

ISSN 2949-5598

ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ

Том 31 № 4 2024

Новосибирск
Издательство Института математики

АЛГОРИТМ ВЕТВЕЙ, ГРАНИЦ И ОТСЕЧЕНИЙ
ДЛЯ НЕЯВНОЙ ДИНАМИЧЕСКОЙ ЗАДАЧИ
КОНКУРЕНТНОГО РАЗМЕЩЕНИЯ

В. Л. Береснев^{1, 2, a}, *А. А. Мельников*^{1, 2, b}

¹ Институт математики им. С. Л. Соболева,
пр. Акад. Коптюга, 4, 630090 Новосибирск, Россия

² Новосибирский гос. университет,
ул. Пирогова, 2, 630090 Новосибирск, Россия

E-mail: ^aberesnev@math.nsc.ru, ^bmelnikov@math.nsc.ru

Аннотация. Рассматривается динамическая задача конкурентного размещения, моделирующая взаимодействие двух соперничающих сторон (Лидера и Последователя), размещающих свои объекты на рассматриваемом горизонте планирования, состоящего из некоторого числа периодов времени. Предполагается, что Лидер открывает свои объекты в начале горизонта планирования и не изменяет это решение внутри горизонта планирования. При этом Последователь может корректировать своё решение в каждом периоде горизонта планирования. Предлагается алгоритм поиска наилучшего решения, построенный на основе вычислительной схемы ветвей и границ. Для вычисления верхних границ используется специальная релаксация исходной двухуровневой задачи, усиленная дополнительными ограничениями. Предлагаются процедуры построения таких ограничений, в которых используются вспомогательные оптимизационные задачи для получения наиболее сильных отсечений. На примере задачи динамического конкурентного размещения на сети с тремя вершинами демонстрируются возможности модели учитывать информацию об изменениях параметров с течением времени. Реализация алгоритма ветвей и границ демонстрирует значительный эффект от использования отсечений, предложенных специально для динамических моделей: верхняя граница вычисляется более точно и сокращается число вершин дерева ветвления. Ил. 2, библиогр. 8.

Ключевые слова: задача конкурентного размещения предприятий, двухуровневое математическое программирование, точный метод, игра Штакельберга.

Введение

Задачи конкурентного размещения относятся к семейству дискретных задач двухуровневого математического программирования, построенных на базе игры Штакельберга [1]. В таких моделях две соперничающие стороны (Лидер и Последователь) последовательно размещают свои предприятия и стремятся захватить потребителей с целью получения максимальной прибыли от их обслуживания. В случае, когда возможность изменения составов открытых сторонами предприятий в модели не предусмотрена, будем говорить о *статических* моделях конкурентного размещения. Обзор результатов исследований в области статических моделей конкурентного размещения можно найти в недавних обзорах [2, 3].

Динамические задачи конкурентного размещения образуют новый, мало изученный к настоящему моменту класс задач. Они обобщают статические задачи на случай, когда, во-первых, некоторые данные (например, множество потребителей) не фиксированы, а изменяются на рассматриваемом горизонте планирования, а во-вторых, решения сторон о составе размещаемых предприятий могут изменяться внутри горизонта планирования. Обычно предполагается, что горизонт планирования разбит на периоды (отрезки) времени, на каждом из которых внешние данные не изменяются и возможна корректировка решений, принятых сторонами ранее. Первое известное упоминание динамической задачи конкурентного размещения содержится, по-видимому, в [4]. Однако общие постановки и предложения способов для построения наилучших решений таких задач нам не известны.

В работе авторов [5] предложена постановка неявной динамической задачи конкурентного размещения. В неявных динамических задачах [6] учитываются изменения внешних факторов, влияющих на принятие решений, но сами решения о размещении предприятий, принятые в начале горизонта планирования, не корректируются в дальнейшем. Исследуемая в [5] модель является неявной динамической задачей в том смысле, что Лидер открывает свои предприятия в начале горизонта планирования и не изменяет состава своих открытых предприятий. Напротив, Последователь имеет возможность изменять состав своих предприятий в каждом периоде рассматриваемого горизонта планирования. Это задача двухуровневого математического программирования. Для её решения предлагается использовать вычислительную схему ветвей и границ и способ вычисления верхних границ значений целевой функции задачи, базирующийся на использовании специфической для двухуровневых моделей релаксации, называемой в англоязычной литературе *high-point relaxation* [7] (задачи HPR). Для усиления задачи HPR предложено использовать модифицированные *s*-отсечения [8] и новые дополнительные

ограничения, названные d-отсечениями [5], которые построены с учётом специфики переменных динамических задач.

В настоящей работе разработаны процедуры построения указанных дополнительных ограничений, в которых используются вспомогательные оптимизационные задачи для получения наиболее сильных c-отсечений и d-отсечений. Такие ограничения отсекают наибольшее число допустимых решений задачи НРР, не являющихся допустимыми решениями исходной двухуровневой задачи. В работе даётся подробное описание процедуры построения дополнительных ограничений и процедуры вычисления верхних границ, а также алгоритма поиска решения рассматриваемой задачи. Приводится иллюстрация возможностей динамической модели конкурентного размещения и демонстрируется позитивный эффект от использования d-отсечений при вычислении верхних оценок в работе алгоритма ветвей и границ.

1. Формулировка задачи

Рассматриваемая в [5] неявная динамическая задача конкурентного размещения (неявная DCompFLP) записывается как следующая дискретная задача двухуровневого математического программирования:

$$\max_{(x_i), (\chi_{ij})} \left(- \sum_{i \in I} f_i x_i + \sum_{t \in T} \sum_{i \in I} \sum_{j \in J_t} p_{ij} \chi_{ij} \right), \quad (1)$$

$$\sum_{\tau=1}^t z_{i\tau}^0 + \sum_{k \preceq_j i} \chi_{kj} \leq 1, \quad i \in I, t \in T, j \in J_t, \quad (2)$$

$$x_i \geq \chi_{ij}, \quad i \in I, j \in J, \quad (3)$$

$$x_i, \chi_{ij} \in \{0, 1\}, \quad i \in I, j \in J, \quad (4)$$

$$(z_{it}^0), (\zeta_{ij}^0) \text{ — оптимальное решение задачи} \quad (5)$$

$$\max_{(z_{it}), (\zeta_{ij})} \left(- \sum_{t \in T} \sum_{i \in I} g_{it} z_{it} + \sum_{t \in T} \sum_{i \in I} \sum_{j \in J_t} q_{ij} \zeta_{ij} \right), \quad (6)$$

$$x_i + \sum_{t \in T} z_{it} \leq 1, \quad i \in I, \quad (7)$$

$$x_i + \sum_{k \preceq_j i} \zeta_{kj} \leq 1, \quad i \in I, j \in J, \quad (8)$$

$$\sum_{\tau=1}^t z_{i\tau} \geq \zeta_{ij}, \quad i \in I, t \in T, j \in J_t, \quad (9)$$

$$z_{it}, \zeta_{ij} \in \{0, 1\}, \quad i \in I, t \in T, j \in J. \quad (10)$$

В этой модели используются следующие обозначения:

- I — множество предприятий (мест возможного размещения предприятий);
- T — множество периодов (единичных отрезков) времени рассматриваемого горизонта планирования;
- $J_t, t \in T$ — множество потребителей в период времени t ; без ограничения общности полагаем, что $J_{t_1} \cap J_{t_2} = \emptyset$ при любых $t_1, t_2 \in T, t_1 \neq t_2$;
- $J = \bigcup_{t \in T} J_t$ — множество потребителей на всём горизонте планирования;
- $f_i, i \in I$ — фиксированные затраты на открытие Лидером предприятия i ;
- $g_{it}, i \in I, t \in T$ — фиксированные затраты на открытие Последователем предприятия i в период t ; для всякого $i \in I$ будем считать, что величины (g_{it}) убывают с ростом номера t ;
- $p_{ij}, i \in I, j \in J$ — величина дохода, получаемого Лидером от обслуживания открытым им предприятием i потребителя j ;
- $q_{ij}, i \in I, j \in J$ — величина дохода, получаемого Последователем от обслуживания открытым им предприятием i потребителя j .

В модели используются следующие двоичные переменные:

- $x_i, i \in I$, равна единице, если Лидер открывает предприятие i , и нулю в противном случае;
- $z_{it}, i \in I, t \in T$, равна единице, если Последователь открывает предприятие i в период t , и нулю в противном случае;
- $\chi_{ij}, i \in I, j \in J$, равна единице, если предприятие i , открытое Лидером, назначено для обслуживания потребителя j , и равна нулю в противном случае;
- $\zeta_{ij}, i \in I, j \in J$, равна единице, если предприятие i , открытое Последователем, назначено для обслуживания потребителя j , и нулю в противном случае.

Линейный порядок \succeq_j на множестве I используется в модели для задания предпочтений потребителя $j \in J$ при выборе предприятия для его обслуживания. Для пары $i_1, i_2 \in I$ отношение $i_1 \succ_j i_2$ означает, что для потребителя $j \in J$ предприятие i_1 предпочтительнее предприятия i_2 . Обозначение $i_1 \succeq_j i_2$ означает, что $i_1 \succ_j i_2$ или $i_1 = i_2$. В модели полагаем, что для обслуживания потребителя может быть назначено любое открытое предприятие одной из сторон, которое более предпочтительно, чем любое предприятие, открытое другой стороной.

Рассматриваемая модель, как и всякая задача двухуровневого программирования, включает задачу верхнего уровня (1)–(5) (задачу Лидера) и задачу нижнего уровня (6)–(10) (задачу Последователя). Задачу (1)–(5) обозначим через \mathcal{L} , а задачу нижнего уровня (6)–(10) — через \mathcal{F} . Для задачи (1)–(10) в целом используем обозначение $(\mathcal{L}, \mathcal{F})$.

Пару (X, Z) , где $X = ((x_i), (\chi_{ij}))$, назовём *допустимым решением задачи* $(\mathcal{L}, \mathcal{F})$, порождённым двоичным вектором $x = (x_i)$, если X — допустимое решение задачи \mathcal{L} , а Z — оптимальное решение задачи \mathcal{F} . Допустимое решение (X, Z) , порождённое вектором x , назовём *пессимистическим*, если $L(X, Z) \leq L(X', Z')$ для любого допустимого решения (X', Z') , порождённого вектором x . Наилучшее pessimisticкое допустимое решение назовём *пессимистическим оптимальным решением задачи* $(\mathcal{L}, \mathcal{F})$.

Пессимистическое допустимое решение, порождённое вектором x , как и в случае статической задачи [8], может быть построено по вектору x в результате решения двух оптимизационных задач. Первая из них есть задача \mathcal{F} , оптимальное значение целевой функции которой обозначим через F^* , а вторая — вспомогательная задача вида

$$\begin{aligned} & \min_{(z_{it}), (\zeta_{ij}), (v_i)} \sum_{j \in J} v_j, \\ v_j & \geq p_{ij} \left(x_i - \sum_{k \succ_j i} \sum_{\tau=1}^t z_{k\tau} \right), \quad i \in I, t \in T, j \in J_t, \\ & - \sum_{i \in I} \sum_{t \in T} g_{it} z_{it} + \sum_{i \in I} \sum_{j \in J} q_{ij} \zeta_{ij} \geq F^*, \\ & x_i + \sum_{t \in T} z_{it} \leq 1, \quad i \in I, \\ & x_i + \sum_{k \preceq_j i} \zeta_{kj} \leq 1, \quad i \in I, j \in J, \\ & \sum_{\tau=1}^t z_{i\tau} \geq \zeta_{ij}, \quad i \in I, t \in T, j \in J_t, \\ & z_{it}, \zeta_{ij} \in \{0, 1\}, v_j \geq 0, \quad i \in I, t \in T, j \in J. \end{aligned}$$

Отметим, что возможность построения pessimisticкого допустимого решения по вектору x позволяет рассматривать задачу $(\mathcal{L}, \mathcal{F})$ как задачу максимизации некоторой псевдодобулевой функции $L(x)$, поэтому значение целевой функции задачи $(\mathcal{L}, \mathcal{F})$ на pessimisticком допустимом решении (X, Z) , порождённом вектором x , будем обозначать также через $L(x)$.

2. Дополнительные ограничения для задачи $(\mathcal{L}, \mathcal{F})$

Основой алгоритма ветвей и границ для поиска pessimisticкого оптимального решения исследуемой задачи $(\mathcal{L}, \mathcal{F})$ является процедура

вычисления верхних границ целевой функции на рассматриваемых в ходе работы алгоритма подмножествах решений. Такие подмножества будем задавать частичным двоичным вектором $y = (y_i)_{i \in I}$, где $y_i \in \{0, 1, *\}$. Частичное решение $y = (y_i)$ определяет фиксированные компоненты двоичных векторов $x = (x_i)$, порождающих пессимистические допустимые решения задачи $(\mathcal{L}, \mathcal{F})$ из рассматриваемого подмножества решений. Если $y_i = *$, то значение переменной x_i не фиксировано. Для частичного решения $y = (y_i)$ положим $I^0(y) = \{i \in I \mid y_i = 0\}$ и $I^1(y) = \{i \in I \mid y_i = 1\}$. Тогда подмножество решений задачи $(\mathcal{L}, \mathcal{F})$, задаваемое частичным решением $y = (y_i)$, представляет собой множество пессимистических допустимых решений задачи $(\mathcal{L}, \mathcal{F})$, порождённых двоичными векторами $x = (x_i)$, для которых

$$x_i = y_i, \quad i \in I^0(y) \cup I^1(y). \quad (11)$$

Задачу $(\mathcal{L}, \mathcal{F})$ с дополнительным ограничением (11) в задаче \mathcal{L} будем обозначать через $(\mathcal{L}(y), \mathcal{F})$.

Основой для вычисления верхних границ задачи $(\mathcal{L}, \mathcal{F})$ может служить релаксированная задача, называемая в англоязычной литературе high-point relaxation (далее задача HPR). Эта задача получается из исходной двухуровневой задачи в результате исключения из неё требования оптимальности для переменных задачи нижнего уровня. Применительно к задаче $(\mathcal{L}(y), \mathcal{F})$ задача HPR может быть записана в следующем виде:

$$\max_{(x_i), (\chi_{ij}), (z_{it})} \left(- \sum_{i \in I} f_i x_i + \sum_{t \in T} \sum_{i \in I} \sum_{j \in J_t} p_{ij} \chi_{ij} \right), \quad (12)$$

$$\sum_{\tau=1}^t z_{i\tau} + \sum_{k \preceq j i} \chi_{kj} \leq 1, \quad i \in I, t \in T, j \in J_t, \quad (13)$$

$$x_i \geq \chi_{ij}, \quad i \in I, j \in J, \quad (14)$$

$$x_i + \sum_{t \in T} z_{it} \leq 1, \quad i \in I, \quad (15)$$

$$x_i = y_i, \quad i \in I^0(y) \cup I^1(y), \quad (16)$$

$$x_i, \chi_{ij}, z_{it} \in \{0, 1\}, \quad i \in I, t \in T, j \in J. \quad (17)$$

Для получения с использованием этой задачи нетривиальных верхних оценок её ограничения необходимо усилить некоторыми дополнительными ограничениями. Эти ограничения должны стимулировать переменные z_{it} принимать ненулевые значения и одновременно выполняться на пессимистических допустимых решениях задачи $(\mathcal{L}(y), \mathcal{F})$. Полученную в результате задачу назовём *усиленной оценочной задачей* (SEP) для задачи $(\mathcal{L}(y), \mathcal{F})$.

В [5] для исследуемой динамической задачи $(\mathcal{L}, \mathcal{F})$ предложены два вида дополнительных ограничений: с- и d-отсечения. Эти ограничения выполняются на пессимистических допустимых решениях, но при этом отсекают оптимальное решение задачи SEP, что позволяет улучшать точность верхней границы путём итеративного добавления новых отсечений. К сожалению, в некоторых случаях указанных дополнительных ограничений недостаточно для гарантированного отсечения оптимального решения задачи SEP, поэтому для улучшения верхней границы будем использовать также f-отсечения, предложенные в [8]. Система неравенств, образующих f-отсечение, гарантированно отсекает оптимальное решение задачи SEP, если оно не является пессимистическим оптимальным решением задачи $(\mathcal{L}(y), \mathcal{F})$.

При построении дополнительных ограничений используются следующие обозначения. Пусть $x = (x_i)_{i \in I} \neq 0$ — двоичный вектор, $J' \subseteq J$ — непустое подмножество, $k \in I$, $j \in J$. Положим

- $I^1(x) = \{i \in I \mid x_i = 1\}$,
- $I^0(x) = \{i \in I \mid x_i = 0\}$,
- $\alpha_j(x)$ — такой элемент $i' \in I^1(x)$, что $i' \succeq_j i$ для всех $i \in I^1(x)$,
- $\alpha_{J'}(x) = \{\alpha_j(x) \mid j \in J'\}$,
- $N_j(x) = \{i \in I \mid i \succ_j \alpha_j(x)\}$,
- $N_{J'}(x) = \bigcup_{j \in J'} N_j(x)$,
- $\bar{N}_j(k) = \{i \in I \mid i \succeq_j k\}$,
- $\bar{N}_{J'}(k) = \bigcup_{j \in J'} \bar{N}_j(x)$.

Пусть (X', z') , $X' = ((x'_i), (\chi'_{ij}))$, $z' = (z'_{it})$ — оптимальное решение задачи SEP, $u'_{it} = \sum_{\tau=1}^t z'_{it}$, $i \in I$, $t \in T$, и $x' = (x'_i)$, $u'_t = (u'_{it})$, $t \in T$. Для всякого $t \in T$ положим

$$J_{0t} = \begin{cases} J_t, & \text{если } u'_t \text{ — нулевой вектор,} \\ \{j \in J_t \mid \alpha_j(x') \succ_j \alpha_j(u'_t)\} & \text{в противном случае.} \end{cases}$$

Исходные данные для построения с-отсечения решения (X', z') представляет собой номер $k \in I$ такой, что $x'_k = 0$ и $\sum_{t \in T} z'_{kt} = 0$. Рассмотрим множества $J_{0t}(k) = \{j \in J_{0t} \mid k \in N_j(x')\}$, $t \in T$. Пусть $t_1 \in T$ — наименьший номер, для которого при некотором $t_0 \in T$, $1 \leq t_0 \leq t_1$, для множества $J_0 = \bigcup_{t=t_0}^{t_1} J_{0t}(k)$ выполняются неравенства

$$\sum_{j \in J_0} p_{\alpha_j(x')j} > 0, \quad \sum_{j \in J_0} q_{kj} \geq g_{kt_0}. \quad (18)$$

Пусть для подмножества $J' \subseteq J_0$ выполняются аналогичные неравенства. Тогда неравенство

$$\sum_{i \in N_{J'}(x')} u_{it_1} \geq 1 + \sum_{i \in \alpha_{J'}(x')} (x_i - 1) - \sum_{i \in \bar{N}_{J'}(k)} x_i \quad (19)$$

есть с-отсечение решения (X', z') , сгенерированное номером k и подмножеством J' .

Исходные данные для построения d-отсечения решения (X', z') представляет собой номер $k \in I$ такой, что $z'_{kl} = 1$ для $l > 1$. Рассмотрим множества $J_{0t}(k) = \{j \in J_{0t} \mid k \in N_j(x')\}$, $t \in T$. Пусть $t_1 \in T$, $t_1 < l$ — наименьший номер, для которого при некотором $t_0 \in T$, $1 \leq t_0 \leq t_1$, существует подмножество J' множества $J_0 = \bigcup_{t=t_0}^{t_1} J_{0t}(k)$, для которого выполняются неравенства

$$\sum_{j \in J'} p_{\alpha_j(x')j} > 0, \quad \sum_{j \in J'} q_{kj} \geq g_{kt_0} - g_{kl}.$$

Тогда неравенство

$$\sum_{i \in N_{J'}(x')} u_{it_1} \geq 1 + (u_{kl} - 1) + \sum_{i \in \alpha_{J'}(x')} (x_i - 1) - \sum_{i \in \bar{N}_{J'}(k)} x_i \quad (20)$$

есть d-отсечение решения (X', Z') , сгенерированное номером k и подмножеством J' .

Неравенства (19) и (20) стимулируют величины u_{it_1} принимать ненулевые значения при двоичном векторе x' . Однако, правая часть указанных неравенств может быть равной единице и при других двоичных векторах. Число таких векторов зависит от числа элементов множеств $\alpha_{J'}(x')$ и $\bar{N}_{J'}(k)$, которые, в свою очередь, зависят от множества J' . Рассмотрим вспомогательные оптимизационные задачи, оптимальные решения которых дают множества J' , генерирующие наиболее сильные с- и d-отсечения для задачи $(\mathcal{L}(y), \mathcal{F})$.

Введём следующие обозначения:

- a_{ij} , $i \in I$, $j \in J_0$ — величина, равная единице, если $i \in \bar{N}_j(k)$, и нулю в противном случае;
- b_{ij} , $i \in I$, $j \in J_0$ — величина, равная единице, если $i = \alpha_j(x')$, и нулю в противном случае;
- c_j , $j \in J_0$ — величина, равная единице, если $\sum_{i \in I} b_{ij} p_{ij} > 0$, и нулю в противном случае;
- u_j , $j \in J_0$ — переменная, определяющая множество J' и равная единице, если $j \in J'$, и нулю в противном случае;
- w_i , $i \in I$ — переменная, определяющая множество $\alpha_{J'}(x')$, равная единице, если $i \in \alpha_{J'}(x')$, и нулю в противном случае;

• $v_i, i \in I$ — переменная, определяющая множество $\overline{N}_{J'}(k)$ и равная единице, если $i \in \overline{N}_{J'}(k)$, и нулю в противном случае.

С использованием введённых обозначений вспомогательная задача (задача АРС) для выбора множества J' при построении с-отсечения решения (X', z') , сгенерированным номером k и множеством J' , записывается следующим образом:

$$\begin{aligned} \min_{(u_j), (v_i), (w_i)} \quad & \sum_{i \in I \setminus I^1(y)} w_i + \sum_{i \in I \setminus I^0(y)} v_i, \\ & \sum_{j \in J_0} q_{kj} u_j \geq g_{kt_0}, \\ & \sum_{j \in J_0} c_j u_j \geq 1, \\ & v_i \geq a_{ij} u_j, \quad i \in I, j \in J_0, \\ & w_i \geq b_{ij} u_j, \quad i \in I, j \in J_0, \\ & u_j, v_i, w_i \in \{0, 1\}, \quad i \in I, j \in J_0. \end{aligned}$$

Оптимальное решение $((u_j), (v_i), (w_i))$ этой задачи определяет подмножество $J' = \{j \in J_0 \mid u_j = 1\}$ и множества $\overline{N}_{J'}(k) = \{i \in I \mid v_i = 1\}$, $\alpha_{J'}(x') = \{i \in I \mid w_i = 1\}$. Неравенство (19), для которого множество J' получено с использованием вспомогательной задачи АРС, будем называть с-отсечением решения (X', z') , сгенерированным номером k .

Вспомогательная задача (задача АРД) для выбора множества J' при построении d-отсечения отличается от рассмотренной задачи АРС только первым ограничением, которое в случае d-отсечения имеет вид

$$\sum_{j \in J_0} q_{kj} u_j \geq g_{kt_0} - g_{kl}.$$

Неравенство (20), для которого множество J' получено с использованием вспомогательной задачи АРД, будем называть d-отсечением решения (X', z') , сгенерированным номером k .

Резюмируя, получаем, что при заданном допустимом решении (X', z') задачи $(\mathcal{L}(y), \mathcal{F})$ процедуры построения с- и d-отсечения этого решения аналогичны. При выбранном номере $k \in I$ процедура построения с-отсечения состоит из следующих этапов. Во-первых, выбор номеров $t_1 \in T$ и $t_0 \in T, 1 \leq t_0 \leq t_1$. Далее построение по номерам t_0 и t_1 множества J_0 и проверка выполнения для этого множества условий (18). Если эти условия выполняются, то вычисляем оптимальное решение вспомогательной задачи АРС и получаем искомое подмножество J' и множества $\overline{N}_{J'}(k)$ и $\alpha_{J'}(x')$. При заданном номере $k \in I$ таком, что $z_{kl} = 1$ для некоторого $l \in T, l > 1$, процедура построения d-отсечения начинается с выбора

номеров $t_1 \in T$ и $t_0 \in T$, $1 \leq t_0 \leq t_1 < l$. Далее строим множество J_0 и проверяем справедливость условий (14) для множества J_0 . Если условия выполняются, то используем оптимальное решение вспомогательной задачи APD для выбора подмножества J' и множеств $\overline{N}_k(x')$ и $\alpha_{J'}(x')$.

В случае, когда для оптимального решения (X', z') задачи SEP не удаётся построить с- или d-отсечения этого решения, будем использовать f-отсечения. Процедура построения такого отсечения сводится к вычислению пессимистического допустимого решения (X^0, Z^0) задачи $(\mathcal{L}, \mathcal{F})$, порождённого вектором $x' = (x'_i)$. Соответствующие дополнительные ограничения имеют вид

$$\sum_{i: x'_i=1} (1 - x_i) + \sum_{i: x'_i=0} x_i \geq z_{kt}^0 - z_{kt} \quad (21)$$

для $k \in I$, $t \in T$ таких, что $z_{kt}^0 = 1$, и вынуждают переменные z_{kt} принимать значения z_{kt}^0 , если вектор $x' = (x'_i)$ есть часть допустимого решения (X', z') .

На начальном этапе построения задачи SEP для $(\mathcal{L}(y), \mathcal{F})$, когда имеется только задача НРР и $I^1(y) \neq \emptyset$, естественно строить с-отсечения для допустимого решения (X', z') , $X' = ((y'_i), (x'_{ij}))$, $z' = (z'_{it})$, где $y'_i = 1$ для $i \in I^1(y)$ и $y'_i = 0$ в противном случае, а $z_{it} = 0$, $i \in I$, $t \in T$.

Исходные данные для такого дополнительного ограничения, которое назовём *начальным с-отсечением*, составляет номер $k \in I \setminus I^1(y')$. При построении с-отсечений решения (X', z') будем использовать следующие обозначения. Пусть $j \in J$, $J' \subseteq J$, $J' \neq \emptyset$. Положим

$$J_{0t}(k) = \{j \in J_t \mid k \in N_j(y')\},$$

$$\overline{N}_j(k) = \{i \in I \mid i \succeq_j k\}, \quad \overline{N}_{J'}(k) = \bigcup_{j \in J'} \overline{N}_j(k).$$

Пусть $t_1 \in T$ — наименьший номер такой, что при некотором $t_0 \in T$, $1 \leq t_0 \leq t_1$, существует подмножество $J' \subseteq J_0 = \bigcup_{t=t_0}^{t_1} J_{0t}(k)$, для которого

$$\sum_{j \in J'} p_{\alpha_j(y')j} > 0, \quad \sum_{j \in J'} q_{kj} \geq g_{kt_0}. \quad (22)$$

Тогда неравенство

$$\sum_{j \in N_{J'}(y')} u_{it_1} \geq 1 - \sum_{i \in \overline{N}_{J'}(k)} x_i \quad (23)$$

назовём начальным с-отсечением для частичного решения $y = (y_i)$, сгенерированным номером k и подмножеством J' .

Утверждение 1. Пусть (X, Z) , $X = ((x_i), (\chi_{ij}))$, $Z = ((z_{it}), (\zeta_{ij}))$ есть пессимистическое допустимое решение задачи $(\mathcal{L}(y), \mathcal{F})$. Тогда неравенство (23) выполняется на этом решении.

ДОКАЗАТЕЛЬСТВО. Если вектор $x = (x_i)$ такой, что $x_i = 1$ для некоторого $i \in \overline{N}_{J'}(k)$, то неравенство (23) выполняется. Предположим, что $x_i = 0$ для всех $i \in \overline{N}_{J'}(k)$ и левая часть неравенства (23) равна нулю, т. е. данное неравенство нарушено. Поскольку $\alpha_j(x) \succeq_j \alpha_j(y')$ для $j \in J'$, то $u_{it_1} = 0$ для $i \in N_{J'}(x)$. В частности, имеем $u_{kt_0} = 0$. При заданном векторе x рассмотрим допустимое решение (X', Z') , где $X' = ((x_i), (\chi'_{ij}))$ и $Z' = ((z'_{it}), (\zeta'_{ij}))$ отличаются от текущего решения (X, Z) только в том, что $z'_{kt_0} = 1$, $\zeta'_{kj} = 1$ для всех $j \in J'$ и $\chi_{ij} = 0$ для всех $i \in I$ и $j \in J_t$ таких, что $t \geq t_0$ и $k \succ_j i$. В силу свойств (22) для целевых функций L и F Лидера и Последователя соответственно имеем $L(X') < L(X)$ и $F(Z') \geq F(Z)$, что противоречит тому, что (X, Z) является пессимистическим допустимым решением. Утверждение 1 доказано.

Отметим, что неравенство (23) стимулирует переменные z_{it} принимать ненулевые значения при любом допустимом решении (X', z') задачи SEP для задачи $(\mathcal{L}(y), \mathcal{F})$, если $\overline{N}_{J'}(k) \subseteq I^0(x')$. Число таких решений зависит от выбора множества J' . Аналогично случаю с-отсечения общего вида (19) рассмотрим вспомогательную задачу для поиска множества J' .

Как и в случае с-отсечений, будем использовать ранее введённые обозначения:

- a_{ij} , $i \in I$, $j \in J_0$ — величина, равная единице, если $i \in \overline{N}_j(k)$, и нулю в противном случае;
- b_{ij} , $i \in I$, $j \in J_0$ — величина, равная единице, если $i = \alpha_j(y')$, и нулю в противном случае;
- c_j , $j \in J_0$ — величина, равная единице, если $\sum_{i \in I} b_{ij} p_{ij} > 0$, и нулю в противном случае;
- u_j , $j \in J_0$ — переменная, определяющая множество J' и равная единице, если $j \in J'$, и нулю в противном случае;
- v_i , $i \in I$ — переменная, определяющая множество $\overline{N}_{J'}(k)$ и равная единице, если $i \in \overline{N}_{J'}(k)$, и нулю в противном случае.

Вспомогательная задача для выбора подмножества J' , генерирующего при заданном $k \in I$ наиболее сильное начальное с-отсечение для задачи $(\mathcal{L}(y), \mathcal{F})$ записывается следующим образом:

$$\min_{(v_i), (u_j)} \sum_{i \in I \setminus I^0(y)} v_i,$$

$$\sum_{j \in J_0} q_{kj} u_j \geq q_{kt_0},$$

$$\begin{aligned} \sum_{j \in J_0} c_j u_j &\geq 1, \\ v_i &\geq a_{ij} u_j, \quad i \in I, j \in J_0, \\ u_j, v_i &\in \{0, 1\}, \quad i \in I, j \in J. \end{aligned}$$

Неравенство (23) с множеством J' , построенное по оптимальному решению рассмотренной вспомогательной задачи, будем называть начальным s -отсечением, сгенерированным номером k .

Заметим, что в силу следующего утверждения множество $K \subseteq I$, включающее номера k , для которых удаётся построить начальные s -отсечения, может быть использовано при выборе номера $k \in I$ для построения s -отсечения решения (X', Z') задачи SEP для задачи $(\mathcal{L}(y), \mathcal{F})$.

Утверждение 2. *Если для задачи $(\mathcal{L}(y), \mathcal{F})$ не существует начального s -отсечения, сгенерированного номером $k \in I$, то при любом допустимом решении (X', Z') задачи SEP не существует s -отсечения решения (X', Z') , сгенерированного номером k .*

Доказательство. Действительно, поскольку для всякого $t \in T$ множество J_{0t} для начального s -отсечения включает множество J_{0t} для s -отсечения решения (X', Z') , это включение сохраняется и для множеств $J_{0t}(k)$ и J_0 . Утверждение 2 доказано.

3. Процедура вычисления верхней границы

В результате выполнения процедуры для вычисления верхней границы целевой функции задачи $(\mathcal{L}, \mathcal{F})$ на подмножестве решений, заданном частичным решением y , получаем величину $UB(y)$ верхней границы для целевой функции задачи $(\mathcal{L}(y), \mathcal{F})$ и два двоичных вектора x^0 и x' . Ненулевой вектор x^0 порождает пессимистическое допустимое решение (X^0, Z^0) задачи $(\mathcal{L}(y), \mathcal{F})$, найденное при выполнении процедуры, а нулевой вектор x^0 означает, что такого решения найти не удалось. Вектор x' — часть оптимального решения (X', z') финальной задачи SEP для задачи $(\mathcal{L}(y), \mathcal{F})$, построенной в процессе вычисления верхней границы.

При выполнении процедуры кроме заданного частичного решения $y = (y_i)$ считаем известной величину L^* , равную значению $L(x^*)$ целевой функции задачи $(\mathcal{L}, \mathcal{F})$ на наилучшем известном (рекордном) пессимистическом допустимом решении (X^*, Z^*) задачи $(\mathcal{L}, \mathcal{F})$, порождённом вектором x^* .

Если частичное решение $y = (y_i)$ представляет собой двоичный вектор, то процедура вычисления верхней границы сводится к построению пессимистического допустимого решения задачи $(\mathcal{L}, \mathcal{F})$, порождённого вектором $y = (y_i)$. В этом случае полагаем $UB(y) = L(y)$, $x^0 = y$, $x' = y$.

Если частичное решение $y = (y_i)$ не является двоичным вектором и $I^1(y) = \emptyset$, то полагаем $UB(y) = \infty$ и $x^0 = 0$. Далее последовательно выбираем некоторые $i \in I \setminus I^0(y)$ и полагаем $x'_i = 1$. Полагаем $x'_i = 0$ для остальных $i \in I$.

В общем случае, когда $y = (y_i)$ не двоичный вектор и $I^1(y) \neq \emptyset$, вычисление верхней границы есть итерационный процесс, состоящий из начальной итерации, некоторого числа основных итераций и, возможно, одной дополнительной итерации.

На начальной итерации обращаемся к HPR для задачи $(\mathcal{L}(y), \mathcal{F})$ и как результат формируем задачу SEP. Для этого при каждом $k \in I \setminus I^1(y)$ обращаемся к процедуре построения начального с-отсечения, сгенерированного номером k . Если отсечение построено, то добавляем соответствующее неравенство (23) к ограничениям задачи SEP, а номер k включаем в множество номеров K , генерирующих начальные с-отсечения.

На каждой основной итерации рассматриваем задачу SEP, полученную на предыдущей итерации, и вычисляем её оптимальное решение (X', z') . Если для значения $L(X', z')$ целевой функции задачи SEP на полученном решении (X', z') выполняется неравенство $L(X', z') \leq L^*$, то полагаем $UB = L(X', z')$, $x^0 = 0$ и завершаем процедуру вычисления верхней границы. В противном случае начинаем построение дополнительных ограничений, отсекающих решение (X', z') . Для этого последовательно перебираем номера $k \in K$ и выполняем процедуру построения с-отсечения. Если для некоторого k соответствующее с-отсечение построено, то полученное неравенство добавляем к ограничениям задачи SEP и начинаем следующую основную итерацию. В противном случае начинаем процесс построения d-отсечения решения (X', z') . Для этого поочерёдно просматриваем номера $k \in I$ такие, что $z'_{kl} = 1$ для некоторого $l \in T$, $l > 1$, и для выбранного k выполняем процедуру построения d-отсечения, сгенерированного номером k . Если процедура построения d-отсечения для рассматриваемого k завершается успешно, то полученное неравенство (20) добавляем к ограничениям задачи SEP и начинаем следующую основную итерацию.

Если отсечений решения (X', z') построить не удалось, и при этом $I^1(x') \neq I^1(y)$, то полагаем $UB(y) = L(X', z')$, $x^0 = 0$ и завершаем процедуру вычисления верхней границы. Если $I^1(x') = I^1(y)$, то переходим к выполнению дополнительной итерации.

На дополнительной итерации используем f-отсечение решения (X', z') . Для этого полагаем $x^0 = x'$ и вычисляем пессимистическое допустимое решение (X^0, Z^0) задачи $(\mathcal{L}(y), \mathcal{F})$, порождённое вектором x^0 . Добавляем соответствующие неравенства к ограничениям задачи SEP и вычисляем новое оптимальное решение (X', z') задачи SEP. Полагаем $UB(y) = L(X', z')$ и завершаем процедуру вычисления верхней границы.

Заметим, что, если для нового оптимального решения (X', z') получаем $x' = x^0$, то построенное допустимое решение (X^0, Z^0) есть пессимистическое оптимальное решение задачи $(\mathcal{L}(y), \mathcal{F})$. Если же $x' \neq x^0$, то $I^1(x') \neq I^1(y)$.

4. Алгоритм вычисления пессимистического оптимального решения

Для вычисления пессимистического оптимального решения исследуемой задачи $(\mathcal{L}, \mathcal{F})$ предлагается алгоритм, построенный на основе вычислительной схемы ветвей и границ со способом ветвления — поиск в глубину. Алгоритм включает начальный шаг и некоторое число основных шагов.

На начальном шаге готовятся начальные данные, необходимые для основных шагов алгоритма. На основном шаге рассматривается *упорядоченное частичное решение*, включающее частичное решение $y = (y_i)$ и вектор порядка (i_1, \dots, i_q) , $q = |I^0(y) \cup I^1(y)|$, показывающий порядок задания значений 0 и 1 элементам частичного решения. На основном шаге с использованием верхней границы проверяется, содержит ли рассматриваемое подмножество решение лучше известного рекордного решения.

На начальном шаге строим исходное рекордное решение задачи $(\mathcal{L}, \mathcal{F})$. Для этого выбираем двоичный вектор x^* и далее строим пессимистическое допустимое решение (X^*, Z^*) , порождённое вектором x^* , а также вычисляем значение $L(x^*)$ целевой функции задачи $(\mathcal{L}, \mathcal{F})$ на решении (X^*, Z^*) . На этом шаге строим также начальное упорядоченное частичное решение. Полагаем $y_i = 1$, если $x_i^* = 1$, и $y_i = *$, если $x_i^* = 0$. Далее полагаем $q = 0$ и для всякого $i \in I$ такого, что $y_i = 1$, последовательно полагаем $q = q + 1$, $i_q = i$.

На каждом основном шаге имеем упорядоченное частичное решение $y = (y_i)$, (i_1, \dots, i_q) , $q = |I^0(y) \cup I^1(y)|$. Шаг начинается с вычисления верхней границы $UB(y)$ и сопутствующих векторов x^0 и x' , $I^1(x') \neq I^1(y)$. Если $x^0 \neq 0$ и $L(x^0) > L(x^*)$, то полагаем $x^* = x^0$, $L^* = L(x^0)$. Далее строим упорядоченное частичное решение, которое будем рассматривать на следующем шаге.

Если $UB(y) \leq L^*$, то отбрасываем подмножество решений, задаваемое рассматриваемым частичным решением, и производим «движение вверх» по дереву частичных решений. Для этого находим наибольший номер k , $1 \leq k \leq q$, для которого $y_{i_k} = 1$. Если такого номера найти не удаётся, то алгоритм завершает работу, а вектор x^* порождает пессимистическое оптимальное решение (X^*, Z^*) задачи $(\mathcal{L}, \mathcal{F})$. В противном случае полагаем $y_{i_k} = 0$; $y_{i_l} = *$, $l = k + 1, \dots, q$; $q = k$ и начинаем следующий основной шаг.

Если $UB(y) > L^*$, то рассматриваемое подмножество решений отбросить не удаётся и производим «движение вниз» по дереву частичных решений. Поскольку $I^1(y) \neq I^1(x')$, для каждого $i \in I^1(x') \setminus I^1(y)$ последовательно полагаем $y_i = 1$, $q = q + 1$, $i_q = i$ и начинаем следующий основной шаг.

5. Демонстрационный пример

Материал данного раздела призван продемонстрировать особенности рассматриваемой модели конкурентного размещения, а также детали работы алгоритма поиска точного решения в рамках данной модели. Для наглядности и простоты анализа детально рассмотрим демонстрационный пример, который позволит проявиться важным эффектам, составляющим практическое воплощение теоретических результатов настоящей работы, и опишем указанные эффекты.

На рис. 1 в качестве основного примера представлена ситуация конкурентного размещения предприятий на сети из трёх вершин. Потребители и места возможного размещения предприятий располагаются в вершинах сети. Предпочтения потребителей задаются расстояниями между вершинами: чем меньше расстояние до предприятия, тем оно предпочтительней для потребителя.

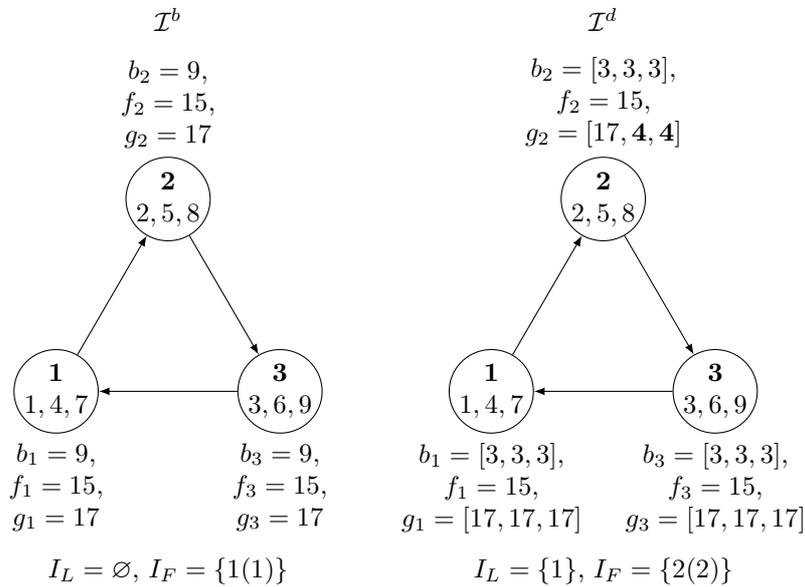


Рис. 1. Входные данные примера: слева — базовый пример \mathcal{I}^b , справа — динамический пример \mathcal{I}^d

Для демонстрации дополнительных возможностей модели динамического конкурентного размещения по отношению к базовой модели конкурентного размещения рассмотрим следующие входные данные \mathcal{I}^b для базовой модели: $I = J = \{1, 2, 3\}$, $f_i = 15$, $g_i = 17$ для всех $i \in I$, $p_{ij} = q_{ij} = 9$ для всех $i \in I, j \in J$. С учётом того, что предпочтения потребителей задаются кратчайшими расстояниями в сети, имеем $1 \succ_1 2 \succ_1 3$, $2 \succ_2 3 \succ_2 1$ и $3 \succ_3 1 \succ_3 2$. Несложно видеть, что оптимальное решение базового примера \mathcal{I}^b нулевое, поскольку общий бюджет потребителей равен 27, и открытие двух и более предприятий оказывается убыточным. При открытии единственного предприятия с номером $i_L \in I$ оптимальным ответом Последователя будет открытие предприятия с номером $i_F = i_L - 1$ при $i_L > 1$ и $i_F = 3$ при $i_L = 1$. При этом значение прибыли Лидера будет равно -6 , а следовательно, все ненулевые решения Лидера убыточны.

Предположим, что в дополнение к информации, содержащейся в примере \mathcal{I}^b , также известно, что через определённое время, равное трети продолжительности обозреваемого горизонта планирования, стоимость открытия предприятия Последователя $i = 2$ снизится и составит 4 единицы. Указанное обстоятельство можно учесть, рассмотрев динамическую модель размещения с горизонтом планирования, разделённым на три равных по длительности периода $T = \{1, 2, 3\}$. Как и ранее, положим $I = \{1, 2, 3\}$, $f_i = 15$, $g_{it} = 17$ для всех $i \in I, t \in T$ за исключением $g_{22} = g_{23} = 4$. Для каждого периода времени заведём отдельный индекс для каждого из трёх потребителей: $J_1 = \{1, 2, 3\}$, $J_2 = \{4, 5, 6\}$, $J_3 = \{7, 8, 9\}$. При этом будем считать, что потребитель, располагающийся в вершине 1, в зависимости от периода времени имеет индекс 1, 4 или 7 соответственно. Бюджет потребителей из базового примера \mathcal{I}^b разделим между периодами. Предполагая, что доход от обслуживания потребителей поступает равномерно с течением времени, и учитывая, что длительности периодов равны, положим $p_{ij} = q_{ij} = 3$ для всех $i \in I, j \in J$.

Полученный таким образом пример динамической задачи конкурентного размещения \mathcal{I}^d может рассматриваться как уточнение статического примера \mathcal{I}^b , причём указанное уточнение оказывается существенным: в отличие от примера \mathcal{I}^b у примера \mathcal{I}^d оптимальное решение ненулевое. Действительно, при $x^* = (1, 0, 0)$ несложно убедиться, что вектор z^* такой, что z_{it}^* равняется единице при $i = 2, t = 2$ и нулю в противном случае, является оптимальным решением задачи Последователя. При этом $L(x^*) = -f_1 + p_{11} + p_{21} + p_{31} + p_{41} + p_{61} + p_{71} + p_{91} = -15 + 3 \cdot 7 = 6$. Таким образом, динамическая модель расширяет возможности базовой, позволяя учитывать дополнительную информацию о параметрах среды, в которой Лидер принимает решение.

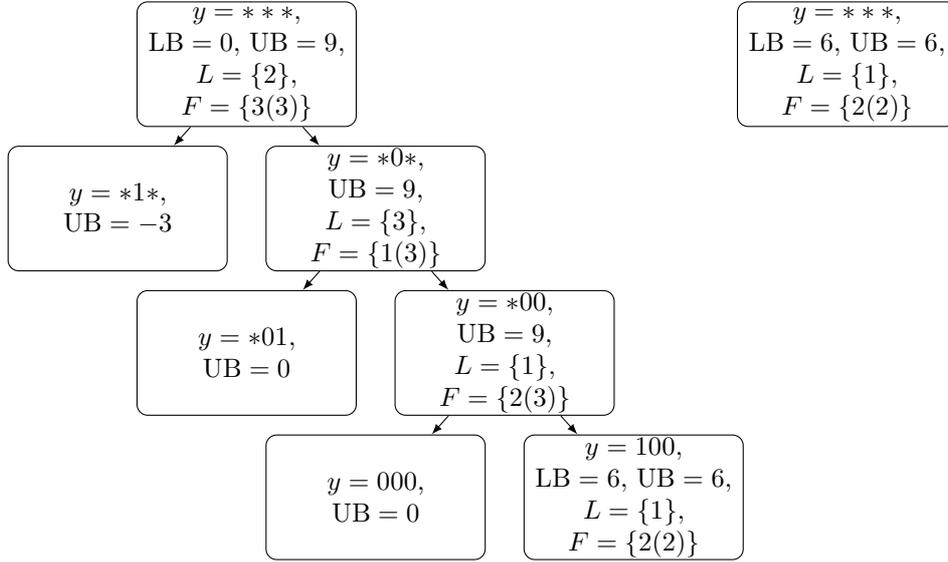


Рис. 2. Дерево ветвления: слева — при использовании только с- и f-отсечений, справа — с дополнительным использованием d-отсечений

Работа алгоритма ветвей, границ и отсечений на примере \mathcal{I}^d представлена в виде схемы на рис. 2. Поле y показывает значение компонент частичного решения в текущей вершине; LB и UB — значение нижней и верхней границ соответственно, если они вычислены; L и F — наборы открытых предприятий Лидера и Последователя соответственно, выбранных в оптимальном решении SEP (в скобках для предприятий Последователя указан момент открытия). Алгоритм запускается в двух конфигурациях: только с с- и f-отсечениями, а также дополнительно вместе с d-отсечениями.

При запуске в конфигурации без d-отсечений в корневой вершине генерируется три с-отсечения:

$$\sum_{i \in \{1,2\}} \sum_{t=1}^3 z_{it} \geq 1 + (x_3 - 1) - x_1 - x_2, \quad (c1)$$

$$\sum_{i \in \{1,3\}} \sum_{t=1}^3 z_{it} \geq 1 + (x_2 - 1) - x_1 - x_3, \quad (c2)$$

$$\sum_{i \in \{2,3\}} \sum_{t=1}^3 z_{it} \geq 1 + (x_1 - 1) - x_2 - x_3. \quad (c3)$$

При этом видим, что сгенерированные отсечения позволяют присваивать ненулевые значения переменным нижнего уровня с большим значением индекса t . Если пользоваться интерпретацией указанных переменных, то можно сказать, что в релаксации двухуровневой задачи, усиленной s -отсечениями, допустимой остаётся возможность открытия предприятий Последователя в насколько возможно поздний период времени. Это создаёт дополнительную возможность для увеличения разрыва между оптимумами исходной двухуровневой задачи и её релаксации: значение верхней границы, вычисленное в корневой вершине, оказывается равным 9 при том, что оптимум равен 6.

Во время запуска с использованием d -отсечений в корневой вершине генерируется девять отсечений: пять s -отсечений и четыре d -отсечения, имеющие вид

$$\sum_{i \in \{1,3\}} \sum_{t=1}^1 z_{it} \geq 1 + \left(\sum_{t=1}^3 z_{3t} - 1 \right) + (x_2 - 1) - x_1 - x_3, \quad (d1)$$

$$\sum_{i \in \{2,3\}} \sum_{t=1}^2 z_{it} \geq 1 + \left(\sum_{t=1}^3 z_{2t} - 1 \right) + (x_1 - 1) - x_2 - x_3, \quad (d2)$$

$$\sum_{i \in \{1\}} \sum_{t=1}^1 z_{it} \geq 1 + \left(\sum_{t=1}^3 z_{1t} - 1 \right) + (x_2 - 1) - x_1, \quad (d3)$$

$$\sum_{i \in \{3\}} \sum_{t=1}^1 z_{it} \geq 1 + \left(\sum_{t=1}^3 z_{3t} - 1 \right) + (x_1 - 1) - x_3. \quad (d4)$$

В совокупности с двумя s -отсечениями, сгенерированными в дополнение к (с1)–(с3), которые здесь приводить не будем, отсечения (d1)–(d4) позволяют найти оптимальное решение исходной двухуровневой задачи, решая её усиленную релаксацию. В частности, видим, что решение с ненулевыми компонентами $x_1 = z_{23} = 1$, возникающее как оптимальное решение релаксации в дереве ветвления с s -отсечениями в вершине с $y = *00$, отсекается неравенством (d2). Это приводит к получению более точной верхней оценки и сокращению числа вершин в дереве ветвления. При этом безусловно следует отметить увеличение времени, затрачиваемого на обработку каждой индивидуальной вершины дерева, вызванное ростом числа итераций в процедуре вычисления верхних границ. Режим использования дополнительных отсечений, позволяющий получать максимальную производительность метода ветвей, границ и отсечений в целом, является предметом для будущей исследовательской работы.

Заключение

В настоящей работе рассмотрен вариант двухуровневой задачи конкурентного размещения предприятий, получивший в литературе название неявной динамической задачи. В таком варианте предполагается, что параметры спроса меняются с течением времени дискретно, и первая сторона — Лидер — принимает решение об открытии предприятий в самом начале горизонта планирования, а вторая сторона — Последователь — может открывать свои предприятия, зная решение Лидера, в любом периоде времени. Для неявной динамической задачи вводится новое семейство дополнительных отсечений, позволяющих дополнительно усилить релаксацию двухуровневой модели. Также представлена процедура генерации отсечений из указанного семейства, вовлекающих наименьшее количество переменных и обладающих в этом смысле максимальной общностью.

Для демонстрации особенностей рассматриваемой модели и предлагаемых процедур рассмотрен пример динамической задачи конкурентного размещения на сети с тремя вершинами. Отмечены возможности модели учитывать информацию о будущих изменениях значений параметров спроса, позволяющие получать с её помощью более эффективные решения. Анализ хода вычислительного процесса поиска оптимального решения рассмотренного примера показывает востребованность использования отсечений предлагаемого в работе семейства для повышения точности верхней границы и сокращения числа вершин в дереве ветвления.

Дальнейшая работа в области изучения динамических моделей конкурентного размещения будет направлена на снятие ограничивающих предположений относительно решения Лидера. На основе алгоритма для такой более общей постановки планируется построение процедуры поддержки решений на неограниченном временном горизонте.

Финансирование работы

Исследование выполнено при финансовой поддержке Российского научного фонда (проект № 23-21-00082). Дополнительных грантов на проведение или руководство этим исследованием получено не было.

Конфликт интересов

Авторы заявляют, что у них нет конфликта интересов.

Литература

1. **Stackelberg Н.** The theory of the market economy. Oxford: Oxford Univ. Press, 1952. 289 p.

2. **Mishra M., Singh S. P., Gupta M. P.** Location of competitive facilities: A comprehensive review and future research agenda // *Benchmarking*. 2022. V. 30, No. 4. P. 1171–1230. DOI: 10.1108/BIJ-11-2021-0638.
3. **Eiselt H. A., Drezner Z.** Competitive location models: A review // *Eur. J. Oper. Res.* 2024. V. 316, No. 1. P. 5–18. DOI: 10.1016/j.ejor.2023.10.030.
4. **Santos-Penate D. R., Suarez-Vega R., Dorta-Gonzalez P.** The leader — follower location model // *Netw. Spat. Econ.* 2007. V. 7. P. 45–61. DOI: 10.1007/s11067-006-9007-2.
5. **Береснев В. Л., Мельников А. А.** Дополнительные ограничения для динамической задачи конкурентного размещения // *Дискрет. анализ и исслед. операций*. 2023. Т. 30, № 3. С. 43–56. DOI: 10.33048/daio.2023.30.774.
6. **Current J., Ratick S., ReVelle C.** Dynamic facility location when the total number of facilities is uncertain: A decision analysis approach // *Eur. J. Oper. Res.* 1998. V. 110, No. 3. P. 597–609. DOI: 10.1016/S0377-2217(97)00303-2.
7. **Dempe S.** Bilevel optimization: Theory, algorithms, applications and a bibliography // *Bilevel optimization: Advances and next challenges*. Cham: Springer, 2020. P. 581–672. DOI: 10.1007/978-3-030-52119-6_20.
8. **Beresnev V. L., Melnikov A. A.** Approximation of the competitive facility location problem with MIPs // *Comput. Oper. Res.* 2019. V. 104. P. 139–148. DOI: 10.1016/j.cor.2018.12.010.

Береснев Владимир Леонидович
Мельников Андрей Андреевич

Статья поступила
18 октября 2024 г.
После доработки —
25 октября 2024 г.
Принята к публикации
1 ноября 2024 г.

A BRANCH, BOUND, AND CUTS ALGORITHM
FOR THE DYNAMIC COMPETITIVE FACILITY
LOCATION PROBLEMV. L. Beresnev^{1,2, a} and A. A. Melnikov^{1,2, b}¹Sobolev Institute of Mathematics,
4 Acad. Koptuyug Avenue, 630090 Novosibirsk, Russia²Novosibirsk State University,
2 Pirogov Street, 630090 Novosibirsk, RussiaE-mail: ^aberesnev@math.nsc.ru, ^bmelnikov@math.nsc.ru

Abstract. We consider a dynamic competitive facility location problem modeling an interaction of two competing parties (Leader and Follower) who place their facilities within a planning horizon split into several time periods. The Leader is assumed to open his/her facilities at the beginning of the planning horizon and does not change his/her decision later, while the Follower can modify his/her choice within each time period. We propose an algorithm that computes the best Leader's decision and is built on the base of the branch-and-bound computational scheme. To compute upper bounds, a special relaxation of the initial bilevel problem strengthened with additional constraints (cuts) is used. The paper describes the construction of these constraints while utilizing auxiliary optimization problems; this provides the strongest cuts. On an instance of a dynamic competitive facility location on a network with three vertices, we demonstrate that the model is capable to take into account information regarding the changes of problem's parameters along the time period. An implementation of the branch-and-bound algorithm shows a significant benefit from using the cuts specially designed for dynamic competitive models: it improves the upper bound's quality and reduces the number of nodes in the branching tree. Illustr. 2, bibliogr. 8.

Keywords: competitive facility location problem, bilevel mathematical programming, exact method, Stackelberg game.

References

1. **H. Stackelberg**, *The Theory of the Market Economy* (Oxford Univ. Press, Oxford, 1952).
2. **M. Mishra**, **S. P. Singh**, and **M. P. Gupta**, Location of competitive facilities: A comprehensive review and future research agenda, *Benchmarking* **30** (4), 1171–1230 (2022), DOI: 10.1108/BIJ-11-2021-0638.
3. **H. A. Eiselt** and **Z. Drezner**, Competitive location models: A review, *Eur. J. Oper. Res.* **316** (1), 5–18 (2024), DOI: 10.1016/j.ejor.2023.10.030.
4. **D. R. Santos-Penate**, **R. Suarez-Vega**, and **P. Dorta-Gonzalez**, The leader — follower location model, *Netw. Spat. Econ.* **7**, 45–61 (2007), DOI: 10.1007/s11067-006-9007-2.
5. **V. L. Beresnev** and **A. A. Melnikov**, Additional constraints for dynamic competitive facility location problem, *Diskretn. Anal. Issled. Oper.* **30** (3), 43–56 (2023), DOI: 10.33048/dai.2023.30.774 [Russian] [*J. Appl. Ind. Math.* **17** (3), 483–490 (2023), DOI: 10.1134/S199047892303002X].
6. **J. Current**, **S. Ratick**, and **C. ReVelle**, Dynamic facility location when the total number of facilities is uncertain: A decision analysis approach, *Eur. J. Oper. Res.* **110** (3), 597–609 (1998), DOI: 10.1016/S0377-2217(97)00303-2.
7. **S. Dempe**, Bilevel optimization: Theory, algorithms, applications and a bibliography, in *Bilevel Optimization: Advances and Next Challenges* (Springer, Cham, 2020), pp. 581–672, DOI: 10.1007/978-3-030-52119-6_20.
8. **V. L. Beresnev** and **A. A. Melnikov**, Approximation of the competitive facility location problem with MIPs, *Comput. Oper. Res.* **104**, 139–148 (2019), DOI: 10.1016/j.cor.2018.12.010.

Vladimir L. Beresnev
Andrey A. Melnikov

Received October 18, 2024
Revised October 25, 2024
Accepted November 1, 2024