A_j^-, 1, A)$) обозначим $\Pi(A_j^+, A)$ (соответственно $\Pi(A_j^-, A)$).

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

Рассмотрим NP-полную (а значит, NP-трудную) задачу распознавания свойств «Минимальное покрытие» (МП) [13], которая соответствует невзвешенной NPO-задаче о покрытии множествами.

Условие. Задано семейство E ($|E| = n$) подмножеств множества S ($|S| = m$) и положительное целое число a.

Вопрос. Содержит ли E покрытие множества S размера не более a, т.е. найдется ли подмножество $E' \subseteq E$ такое, что $|E'| \le a$ и $\bigcup_{e \in E'} e = S$?

Теорема 4. Если $P \ne \mathrm{NP}$, то для реоптимизации задачи $\Pi(A_j^+, A)$ не существует PTAS.

Доказательство. Применим теорему 3. В качестве X — NP-трудной проблемы распознавания свойств — рассмотрим задачу МП, в качестве $\Pi(I', I)$ — задачу $\Pi(A_j^+, A)$. Пусть $T(A)$ — преобразование матрицы A, состоящее в замене произвольного элемента 0 в столбце j на 1, т.е. $A_j^+ = T(A)$. Ясно, что T — полиномиально вычислимое преобразование экземпляров X во множество экземпляров $\Pi(A_j^+, A)$. Положим $b = a + 1$ и покажем, что выполняются условия 1, 2 теоремы 3.

Пусть $E' = \{E_{j_1}, E_{j_2}, \dots, E_{j_k}\}$ — покрытие множества S, что соответствует да-экземпляру задачи X из п. 1 теоремы 3 ($|E'| = k \le a$). Этот экземпляр отображается на допустимое решение $x = (x_1, \dots, x_n)$ задачи $\Pi(A_j^+, A)$ такое, что $x_{j_1} = x_{j_2} = \dots \dots = x_{j_k} = 1$ (допустимое решение в силу преобразования T) со значением целевой функции $\sum_{j \in \{j_1, \dots, j_k\}} c_j = k \le a$ ($c_i = 1$, $i = 1, \dots, n$), и п. 1 теоремы 3 выполнен.

Пусть $E' = \{E_{j_1}, E_{j_2}, \dots, E_{j_k}\}$ — покрытие множества S, что соответствует нет-экземпляру задачи X из п. 2 ($|E'| = k \ge a+1$). Этот экземпляр отображается на допустимое решение $x = (x_1, \dots, x_n)$ задачи $\Pi(A_j^+, A)$ такое, что $x_{j_1} = x_{j_2} = \dots = x_{j_k} = 1$ (допустимое решение в силу преобразования T) со значением целевой функции $\sum_{j \in \{j_1, \dots, j_k\}} c_j = k \ge a+1$, и п. 2 выполнен.

Таким образом, условия теоремы 3 выполнены. Следовательно, для реоптимизации $\Pi(A_j^+, A)$ не существует полиномиального ρ-приближенного алгоритма с коэффициентом $\rho < \frac{a+1}{a} = 1 + \frac{1}{a}$, если $P \ne \mathrm{NP}$. Это означает, что для реоптимизации $\Pi(A_j^+, A)$ не существует PTAS, если $P \ne \mathrm{NP}$.

Теорема доказана.

Теорема 5. Если $P \ne \mathrm{NP}$, то для реоптимизации задачи $\Pi(A_j^-, A)$ не существует PTAS.

Доказательство. Применим теорему 3. В качестве X — NP-трудной проблемы распознавания свойств — рассмотрим задачу МП, в качестве $\Pi(I', I)$ — задачу $\Pi(A_j^-, A)$. Пусть $T(A)$ — преобразование матрицы A, состоящее в замене произвольного элемента 1 в столбце j на 0, т.е. $A_j^- = T(A)$. Очевидно, что T — полиномиально вычислимое преобразование экземпляров X во множество экземпляров $\Pi(A_j^-, A)$. Положим $b = a+1$ и покажем, что выполняются условия 1 и 2 теоремы 3.

Пусть $E' = \{E_{j_1}, E_{j_2}, \dots, E_{j_k}\}$ — покрытие множества S, что соответствует да-экземпляру задачи X из п. 1 теоремы 3 ($|E'| = k \le a$). Если $x = (x_1, \dots, x_n)$ такое, что $x_{j_1} = x_{j_2} = \dots = x_{j_k} = 1$ — допустимое решение задачи $\Pi(A_j^-, A)$, то действуем, как при доказательстве теоремы 4. Иначе, пусть элемент 1 матрицы A, который заменяется на 0, находится в строке $i_1 \in \{1, \dots, m\}$ и столбце $j_s \in \{j_1, \dots, j_k\}$. Таким образом, решение $x = (x_1, \dots, x_n)$, такое, что $x_{j_1} = x_{j_2} = \dots = x_{j_k} = 1$, не покрывает строку i_1. Тогда в силу условия **A** существует такое j_{s_1}, что $a_{i_1 j_{s_1}} = 1$. В этом случае $E' = \{\bigcup E_j : j \in \{j_1, \dots, j_k\} \setminus j_s\} \bigcup \bigcup \{E_{j_{s_1}}\}$ составляет покрытие $A_j^- = T(A)$, причем $|E'| = k \le a$. Построим покрытие, соответствующее $\Pi(A_j^-, A)$. В силу условия **A** существует такое j_{s_1}, что элемент матрицы $a_{i_1 j_{s_1}} = 1$. Тогда $x = (x_1, \dots, x_n)$, такое, что $x_j = 1$ при $j \in (\{j_1, \dots \dots, j_k\} \setminus j_s) \bigcup \{j_{s_1}\} = J$, является покрытием $\Pi(A_j^-, A)$ таким, что $\sum_{j \in J} c_j = k \le a$, и п. 1 выполнен.

Для доказательства выполнения п. 2 действуем аналогично. При этом нет-экземпляр задачи X ($|E'| \geq a+1$) отображается на допустимое решение $\Pi(A_j^-, A)$ со значением целевой функции, не меньшим $a+1$. Таким образом, все условия теоремы 3 выполнены. Следовательно, для реоптимизации $\Pi(A_j^-, A)$ не существует полиномиального ρ-приближенного алгоритма с коэффициентом $\rho < \frac{a+1}{a} = 1 + \frac{1}{a}$, если $P \neq \mathrm{NP}$. Это означает, что для реоптимизации $\Pi(A_j^-, A)$ не существует PTAS, если $P \neq \mathrm{NP}$.

Теорема доказана.

Замечание 2. Аналогичное доказательство корректно и для любой взвешенной задачи о покрытии множествами, однако «щель» будет более широкой $(b > a+1)$.

ЗАДАЧА «РАСКРАШИВАЕМОСТЬ ГРАФА»

Рассмотрим задачу «Раскрашиваемость графа» (РГ) [13].

Условие. Заданы граф $G = (V, E)$ и положительное целое число $K (K \leq |V|)$.

Вопрос. Верно ли, что граф G K-раскрашиваем? (Граф называется K-раскрашиваемым, если существует некая функция $f : V \to \{1, 2, \dots, K\}$ такая, что если $\{u, v\} \in E$, то $f(u) \neq f(v)$.)

Задача РГ — NP-полна (а значит, NP-трудна) для $K \geq 3$ [13].

С задачей РГ сопоставим NPO-задачу «минимальная раскрашиваемость графа» (Min РГ) — нахождение минимального K такого, что граф $G = (V, E)$ K-раскрашиваем. Задачу Min РГ запишем в виде $G = (V, E, w)$, где w — целевая функция (число красок), $w \to \min$. Рассмотрим следующую реоптимизационную задачу. Пусть граф $G' = (V', E', w)$ получен из графа $G = (V, E, w)$ вставкой произвольной вершины с не более чем двумя ребрами, инцидентными ей. Задача реоптимизации G' состоит в нахождении ее решения (приближенного), исходя из оптимального решения G со значением $w_{\min}$.

Теорема 6. Если $P \neq \mathrm{NP}$, то для реоптимизации задачи $G' = (V', E', w)$ не существует PTAS.

Доказательство. Применим теорему 3. В качестве X — NP-трудной проблемы распознавания свойств — рассмотрим задачу РГ с $K = 3$, в качестве $\Pi(I', I)$ — задачу G'. Пусть T — преобразование, сопоставляющее множествам V, E соответственно множества V', E' и решению G (допустимому) — решение G' (допустимое), которое тождественно совпадает с решением G; T — полиномиально вычислимое преобразование. Покажем выполнение условий 1, 2 теоремы 3 (положим $a = 3$, $b = 4$).

Да-экземпляр задачи $G = (V, E)$ — это 3-раскраска графа G. Докажем, что она будет 3-раскраской графа G'. Действительно, исходя из 3-раскраски G, можно раскрасить вершину v графа G' (добавленную, с не более чем двумя инцидентными ей ребрами) цветом, отличным от двух цветов (если добавлены два ребра), которыми раскрашены в G соответствующие вершины. Таким образом, получили 3-раскраску графа G', т.е. экземпляр $G' = (V', E', w)$ со значением $w \leq a = 3$, и п. 1 выполнен.

Нет-экземпляр задачи X — это раскраска графа G не менее четырьмя цветами, что согласно п. 1 дает возможность раскрасить G' также не менее четырьмя цветами, т.е. получили экземпляр $G' = (V', E', w)$ со значением $w \geq 4$, и п. 2 выполнен. Тогда согласно теореме 3 для реоптимизации $G' = (V', E', w)$ не существует полиномиального ρ-приближенного алгоритма с коэффициентом $\rho < \frac{b}{a} = \frac{4}{3}$, если $P \neq \mathrm{NP}$. Это означает, что для реоптимизации $G' = (V', E', w)$ не существует PTAS, если $P \neq \mathrm{NP}$.

Пусть $G_1 = (V_1, E_1, w)$ получается из графа $G = (V, E, w)$ удалением произвольной вершины со всеми инцидентными ей ребрами.

Теорема 7. Если $P \neq \mathrm{NP}$, то для реоптимизации задачи $G_1 = (V_1, E_1, w)$ не существует PTAS.

Доказательство аналогично предыдущему и следует из того факта, что правильная раскраска графа — «наследственное» свойство, а именно, из 3-раскраски графа $G = (V, E, w)$ вытекает 3-раскраска графа $G_1 = (V_1, E_1, w)$ и из не менее чем 4-раскраски графа G — не менее чем 4-раскраска графа G_1.

ЗАДАЧА «УПАКОВКА В КОНТЕЙНЕРЫ»

Рассмотрим задачу «Упаковка в контейнеры» (УК) [13].

Условие. Заданы конечное множество U предметов, размер $s(u) \in Z^+$ каждого предмета $u \in U$, положительное целое число B — вместимость контейнера, положительное целое число K.

Вопрос. Существует ли такое разбиение множества U на непересекающиеся подмножества $U_1, U_2, \ldots, U_K$, что сумма размеров предметов из каждого подмножества U_i не превосходит B?

При $K \geq 2$ эта задача NP-полна [13]. NPO-версия задачи УК (Minimum Bin Packing, Min УК) состоит в минимизации K задачи УК (обозначается Min_УК(U, s, B, K), $K \to \min$). Рассмотрим задачу Min_УК (U', s, B, K), которая получается из задачи Min_УК (U, s, B, K) удалением произвольного предмета $u \in U$.

Теорема 8. Если $P \neq \mathrm{NP}$, то для реоптимизации Min_УК(U', s, B, K) не существует PTAS.

Доказательство. Применим теорему 3. В качестве X — NP-трудной проблемы распознавания свойств — рассмотрим задачу УК с $K = 2$, в качестве $\Pi(I', I)$ — задачу Min_УК(U', s, B, K). Пусть T — полиномиально вычислимое преобразование, которое сопоставляет множеству U множество U', допустимому решению U — допустимое решение U' из пп. 1 и 2 теоремы 3 следующим образом (положим $a = 2$, $b = 3$).

Да-экземпляру задачи УК соответствует разбиение U на два непересекающихся подмножества — U_1, U_2 (таких, что сумма размеров предметов из каждого подмножества не превосходит B). Это соответствует разбиению U' из Min УК также на два подмножества — U_1, U_2 (элемент u удаляется либо из U_1, либо из U_2) со значением целевой функции $K \leq 2$, и п. 1 выполнен.

Нет-экземпляру задачи УК соответствует разбиение U на не менее чем три непересекающихся подмножества (таких, что сумма размеров предметов из каждого подмножества не превосходит B). Это соответствует разбиению U' из Min УК также на не менее чем три непересекающихся подмножества (путем удаления из одного из них элемента $u \in U$) со значением целевой функции $K \geq 3$, и п. 2 выполнен.

Тогда согласно теореме 3 для реоптимизации Min_УК (U', s, B, K) не существует полиномиального ρ-приближенного алгоритма с коэффициентом $\rho < \frac{b}{a} = \frac{3}{2}$, если $P \neq \mathrm{NP}$. Это означает, что для реоптимизации Min_УК (U', s, B, K) не существует PTAS, если $P \neq \mathrm{NP}$.

Замечание 3. Легко видеть, что теорема, аналогичная теореме 8, имеет место для реоптимизации Min_УК (U, s', B, K), где s' получается с помощью s уменьшением на некоторое положительное целое число для некоторых элементов $u \in U$.

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 34 th 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 // Chapter of the book 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.
9. Feige U. A threshold of lnn for approximating set cover // J. ACM. — 1998. — **45**, N 4. — P. 634–652.
10. Berg T. and Hempel H. Reoptimization of TSP and Steiner tree: Changing single edge-weights // Lect. Notes Comput. Sci. — 2009. — **5457**. — P. 141–151.
11. Paz A., and Moran S. Non deterministic polynomial optimization problems and their approximations // Theoret. Comput. Sci. — 1981. — **15**. — P. 251–277.
12. Lenstra J.K., and Renooy Kan A.H.G. Complexity of scheduling under precedence constraints // Oper. Res. — 1978. — **26**. — P. 22–35.
13. Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи. — М.: Мир, 1982. — 416 с.

Поступила 12.08.2010