ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Сентябрь—октябрь 2014. Том 21, № 5. С. 54–66

УДК 519.87+519.854

О СЛОЖНОСТИ ДВУХУРОВНЕВЫХ ЗАДАЧ РАЗМЕЩЕНИЯ И ЦЕНООБРАЗОВАНИЯ *)

А. А. Панин, А. В. Плясунов

Аннотация. Рассматриваются двухуровневые частично-целочисленные задачи размещения и ценообразования. Каждая из них определяется оптимизационными задачами верхнего и нижнего уровней, первая из которых описывает выбор размещения и ценообразования, а вторая моделирует реакцию потребителей на решение верхнего уровня. Основные усилия в работе концентрируются вокруг изучения вычислительной сложности двухуровневых задач с разными стратегиями ценообразования: равномерной, фабричной и дискриминационной. Показано, что при любой стратегии ценообразования соответствующая оптимизационная задача NP-трудна в сильном смысле, принадлежит классу Poly-APX и является полной в нём относительно AP-сводимости.

Ключевые слова: двухуровневая задача, размещение, ценообразование, вычислительная и аппроксимационная сложности, NP-трудность в сильном смысле, AP-сводимость, Poly-APX-полнота.

Введение

Процессы размещения производства и ценообразования обычно исследуются отдельно и независимо друг от друга [3, 9, 13, 14, 17]. Основная причина этого связана с тем, что они относятся к разным горизонтам планирования. Процессы размещения являются долговременными, а процессы ценообразования относятся к краткосрочному планированию. Как следствие, в большинстве случаев сначала выбирается размещение, а затем цены. Однако ещё в [15] отмечалось, что разделение размещения и ценообразования неприемлемо из-за того, что размещение предприятий (магазинов) должно проводиться с учётом имеющегося спроса, который так или иначе зависит от цен. С другой стороны, выбор наилучших цен зависит от того, где расположены пункты производства. Таким образом, разделение размещения и ценообразования при моделировании приводит

*) Исследование выполнено при финансовой поддержке Российского фонда фундаментальных исследований (проекты 13–06–00023 и 13–07–00016).

к невозможности получить наилучшие варианты размещения и цен. Более того, даже в ситуации, когда нет необходимости знать точные цены на производимый продукт, а достаточно знать лишь диапазон цен в выбранной нише рынка, позволяющих противостоять конкурентам, разделение размещения и ценообразования также нецелесообразно [6]. Поэтому современные подходы к выбору эффективного механизма взаимодействия процессов размещения производства и ценообразования основываются на их совместном анализе в рамках одной модели [5, 6, 11, 16]. Однако для оценки качества принимаемых решений необходимо уметь адекватно оценивать и реакцию рынка, в частности, потребителей, на предлагаемый вариант размещения и ценообразования. С этой целью удобно моделировать весь процесс в виде задачи двухуровневого программирования [2, 12].

Статья организована следующим образом: в разд. 1 приводятся математические постановки исследуемых задач, в разд. 2 излагаются полученные результаты.

1. Постановки задач

Рассмотрим следующую игру Штакельберга. В игре участвуют два типа игроков: лидер (производитель) и последователи (потребители). Первым делает ход производитель. Он открывает (размещает) предприятия по производству однородной продукции и назначает цену на производимый товар. Затем каждый потребитель выбирает такое открытое предприятие, на котором его суммарные затраты на покупку и транспортировку товара минимальны, и совершает покупку в том случае, если эти затраты не превышают его бюджет. Цель игры: открыть такие предприятия и установить такие цены, при которых доход производителя (сумма цен на товары, купленные потребителями) за вычетом затрат на открытие предприятий максимален.

Далее будем рассматривать лишь кооперативный вариант вышеописанной игры. Предположим, что если затраты какого-нибудь из потребителей минимальны в нескольких предприятиях, то он выберет то предприятие, которое ближе к нему, т. е. с точки зрения производителя будет выбрано предприятие с наибольшей ценой товара. Содержательно это означает, что потребители всегда делают такой выбор, который сохраняет прибыль производителя.

Ограничимся рассмотрением только трёх следующих стратегий ценообразования [15]: фабричное ценообразование (mill pricing), равномерное ценообразование (uniform pricing)и дискриминационное ценообразование (discriminator pricing).

Когда используется первая стратегия ценообразования, на каждом открытом предприятии устанавливается своя цена. При равномерном ценообразовании на всех открытых предприятиях устанавливается одна и та же цена. Дискриминационное ценообразование — стратегия, при которой могут быть ущемлены интересы каких-то групп потребителей, т. е. на каждом открытом предприятии могут устанавливаться разные цены для разных потребителей.

Введём следующие обозначения:

$I = \{1, \ldots, n\}$ — множество возможных мест открытия предприятий;

$J = \{1, \ldots, m\}$ — множество потребителей;

$b_j \geqslant 0$ — ценовой порог (бюджет) j-го потребителя;

$c_{ij} \geqslant 0$ — матрица транспортных затрат потребителей;

$f_i \geqslant 0$ — стоимость открытия предприятия в месте $i \in I$ (i-го предприятия);

$$y_i = \begin{cases} 1, & \text{если } i\text{-е предприятие открыто,} \\ 0 & \text{в противном случае;} \end{cases}$$

$$x_{ij} = \begin{cases} 1, & \text{если } i\text{-е предприятие обслуживает } j\text{-го потребителя,} \\ 0 & \text{в противном случае.} \end{cases}$$

Эти обозначения будут использоваться в постановках всех трёх исследуемых задач: LDP (location and discriminator pricing problem), LMP (location and mill pricing problem) и LUP (location and uniform pricing problem), которые соответствуют всем трём вышеописанным стратегиям ценообразования. В дополнение введём следующие обозначения, отличные для каждой из постановок:

$p_{ij} \geqslant 0$ — цена товара на i-м предприятии для j-го потребителя;

$p_i \geqslant 0$ — цена товара на i-м предприятии (для всех потребителей одинакова);

$p \geqslant 0$ — цена товара (на всех предприятиях и для всех потребителей одинакова).

Используя данные обозначения, запишем игру Штакельберга для случая дискриминационного ценообразования в виде следующей задачи двухуровневого квадратичного программирования (LDP):

$$\sum_{i \in I} \sum_{j \in J} p_{ij} x_{ij} - \sum_{i \in I} f_i y_i \to \max_{p,x,y},$$

$$p_{ij} \geqslant 0, \quad y_i \in \{0, 1\}, \quad i \in I, \ j \in J,$$

где вектор x — оптимальное решение задачи нижнего уровня:

$$\sum_{i\in I}\sum_{j\in J}(b_j - c_{ij} - p_{ij})x_{ij} \to \max_x,$$

$$\sum_{i\in I} x_{ij} \leqslant 1, \quad j \in J,$$

$$x_{ij} \leqslant y_i, \quad i \in I,\ j \in J,$$

$$x_{ij} \in \{0,1\}, \quad i \in I,\ j \in J.$$

Целевая функция задачи определяет доход производителя. Целевая функция нижнего уровня выражает величину сэкономленного потребителями бюджета, а ограничения гарантируют, что каждый потребитель обслуживается не более чем одним открытым предприятием производителя. Также из этих ограничений и определения целевой функции следует, что покупка совершается в том случае, когда это позволяет бюджет потребителя.

Опишем на основе задачи LDP задачи LMP и LUP. Назовём *задачей размещения и фабричного ценообразования* LMP задачу LDP, в которой вместо переменной p_{ij} используется переменная p_i, а *задачей размещения и равномерного ценообразования* LUP будем называть задачу LDP, в которой вместо переменной p_{ij} используется переменная p.

Введённое выше условие кооперативности позволяет говорить об оптимальном решении задач LDP, LMP и LUP. Каждый потребитель выбирает в случае равенства затрат самое близкое предприятие относительно матрицы транспортных затрат, что позволяет производителю явным образом вычислить свой доход.

Перепишем двухуровневую задачу в виде следующей задачи квадратичного программирования со смешанными переменными:

$$\sum_{i\in I}\sum_{j\in J} p_{ij}x_{ij} - \sum_{i\in I} f_i y_i \to \max_{p,x,y}, \tag{1}$$

$$\sum_{i\in I}(b_j - c_{ij} - p_{ij})x_{ij} \geqslant 0, \quad j \in J, \tag{2}$$

$$\sum_{i\in I}(c_{ij} + p_{ij})x_{ij} \leqslant c_{kj} + p_{kj}, \quad k \in I,\ j \in J, \tag{3}$$

$$\sum_{i\in I} x_{ij} \leqslant 1, \quad j \in J, \tag{4}$$

$$x_{ij} \leqslant y_i, \quad i \in I, \ j \in J, \tag{5}$$

$$p_{ij} \geqslant 0, \quad x_{ij}, y_i \in \{0,1\}, \quad i \in I, \ j \in J. \tag{6}$$

Целевая функция (1) определяет доход производителя. Ограничения (2) гарантируют, что потребитель не выйдет за рамки своего бюджета. Выполнение ограничений (3) приводит к тому, что транспортные затраты потребителя и его затраты на приобретение продукции в сумме минимальны. Ограничения (4) означают, что каждый потребитель может быть обслужен не более чем в одном предприятии. Из ограничений (5) следует, что потребитель может обслуживаться только открытым предприятием.

Сохраним для данной переформулировки двухуровневой задачи то же обозначение LDP. Аналогично получим эквивалентные одноуровневые представления для двухуровневых задач размещения с равномерным и фабричным ценообразованием. Также будем использовать обозначения LMP и LUP для соответствующих одноуровневых переформулировок двухуровневых задач.

В дальнейшем предполагаем, что все исходные данные f_i, b_j и c_{ij} являются рациональными числами.

2. Вычислительная сложность

При исследовании сложности нахождения оптимального или даже допустимого решения оптимизационной задачи одним из ключевых вопросов является связь данной задачи с полиномиальной иерархией задач распознавания. Обычно ограничиваются рассмотрением только нулевого и первого уровня иерархии, а именно классов P, NP и co-NP. Относительно исследуемых в этой статье задач имеет место

Теорема 1. *Задачи* LDP, LMP *и* LUP NP-*трудны в сильном смысле.*

Доказательство. Рассмотрим NP-трудную в сильном смысле задачу о минимальном покрытии [1]. Пусть заданы множество $M = \{1, \ldots, m\}$ и набор его подмножеств $M_1, \ldots, M_n$ таких, что $\bigcup\limits_{i \in N} M_i = M$, где $N = \{1, \ldots, n\}$. Совокупность $\widetilde{N} \subset N$ подмножеств M_i, $i \in \widetilde{N}$, называется *покрытием множества* M, если $\bigcup\limits_{i \in \widetilde{N}} M_i = M$. Каждому M_i приписан единичный вес. Требуется найти покрытие минимального веса.

Сведём задачу о минимальном покрытии к задачам LDP, LMP и LUP, т. е. построим функции g и h такие, что функция g строит по входу t задачи о минимальном покрытии вход v задачи LDP (LMP, LUP), длина которого ограничена полиномом от длины $|t|$ входа t, за полиномиальное от $|t|$ время, а функция h строит по оптимальному решению задачи LDP (LMP, LUP) для входа v оптимальное решение задачи t за полиномиальное от $|t|$ время.

Сперва построим по произвольному входу задачи о минимальном покрытии вход задачи LDP (LMP, LUP), т. е. определим функцию g. Пусть $I := N$ — множество возможных мест открытия предприятий, $J := M$ — множество потребителей. Определим бюджет каждого потребителя $b_j := 2,\ j \in J$. Положим $c_{ij} := 1,\ i \in I,\ j \in M_i$. Иначе транспортные затраты положим равными 2, т. е. сделаем оставшиеся транспортные пути закрытыми. Пусть $f_i := 1,\ i \in I$. Введём обозначение: $\widetilde{I} := \{i \in I\ :\ y_i = 1\}$. Докажем следующее вспомогательное утверждение.

Лемма 1. *Задачи* LDP, LMP *и* LUP *для входа* $g(t)$ *эквивалентны задаче* $|J| - |\widetilde{I}| \to \max\limits_{\widetilde{I} \subseteq I}$ *при ограничении* $\bigcup\limits_{i \in \widetilde{I}} M_i = J$.

Доказательство. Сначала покажем, что существуют оптимальные решения $(p^{\mathrm{LDP}}, x^{\mathrm{LDP}}, y^{\mathrm{LDP}})$, $(p^{\mathrm{LMP}}, x^{\mathrm{LMP}}, y^{\mathrm{LMP}})$ и $(p^{\mathrm{LUP}}, x^{\mathrm{LUP}}, y^{\mathrm{LUP}})$ задач LDP, LMP и LUP для входа $g(t)$, в которых $p_{ij}^{\mathrm{LDP}} = p_i^{\mathrm{LMP}} = p^{\mathrm{LUP}} = 1$, $i \in I,\ j \in J$. Рассмотрим произвольное оптимальное решение $(p^{\mathrm{LDP}}, x^{\mathrm{LDP}}, y^{\mathrm{LDP}})$ задачи LDP с входом $g(t)$ и набор $(\widetilde{p}^{\mathrm{LDP}}, x^{\mathrm{LDP}}, y^{\mathrm{LDP}})$, где $\widetilde{p}_{ij}^{\mathrm{LDP}} = 1$, $i \in I,\ j \in J$. Проверив все ограничения, нетрудно убедиться, что этот набор — допустимое решение. Из ограничений задачи LDP следует, что если $x_{ij}^{\mathrm{LDP}} = 1$, то $p_{ij}^{\mathrm{LDP}} \leqslant 1$. Получаем

$$\begin{aligned} w_{\mathrm{LDP}}(p^{\mathrm{LDP}}, x^{\mathrm{LDP}}, y^{\mathrm{LDP}}) &= \sum_{i \in I} \sum_{j \in J} p_{ij}^{\mathrm{LDP}} x_{ij}^{\mathrm{LDP}} - \sum_{i \in I} y_i^{\mathrm{LDP}} \\ &\leqslant \sum_{i \in I} \sum_{j \in J} x_{ij}^{\mathrm{LDP}} - \sum_{i \in I} y_i^{\mathrm{LDP}} = w_{\mathrm{LDP}}(\widetilde{p}^{\mathrm{LDP}}, x^{\mathrm{LDP}}, y^{\mathrm{LDP}}), \end{aligned}$$

где w_{LDP} — целевая функция (1). Отсюда $(\widetilde{p}^{\mathrm{LDP}}, x^{\mathrm{LDP}}, y^{\mathrm{LDP}})$ — оптимальное решение. Для задач LMP и LUP аналогично.

Получается, что для нахождения оптимального решения задач LDP, LMP и LUP для входа $g(t)$ можно ограничиться перебором допустимых решений вида $(1, \ldots, 1, x, y)$. При этом, если некоторый потребитель $j \in M_i$ не обслуживается, назначаем этого потребителя на предприятие i. Если это предприятие закрыто, то открываем его. Тем самым значение целевой функции не уменьшается. Таким образом данные частные случаи эквивалентны задаче: $|J| - |\widetilde{I}| \to \max\limits_{\widetilde{I} \subseteq I}$ при ограничении $\bigcup\limits_{i \in \widetilde{I}} M_i = J$. Лемма 1 доказана.

Согласно лемме 1 максимум вышеописанных частных случаев исследуемых задач достигается на $\widetilde{I} = \widetilde{N}^*$ — покрытии минимального веса, т. е. когда $\{i \in I : y_i = 1\} = \widetilde{N}^*$. Теорема 1 доказана.

Доказать NP-трудность задачи LMP можно ещё проще. Для этого достаточно рассмотреть её частный случай, когда все предприятия уже построены, т. е. при условии $f_i = 0$. Сведéние задачи о минимальном покрытии к этому частному случаю задачи LMP описано в [3].

Рассмотрим полный взвешенный граф K^{n+m}, в котором вершинами являются потребители и возможные места открытия предприятий, а веса рёбер — транспортные затраты. Задачу LDP (LMP, LUP), определённую на графе K^{n+m}, обозначим через LDP^K (LMP^K, LUP^K). Тогда из теоремы 1, если $c_{ik} = 2$, где либо $i, k \in I$, либо $i, k \in J$, вытекает

Следствие 1. *Задачи* LDP^K, LMP^K *и* LUP^K NP-*трудны в сильном смысле, даже если транспортные затраты удовлетворяют неравенству треугольника.*

Введём ряд обозначений, соответствующих произвольной оптимизационной задаче A с критерием максимизации целевой функции:

$L(A)$ — множество примеров задачи A (при этом произвольный пример $t \in L(A)$ будем называть задачей t);

$\mathrm{OPT}_A(t)$ — оптимальное значение целевой функции в задаче $t \in L(A)$;

$D_A(t)$ — множество допустимых решений задачи $t \in L(A)$;

$F_A(t, s)$ — значение целевой функции в задаче $t \in L(A)$ на решении $s \in D_A(t)$.

Из теоремы 1 следует, что при условии P$\neq$NP нахождение оптимального решения любой из задач LDP, LMP и LUP с ростом размерности становится достаточно трудоёмким процессом. Тогда имеет смысл рассмотреть вопрос нахождения «неплохого» допустимого решения. Обычно в этом случае рассматривают сложность задачи с точки зрения построения эффективного алгоритма нахождения приближённого решения с гарантированной оценкой точности, т. е. положение оптимизационной задачи в иерархии аппроксимационных классов [7]

$$\begin{aligned} &\mathrm{PO} \subseteq \mathrm{FPTAS} \subseteq \mathrm{PTAS} \subseteq \mathrm{APX} \subseteq \text{Log-APX} \\ &\qquad\qquad\qquad\qquad \subseteq \text{Poly-APX} \subseteq \text{Exp-APX} \subseteq \mathrm{NPO}. \end{aligned}$$

Каждый из этих классов описывает определённое качество аппроксимации, которым обладают образующие его оптимизационные задачи. Данная иерархия используется для описания свойств задач из класса NPO. Содержательно его можно описать как класс оптимизационных задач, у которых соответствующая задача распознавания принадлежит классу NP. Класс PO образован задачами, для каждой из которых существует точный полиномиальный алгоритм решения. Класс FPTAS состоит из задач, для которых существуют вполне полиномиальные приближённые схемы решения, а класс PTAS образован задачами, для которых существуют полиномиальные приближённые схемы решения. Классы APX, Log-APX, Poly-APX, Exp-APX состоят из задач, для которых существуют полиномиальные приближённые алгоритмы решения с константной, логарифмической, полиномиальной и экспоненциальной оценками точности погрешности соответственно. В последних трёх случаях значения указанных функций зависят от длины записи исходных данных задачи. Формальные определения можно найти в [7, 8]. Также известно, что при условии P≠NP указанные выше включения между классами строгие [7, 8, 10].

При определении данной иерархии погрешность решения $s \in D_A(t)$ задачи A с входом $t \in L(A)$ задаётся величиной

$$R_A(t,s) = \max\left\{\frac{F_A(t,s)}{\mathrm{OPT}_A(t)}, \frac{\mathrm{OPT}_A(t)}{F_A(t,s)}\right\} \geqslant 1.$$

Понятно, что если A является оптимизационной задачей с критерием максимизации целевой функции, то $R_A(t,s) = \frac{\mathrm{OPT}_A(t)}{F_A(t,s)}$.

Ранее в [4] показано, что при фиксированном размещении задача LMP принадлежит классу Log-APX, а задачи LDP и LUP полиномиально разрешимы. Далее рассмотрим лемму, в которой устанавливается в некотором смысле «ограничение сверху» на положение исследуемых задач в аппроксимационной иерархии.

Лемма 2. *Задачи* LDP*,* LMP *и* LUP *принадлежат классу* Poly-APX*.*

Доказательство. Рассмотрим частные случаи исследуемых задач. Обозначим через $\mathrm{L^1DP}$, $\mathrm{L^1MP}$ и $\mathrm{L^1UP}$ задачи LDP, LMP и LUP, в которых открывается не более одного предприятия. В [3] показано, что задачи $\mathrm{L^1DP}$, $\mathrm{L^1MP}$ и $\mathrm{L^1UP}$ полиномиально разрешимы. Оптимальные решения этих задач и будем использовать в качестве допустимого решения исходных задач. Далее ограничимся рассмотрением только задачи LDP. Для задач LMP и LUP доказательство проводится аналогично.

Очевидно, что произвольное оптимальное решение задачи $\mathrm{L^1DP}$ является допустимым решением задачи LDP. Покажем, что

$$\mathrm{OPT_{LDP}}(t) \leqslant n * \mathrm{OPT_{L^1DP}}(t),$$

где $n = |I|$. Пусть $(\widetilde{y}, \widetilde{p}, \widetilde{x})$ — произвольное допустимое решение задачи LDP. Введём следующие обозначения:

$$i^* = \arg \max_{i \in I \,:\, \widetilde{y}_i = 1} \Big\{ \sum_j \widetilde{p}_{ij} \widetilde{x}_{ij} - f_i \Big\},$$

$$y_i^{i^*} = \begin{cases} 1, & \text{если } i = i^*, \\ 0 & \text{в противном случае,} \end{cases}$$

$$x_{ij}^{i^*} = \begin{cases} \widetilde{x}_{ij}, & \text{если } i = i^*, \\ 0 & \text{в противном случае.} \end{cases}$$

Тогда $(y^{i^*}, \widetilde{p}, x^{i^*})$ является допустимым решением задачи $\mathrm{L^1DP}$. Кроме того, по определению верно следующее соотношение:

$$F_{\mathrm{LDP}}(t, (\widetilde{y}, \widetilde{p}, \widetilde{x})) = \sum_{i,j} \widetilde{p}_{ij} \widetilde{x}_{ij} - \sum_i f_i \widetilde{y}_i \leqslant n * \Big(\sum_{i,j} \widetilde{p}_{ij} x_{ij}^{i^*} - \sum_i f_i y_i^{i^*} \Big) \leqslant n * \mathrm{OPT}_{\mathrm{L^1DP}}(t).$$

Получается, что оптимальное решение задачи $\mathrm{L^1DP}$ является допустимым решением задачи LDP с оценкой точности n. Лемма 2 доказана.

Следствие 2. *Задачи* LDP^K, LMP^K *и* LUP^K *принадлежат классу* Poly-APX, *даже если транспортные затраты удовлетворяют неравенству треугольника.*

В лемме 2 представлены эффективные алгоритмы нахождения допустимых решений задач LDP, LMP и LUP с оценками точности, ограниченными полиномами от длины входа соответствующих задач. Но данный результат не гарантирует отсутствия полиномиальных алгоритмов с лучшими оценками, соответствующих нижележащим классам иерархии. Теорема 2 позволяет решить данную проблему.

Введём определение AP-сводимости [7]. Пусть A и B — пара задач из класса NPO с критерием максимизации целевой функции. Будем говорить, что задача A AP-*сводится к задаче* B тогда и только тогда, когда существуют две функции φ и ρ и положительная константа α такие, что

(i) $\varphi(t, r) \in L(B)$ для любых $t \in L(A)$ и $r > 1$ и функция φ вычислима за полиномиальное время от длины $|t|$ входа t и r;

(ii) $\rho(t, s, r) \in D_A(t)$ для любых $t \in L(A)$, $r > 1$, $s \in D_B(\varphi(t, r))$ и ρ вычислима за полиномиальное время от $|t|$, длины решения s и r;

(iii) из того, что $R_B(\varphi(t, r), s) \leqslant r$, следует, что $R_A(t, \rho(t, s, r)) \leqslant 1 + \alpha(r - 1)$ для любых $t \in L(A)$, $r > 1$ и $s \in D_B(\varphi(t, r))$.

Напомним определение замкнутого подкласса класса NPO относительно некоторой сводимости. Подкласс C класса NPO *замкнуты относительно сводимости* Γ, если из Γ-сводимости задачи A к задаче $B \in C$ следует, что $A \in C$. Известно, что при условии P$\neq$NP включения в аппроксимационной иерархии между классами строгие, а классы FPTAS, PTAS, APX, Log-APX замкнуты относительно AP-сводимости [7, 8, 10]. Учитывая транзитивность AP-сводимости, получаем, что для любой Poly-APX-полной относительно AP-сводимости задачи при условии P$\neq$NP не может существовать полиномиальных приближённых алгоритмов с лучшими оценками погрешности допустимых решений от оптимальных, чем оценки, ограниченные полиномом от длины входа. Для исследуемых задач имеет место

Теорема 2. *Задачи* LDP, LMP *и* LUP *являются* Poly-APX-*полными относительно* AP-*сводимости.*

Доказательство. Рассмотрим Poly-APX-полную относительно AP-сводимости задачу о максимальном независимом множестве [7, 8, 10]. Пусть задан произвольный граф $G = (V, E)$, где V — множество вершин, а $E \subseteq V \times V$ — множество рёбер. *Независимым множеством* в графе G назовём подмножество вершин $\widetilde{V} \subseteq V$ такое, что $\widetilde{V} \times \widetilde{V} \cap E = \varnothing$. Тогда задачу о максимальном независимом множестве (MIS) можно представить в виде следующей оптимизационной задачи: $|\widetilde{V}| \to \max\limits_{\widetilde{V} \subseteq V}$ при ограничении $\widetilde{V} \times \widetilde{V} \cap E = \varnothing$.

Далее рассмотрим свéдение задачи MIS к задаче LDP, которое легко преобразовывается в свéдение к задачам LMP и LUP. Будем строить по входу $t \in L(\mathrm{MIS})$ вход $\varphi(t) \in L(\mathrm{LDP})$ следующим образом. Пусть $w(i) := |j \in V : (i, j) \in E|$, т. е. $w(i)$ — число вершин, смежных с вершиной i. Для каждого $i \in I$ рассмотрим набор потребителей J_i такой, что $J_i \cap J_k = \varnothing$, $k \in I \setminus i$, и $J_i \cap E = \varnothing$, а также $|J_i| = |I| - w(i) = n - w(i)$. Обозначим через $\varphi(t)$ вход, в котором $I := V$, $J := \big(\bigcup\limits_{i \in I} J_i\big) \cup E$, затраты на открытие любого предприятия равны $n-1$, бюджет каждого потребителя равен 2, а транспортные затраты определяются следующим образом: на предприятии i транспортные затраты c_{ij} равны 1, если $j \in J_i$ или $j = (i, k) \in E$, для некоторого $k \in I$, иначе транспортные затраты равны 2.

Пусть (y, p, x) — допустимое решение задачи LDP. Применим следующий алгоритм, строящий по решению (y, p, x) вспомогательное допустимое решение $(\widetilde{y}, \widetilde{p}, \widetilde{x})$ задачи LDP, по которому легко строится решение задачи MIS. Положим $\widetilde{p}_{ij} := 1$ при всех $i \in I$ и $j \in J$, а также $\widetilde{y} := y$ и $\widetilde{x} := x$. Просмотрим поочерёдно все предприятия $i \in I$. Если $\widetilde{y}_i = 1$

и предприятие i не приносит прибыли, т. е. $\sum_{j \in J} \widetilde{p}_{ij}\widetilde{x}_{ij} - f_i \leqslant 0$, то положим $\widetilde{y}_i := 0$ и $\widetilde{x}_{ij} := 0$ для всех $j \in J$.

В силу выбора цен и входных данных решение $(\widetilde{y}, \widetilde{p}, \widetilde{x})$ допустимо. Также очевидно, что нет никакой разницы, к какой именно задаче размещения и ценообразования относится это решение. Также отметим, что множество $\widetilde{V} = \{i \in I \, : \, \widetilde{y}_i = 1\}$ независимое, т. е. допустимое решение задачи MIS. Действительно, если существует пара отличных друг от друга предприятий $i, k \in \widetilde{V}$, то поскольку потребитель (i, k) может обслуживаться только одним предприятием (для определённости будем считать, что предприятие i его не обслуживает), прибыль, приносимая предприятием i, равна

$$\sum_{j \in J} \widetilde{p}_{ij}\widetilde{x}_{ij} - f_i = \sum_{j \in J \setminus (i,k)} \widetilde{p}_{ij}\widetilde{x}_{ij} - f_i \leqslant \sum_{s=\overline{1,n-1}} (1) - (n-1) = 0$$

в силу определения транспортных затрат и затрат на открытие предприятия. Отсюда следует, что $i \notin \widetilde{V}$. Получается, что $\widetilde{V}$ — независимое множество. Трудоёмкость вышеописанного алгоритма равна $O(n^2)$, а значит, полиномиальна от длины входа задачи MIS. Тем самым мы определили функцию ρ из определения AP-сводимости. Положим $\alpha = 1$. Чтобы вышеописанное сведéние задачи MIS к задаче LDP было AP-сведéнием, остаётся показать следующее:

$$\text{если } R_B(\varphi(t,r), s) \leqslant r, \text{ то } R_A(t, \rho(t,s,r)) \leqslant 1 + \alpha(r-1).$$

По построению решения $(\widetilde{y}, \widetilde{p}, \widetilde{x})$ выполнено неравенство

$$F_{\mathrm{LDP}}(\varphi(t), (y,p,x)) \leqslant F_{\mathrm{LDP}}(\varphi(t), (\widetilde{y}, \widetilde{p}, \widetilde{x})).$$

Оптимальное значение задачи LDP равно $n|V^*| - (n-1)|V^*| = |V^*|$, где V^* — оптимальное решение задачи MIS. Таким образом, получим

$$r \geqslant R_{\mathrm{LDP}}(\varphi(t), (y,p,x)) \geqslant \frac{\mathrm{OPT}_{\mathrm{LDP}}(t)}{F_{\mathrm{LDP}}(\varphi(t), (\widetilde{y}, \widetilde{p}, \widetilde{x}))} = \frac{n|V^*| - (n-1/n)|V^*|}{n|\widetilde{V}| - (n-1/n)|\widetilde{V}|} = \frac{|V^*|}{|\widetilde{V}|}.$$

Учитывая, что $\alpha = 1$, получим неравенство

$$R_{\mathrm{MIS}}(t, \rho(t, (y,p,x))) = \frac{|V^*|}{|\widetilde{V}|} \leqslant r.$$

Для задач LMP и LUP аналогично.

Получается, что любая задача из класса Poly-APX AP-сводится к исследуемым задачам размещения и ценообразования, а из леммы 2 следует их принадлежность классу Poly-APX. Теорема 2 доказана.

Если положить $c_{ik} = 2$, где либо $i, k \in I$, либо $i, k \in J$, из теоремы 2 и следствия 2 вытекает

Следствие 3. *Задачи* LDP^K, LMP^K *и* LUP^K Poly-APX-*полны относительно* AP-*сводимости, даже если транспортные затраты удовлетворяют неравенству треугольника.*

Заключение

Основной результат работы заключается в том, что исследуемые двухуровневые задачи с равномерным, фабричным и дискриминационным ценообразованием Poly-APX-полны относительно AP-сводимости. Таким образом, при условии P≠NP для данных задач не может существовать полиномиальных точных или приближённых алгоритмов с лучшими оценками погрешности допустимых решений от оптимальных, чем оценки, ограниченные полиномом от длины входа.

В дальнейшем также предполагается уточнить связи данных задач с полиномиальной иерархией.

ЛИТЕРАТУРА

1. **Гэри М., Джонсон Д.** Вычислительные машины и труднорешаемые задачи. — М.: Мир, 1982. — 416 с.
2. **Кочетов Ю. А., Плясунов А. В.** Полиномиально разрешимый класс задач двухуровневого линейного программирования // Дискрет. анализ и исслед. операций. Сер. 2. — 1997. — Т. 1, № 2. — С. 23–33.
3. **Панин А. А., Плясунов А. В.** Задача ценообразования. Ч. 1. Точные и приближённые алгоримы решения // Дискрет. анализ и исслед. операций. — 2012. — Т. 19, № 2. — С. 31–40.
   Panin A. A., Plyasunov A. V. The pricing problem. I: Exact and approximate algorithms // J. Appl. Industr. Math. — 2013. — Vol. 7, N 2. — P. 241–251.
4. **Панин А. А., Плясунов А. В.** Задача ценообразования. Ч. 2. Вычислительная сложность // Дискрет. анализ и исслед. опер. — 2012. — Т. 19, № 6. — С. 56–71.
   Panin A. A., Plyasunov A. V. The pricing problem. II: Computational complexity // J. Appl. Industr. Math. — 2013. — Vol. 7, N 3. — P. 420–430.
5. **Aboolian R., Berman O., Krass D.** Competitive facility location model with concave demand // Eur. J. Oper. Res. — 2007. — Vol. 181. — P. 598–619.

6. **Aboolian R., Berman O., Krass D.** Optimizing pricing and location decisions for competitive service facilities charging uniform price // J. Oper. Res. Soc. — 2008. — Vol. 59. — P. 1506–1519.
7. **Ausiello G., Crescenzi P., Gambosi G., Kann V., Marchetti-Spaccamela A., Protasi M.** Complexity and approximation: combinatorial optimization problems and their aproximability properties. — Berlin: Springer-Verl., 1999. — 524 p.
8. **Bazgan C., Escoffer B., Paschos V. Th.** Completeness in standard and differential approximation classes: Poly-(D)APX- and (D)PTAS-completeness // Theor. Comput. Sci. — 2005. — Vol. 339. — P. 272–292.
9. **Bouhtou M., Grigoriev A., Van Der Kraaij A. F., Van Hoesel S., Spieksma F., Uetz M.** Pricing bridges to cross a river // Naval Res. Logistics. — 2007. — Vol. 54, N 4. — P. 411–420.
10. **Crescenzi P., Kann V., Silvestri R., Trevisan L.** Structure in approximation classes // SIAM J. Comput. — 1999. — Vol. 28, N 5. — P. 1759–1782.
11. **Dasci A., Laporte G.** Location and pricing decisions of a multistore monopoly in a spatial market // J. Region. Sci. — 2004. — Vol. 44. — P. 489–515.
12. **Dempe S. J.** Foundations of bilevel programming. — Dordrecht: Kluwer Acad. Publ., 2002. — 481 p.
13. **Eiselt H. A., Marianov V.** Foundations of location analysis. — New York: Springer-Verl., 2011. — 510 p.
14. **Grigoriev A., van Loon J., Sviridenko M., Uetz M., Vredeveld T.** Optimal bundle pricing with monotonicity constraint // Oper. Res. Lett. — 2008. — Vol. 36, N 5. — P. 609–614.
15. **Hanjoul P., Hansen P., Peeters D., Thisse J.-F.** Uncapacitated plant location under alternative spatial price policies // Market Sci. — 1990. — Vol. 36. —P. 41–57.
16. **Serra D., ReVelle C.** Competitive locations and pricing on networks // Geograph. Anal. — 1999. — Vol. 31. — P. 109–129.
17. **Vives X.** Oligopoly pricing: old ideas and new tools. — Cambridge, MA: MIT Press, 1999. — 425 p.

Панин Артём Александрович,
e-mail: arteam1897@gmail.com
Плясунов Александр Владимирович,
e-mail: apljas@math.nsc.ru

Статья поступила
15 октября 2013 г.

Переработанный вариант —
29 января 2014 г.