ДОКЛАДЫ РОССИЙСКОЙ АКАДЕМИИ НАУК. МАТЕМАТИКА, ИНФОРМАТИКА, ПРОЦЕССЫ УПРАВЛЕНИЯ, 2023, том 514, с. 20–25

МАТЕМАТИКА

УДК 517.9

АЛГОРИТМ ВЫЧИСЛЕНИЯ ВЕРХНИХ ГРАНИЦ ДЛЯ ЗАДАЧИ КОНКУРЕНТНОГО РАЗМЕЩЕНИЯ В УСЛОВИЯХ НЕОПРЕДЕЛЕННОСТИ СПРОСА

© 2023 г. В. Л. Береснев[1,2,*], А. А. Мельников[1,2,]**

Представлено академиком РАН В.Г. Романовым
Поступило 06.04.2023 г.
После доработки 26.09.2023 г.
Принято к публикации 14.10.2023 г.

Рассматривается математическая модель конкурентного размещения предприятий (средств обслуживания) двумя соперничающими сторонами в ситуации альтернативных сценариев реализации множества потребителей. Исследуемая задача выбора наилучших решений сторонами формулируется как дискретная задача двухуровневого математического программирования. Предлагается способ вычисления верхних границ значений целевой функции задачи на подмножествах решений для использования в алгоритмах поиска оптимального решения рассматриваемой задачи. Основу предлагаемого способа составляют построение дополнительных ограничений (отсечений) для HP-релаксации (high-point relaxation в англоязычной литературе) рассматриваемой задачи и получение в результате более сильных оценочных задач. Предложена новая процедура генерации таких ограничений, позволяющая получить наиболее сильные ограничения без использования процедур перебора при их построении.

Ключевые слова: двухуровневое программирование, игра Штакельберга, конкурентное размещение предприятий, пессимистическое оптимальное решение

DOI: 10.31857/S2686954323700327, **EDN:** ZDGOZP

1. ФОРМУЛИРОВКА ЗАДАЧИ

В [1, 2] рассматривалась математическая модель конкурентного размещения предприятий (CompFLP), построенная на основе игры Штакельберга. Разнообразие родственных постановок, а также результаты, связанные с ними, широко представлены в недавних обзорах работ [3–6]. Модель, рассматриваемая в настоящей работе, предполагает наличие двух соперничающих сторон (Лидера и Последователя), которые последовательно размещают свои предприятия для обслуживания заданного множества потребителей. Задача Лидера состоит в выборе решения, дающего максимальную прибыль при условии оптимального поведения Последователя, также стремящегося получить наибольшую прибыль от обслуживания "захваченных" потребителей.

Будем полагать, что Лидер при размещении своих предприятий не обладает полной информацией о множестве потребителей. Последователь, напротив, открывает предприятия после Лидера и имеет информацию не только о решении Лидера, но и множестве потребителей. Неопределенность данных о потребителях преодолевается рассмотрением в модели набора возможных вариантов (сценариев) множества потребителей, один из которых будет реализован. Наилучшим считается решение Лидера, дающее наибольшую прибыль при наихудших для данного решения сценарии и оптимальном решении Последователя.

Для формальной записи модели конкурентного размещения предприятий с альтернативными множествами потребителей (MSCompFLP) будут использованы следующие обозначения. Множество I включает индексы потенциальных пред-

[1]*Институт математики им. С.Л. Соболева Сибирского отделения Российской академии наук, Новосибирск, Россия*
[2]*Новосибирский государственный университет, Новосибирск, Россия*
**E-mail: beresnev@math.nsc.ru*
***E-mail: melnikov@math.nsc.ru*

приятий, открываемых Лидером и Последователем, а множество S содержит индексы рассматриваемых сценариев. Множество потребителей при сценарии $s \in S$ обозначим J_s. Предполагаем, что $J_{s_1} \cap J_{s_2} = \varnothing$ для всех $s_1, s_2 \in S$, $s_1 \neq s_2$. Это можно сделать без ограничения общности, поскольку потребителей, которых необходимо включить сразу в несколько различных сценариев, можно заменить требуемым количеством своих дубликатов. Объединенное множество потребителей из всех сценариев обозначим через $J = \bigcup_{s \in S} J_s$.

Фиксированные затраты, связанные с открытием предприятия $i \in I$ Лидером и Последователем, обозначим, соответственно, через f_i и g_i. Для заданных $i \in I$, $j \in J$ величины p_{ij} и q_{ij} выражают доход, получаемый при обслуживании потребителя $j \in J$ предприятием $i \in I$, открытым, соответственно, Лидером и Последователем. Кроме того, для всякого $s \in S$ введем величину P_s, выражающую "дополнительный" доход Лидера в случае реализации сценария $s \in S$ и используемую для корректировки влияния сценария s на решение Лидера.

В предлагаемой модели MSCompFLP мы используем переменные, аналогичные использованным в модели CompFLP. Бинарная переменная x_i равняется единице, если Лидер открывает предприятие $i \in I$ и нулю в противном случае. Бинарная переменная z_{is} равняется единице, если Последователь открывает предприятие $i \in I$ при реализации сценария $s \in S$, и нулю в противном случае. Переменные χ_{ij} и ζ_{ij} равняются единице, если открытое, соответственно, Лидером и Последователем предприятие $i \in I$ назначается для обслуживания потребителя $j \in J$, и нулю в противном случае.

При моделировании предполагается, что при назначении открытого предприятия для обслуживания потребителя $j \in J$ учитываются предпочтения этого потребителя, представленные отношением порядка $\succeq_j$ на множестве I. Для $i, k \in I$ запись $i \succ_j k$ означает $i \succeq_j k$ и $i \neq k$.

С использованием введенных обозначений задача выбора наилучшего решения в модели MSCompFLP может быть записана как следующая задача двухуровневого математического программирования:

$$\max_{(x_i),(\chi_{ij})} \left(-\sum_{i \in I} f_i x_i + \min_{s \in S} \left(P_s + \sum_{j \in J_s} \sum_{i \in I} p_{ij} \chi_{ij} \right) \right) \tag{1}$$

$$\tilde{z}_{is} + \sum_{k | i \succeq_j k} \chi_{kj} \leqslant 1, \quad s \in S, \quad j \in J_s; \tag{2}$$

$$x_i \geqslant \chi_{ij}, \quad i \in I, \quad j \in J; \tag{3}$$

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

$$(\tilde{z}_{is}), (\tilde{z}_{ij}) \text{ -- оптимальное решение задачи} \tag{5}$$

$$\max_{(z_{is}),(\zeta_{ij})} \sum_{s \in S} \left(-\sum_{i \in I} g_i z_{is} + \sum_{j \in J_s} \sum_{i \in I} q_{ij} \zeta_{ij} \right); \tag{6}$$

$$x_i + z_{is} \leqslant 1, \quad i \in I, \quad s \in S; \tag{7}$$

$$x_i + \sum_{k | i \succeq_j k} \zeta_{kj} \leqslant 1, \quad i \in I, \quad j \in J; \tag{8}$$

$$z_{is} \geqslant \zeta_{ij}, \quad i \in I, \quad s \in S, \quad j \in J_s; \tag{9}$$

$$z_{is}, \zeta_{ij} \in \{0,1\}, \quad i \in I, \quad s \in S, \quad j \in J. \tag{10}$$

Целевая функция (1) выражает величину прибыли Лидера в худшем сценарии с учетом аддитивных поправок P_s, $s \in S$ в величину дохода для каждого из сценариев. Целевая функция внутренней задачи (6) представляет суммарную прибыль Последователя по всем сценариям, и ее оптимизация эквивалентна оптимизации прибыли в каждом индивидуальном сценарии. Ограничения (2) и (8) реализуют правило выбора обслуживающего предприятия для каждого потребителя, а (3) и (9) не позволяют использовать для обслуживания закрытые предприятия. В одном месте не могут располагаться предприятия обеих конкурирующих сторон (7).

Обозначим задачу верхнего уровня (1)–(5) через $\mathscr{L}$, а задачу нижнего уровня (6)–(10) через $\mathscr{F}$. Для задачи (1)–(10) в целом используем обозначение $(\mathscr{L}, \mathscr{F})$.

Пару (X, Z), $X = ((x_i), (\chi_{ij}))$, $Z = ((z_{is}), (\zeta_{ij}))$, где X – допустимое решение задачи верхнего уровня, а Z – оптимальное решение задачи нижнего уровня, назовем *допустимым решением* задачи $(\mathscr{L}, \mathscr{F})$. Поскольку элементы решения (X, Z) определяются бинарным вектором $x = (x_i)$, то решение (X, Z) назовем допустимым решением, порожденным вектором x. Значение целевой функции задачи $(\mathscr{L}, \mathscr{F})$ на допустимом решении (X, Z) обозначим $L(X, Z)$. Для модели MSCompFLP так же, как и для модели CompFLP, ставится задача поиска наилучшего решения среди *пессимистических допустимых решений* [7]. Пессимистическое допустимое решение, порожденное вектором x, дает наименьшее значение целевой функции $L(X, Z)$ среди всех допустимых решений (X, Z), порожденных вектором x.

С задачей $(\mathscr{L}, \mathscr{F})$ связано семейство задач, получаемых из нее удалением всех сценариев, кроме одного. Задача со сценарием $s \in S$ есть задача CompFLP [2]. Обозначим эту задачу $(\mathscr{L}_s, \mathscr{F}_s)$, а значение ее целевой функции на допустимом решении (X_s, Z_s) – через $L_s(X_s, Z_s)$. Для построения пессимистического допустимого решения (X, Z) исходной задачи $(\mathcal{L}, \mathcal{F})$, порожденного заданным бинарным вектором x необходимо построить пессимистические допустимые решения (X_s, Z_s), порожденные вектором x для всякого $s \in S$. Каждое из решений (X_s, Z_s), $s \in S$ будет частью решения (X, Z) и при этом $L(X, Z) = \min_{s \in S}\{L_s(X_s, Z_s)\}$. Для построения пессимистического допустимого решения задачи $(\mathcal{L}_s, \mathcal{F}_s)$, порожденного заданным вектором x, необходимо решить две задачи целочисленного программирования, одна из которых есть задача $\mathcal{F}_s$, а другая – вспомогательная задача той же размерности, что и задача $(\mathcal{L}_s, \mathcal{F}_s)$.

Поиск оптимального пессимистического решения может быть организован по схеме неявного перебора в пространстве векторов размещения x, определяющих соответствующие им пессимистические допустимые решения. В своей работе такие схемы опираются на процедуру вычисления качественных верхних оценок для значений, принимаемых целевой функцией на подмножествах множеств допустимых решений. Разработка такой процедуры открывает возможность точного решения модели конкурентного размещения в условиях неопределенности спроса и обсуждается в последующих разделах статьи.

2. УСИЛЕННАЯ ОЦЕНОЧНАЯ ЗАДАЧА

Основой для вычисления верхней границы значений целевой функции задачи $(\mathcal{L}, \mathcal{F})$ на подмножествах решений является ее релаксация, называемая HP-релаксацией [7]. Такая задача получается из исходной двухуровневой модели (1)–(10) исключением из нее условия (5) и целевой функции задачи $\mathcal{F}$. Кроме того, из задачи $\mathcal{F}$ могут быть исключены переменные (ζ_{ij}), поскольку они не влияют на значение целевой функции полученной задачи. Таким образом, HP-релаксация исследуемой задачи $(\mathcal{L}, \mathcal{F})$ есть задача $\mathcal{L}$, дополненная ограничениями (7) и (10), а ее допустимое решение (X, z) имеет вид $X = ((x_i), (\chi_{ij}))$, $z = (z_{is})$. Значение целевой функции релаксированной задачи на решении (X, z) обозначим $L(X, z)$.

Понятно, что оптимальное решение (X, z) релаксированной задачи имеет нулевые значения переменных (z_{is}) и дает завышенную верхнюю границу для оптимального значения целевой функции исходной задачи $(\mathcal{L}, \mathcal{F})$. Общий подход к улучшению верхней границы состоит в построении дополнительных ограничений, стимулирующих переменные (z_{is}) релаксированной задачи принимать ненулевые значения. Если эти ограничения выполняются на пессимистических допустимых решениях задачи $(\mathcal{L}, \mathcal{F})$, то HP-релаксация задачи $(\mathcal{L}, \mathcal{F})$, называемая *усиленной оценочной задачей* (SEP), может быть использована для вычисления верхних границ.

Поскольку в задаче $(\mathcal{L}, \mathcal{F})$ оптимальные значения переменных (z_{is}) вычисляются независимо для различных $s \in S$, то дополнительные ограничения для задачи $(\mathcal{L}, \mathcal{F})$ можно искать в виде дополнительных ограничений для задач $(\mathcal{L}_s, \mathcal{F}_s)$ при фиксированных $s \in S$.

В [1, 2] предложены дополнительные ограничения для усиления HP-релаксации задачи CompFLP, названные c-отсечениями. При рассмотрении c-отсечений задачи $(\mathcal{L}_s, \mathcal{F}_s)$ используются следующие обозначения. Пусть $I' \subseteq I$, $J' \subseteq J_s$ – непустые подмножества, и пусть $j \in J_s$. Положим

$\alpha_j(I') = i'$, где $i' \succeq_j i$ для всех $i \in I'$;

$$\alpha_{J'}(I') = \{\alpha_j(I') \mid j \in J'\};$$

$$N_j(I') = \{i \in I \mid i \succ_j \alpha_j(I')\};$$

$$N_{J'}(I') = \cup_{j \in J'} N_j(I').$$

Если $x = (x_i)$, $i \in I$ – ненулевой бинарный вектор, то положим также $\alpha_j(x) = \alpha_j(I^1(x))$, $N_J(x) = N_J(I^1(x))$, где $I^1(x) = \{i \in I \mid x_i = 1\}$.

Пусть $I' \subseteq I$, $J' \subseteq J_s$, $k \in N_{J'}(I')$, и пусть для множества $J'_k \subseteq \{j \in J_s \mid k \in N_j(I')$ и $N_j(I') \subseteq N_{J'}(I')\}$ справедливы соотношения

$$\sum_{j \in J'_k} p_{\alpha_j(I')j} > 0, \qquad \sum_{j \in J'_k} q_{kj} \geqslant g_k.$$

Тогда неравенство

$$\sum_{i \in N_{J'}(I')} z_{is} \geqslant 1 + \sum_{i \in \alpha_{J'}(I')} (x_i - 1) - \sum_{i \in N_{J'}(I')} x_i \qquad (11)$$

выполняется на любом пессимистическом допустимом решении задачи $(\mathcal{L}_s, \mathcal{F}_s)$ и называется *c-отсечением, сгенерированным парой подмножеств (I', J') и номером $k \in N_{J'}(I')$*.

Для построения c-отсечения в [1, 2] предложена переборная процедура поиска для заданного подмножества I' требуемых подмножества J' и номера k.

Построение пары подмножеств (I', J'), генерирующей c-отсечение, существенно упрощается, если ограничиться рассмотрением пар подмножеств специального вида. Пара непустых подмножеств (I', J'), $I' \subseteq I$, $J' \subseteq J_s$ при $k \in I$ имеет *канонический вид*, если $J' \subseteq \{j \in J_s | k \in N_j(I')\}$ и $I' = \alpha_{J'}(I')$.

Утверждение 1. Пусть пара подмножеств (I', J') при $k \in N_{J'}(I')$ генерирует с-отсечение. Тогда существует пара подмножеств (I'', J'') канонического вида, которая при данном k генерирует с-отсечение. При этом $I'' \subseteq I'$, и $N_{J''}(I'') \subseteq N_{J'}(I')$.

С использованием утверждения 1 построение с-отсечения может быть упрощено. В отличие от схемы перебора подмножеств I', использованной в [2], утверждение 1 открывает путь к построению отсечений путем выбора подходящего k. Для этого достаточно построить соответствующее множество J_k' и проверить выполнимость неравенств $\sum_{j \in J_k'} p_{\alpha_j(I')j} > 0$ и $\sum_{j \in J_k'} q_{kj} \geqslant g_k$. В случае положительного результата проверки пара подмножеств канонического вида может строиться на основании эвристических правил, направленных на нахождение наиболее сильных с-отсечений. К примеру, вовлекающих минимальное количество переменных.

3. ВЫЧИСЛЕНИЕ ВЕРХНИХ ГРАНИЦ

Построение алгоритмов неявного перебора для поиска пессимистического оптимального решения задачи $(\mathcal{L}, \mathcal{F})$ предполагает вычисление верхних границ значений целевой функции задачи на рассматриваемых в ходе работы алгоритма подмножествах решений. Такие подмножества задаются частичными решениями $y = (y_i)$, $y_i \in \{0, 1, *\}$, $i \in I$, фиксирующими значения некоторых переменных x_i, $i \in I$. Пусть $I^0(y) = \{i \in I | y_i = 0\}$, $I^1(y) = \{i \in I | y_i = 1\}$. Тогда элементами подмножества решений задачи $(\mathcal{L}, \mathcal{F})$, заданного частичным решением $y = (y_i)$, будут пессимистические допустимые решения, порождаемые векторами $x = (x_i)$, $i \in I$, такими, что

$$x_i = y_i, \quad i \in I^0(y) \cup I^1(y). \tag{12}$$

Задачи $(\mathcal{L}, \mathcal{F})$ и $(\mathcal{L}_s, \mathcal{F}_s)$ с указанными дополнительными условиями (12) в задаче верхнего уровня обозначим, соответственно, через $(\mathcal{L}(y), \mathcal{F})$ и $(\mathcal{L}_s(y), \mathcal{F}_s)$, $s \in S$.

Если y есть бинарный вектор, то верхняя граница $UB(y)$ целевой функции задачи $(\mathcal{L}(y), \mathcal{F})$ вычисляется точно путем нахождения значения целевой функции задачи $(\mathcal{L}, \mathcal{F})$ на пессимистическом допустимом решении, порожденном вектором y.

В общем случае, когда y не является бинарным вектором, процедура вычисления верхней границы есть итерационный процесс, состоящий из начальной итерации и некоторого числа однотипных основных итераций.

На начальной итерации строим задачу SEP для задачи $(\mathcal{L}(y), \mathcal{F})$. Для этого к ограничениям HP-релаксации задачи $(\mathcal{L}, \mathcal{F})$ добавляем ограничение (12) и известные к данному моменту дополнительные с-отсечения для задачи $(\mathcal{L}_s, \mathcal{F}_s)$, $s \in S$.

На основной итерации рассматриваем построенную на предыдущей итерации задачу SEP для задачи $(\mathcal{L}(y), \mathcal{F})$ и вычисляем ее оптимальное решение (X', z'), $X' = ((x_i'), (\chi_{ij}'))$, $z' = (z_{is}')$, $i \in I$, $s \in S$. Далее для всякого $s \in S$ рассматриваем решение (X_s', z_s'), которое является частью решения (X', z') и допустимым решением подзадачи SEP, соответствующей задаче $(\mathcal{L}_s, \mathcal{F}_s)$. После этого для задачи $(\mathcal{L}_s, \mathcal{F}_s)$ строим с-отсечения, которые нарушаются на решении (X_s', z_s'). Такие отсечения назовем *с-отсечениями решения* (X_s', z_s').

Пусть $x' = (x_i')$, $z_{s'} = (z_{is}')$ и пусть $J_{0s} = J_s$, если $z_s' = 0$, и $J_{0s} = \{j \in J_s | \alpha_j(x') \succ_j \alpha_j(z_s')\}$ в противном случае.

Утверждение 2. Неравенство (11), сгенерированное парой (I', J'), нарушается на решении (X_s', z_s') тогда и только тогда, когда $J' \subseteq J_{0s}$ и $I^1(x') \supseteq I' \supseteq \alpha_{J'}(x')$.

Таким образом, построение с-отсечения решения (X_s', z_s') сводится к поиску подмножества J' множества $\{j \in J_{0s} | k \in N_j(x')\}$ такого, что пара (I', J'), где $I' = \alpha_{J'}(x')$, генерирует с-отсечение.

Утверждение 3. При заданном $k \in I$ множество $J' \subseteq J_{0s}(k)$ такое, что пара (I', J'), где $I' = \alpha_{J'}(x')$, генерирует с-отсечение, существует тогда и только тогда, когда

$$\sum_{j \in J_{0s}(k)} p_{\alpha_j(x')j} > 0, \qquad \sum_{j \in J_{0s}(k)} q_{kj} \geqslant g_k.$$

Если для заданного $k \in N_{J_{0s}}(x')$ указанный критерий существования множества J' выполняется, то возникает задача выбора элементов множества $\{j \in J_0 s | k \in N_j(x')\}$, из которых будет составлено J'. Этот выбор должен быть продиктован стремлением к тому, чтобы пара (I', J'), где $I' = \alpha_{J'}(x')$, генерировала наиболее сильное с-отсечение решения (X_s', z_s'). Для построения такого множества можно сформулировать вспомогательную оптимизационную задачу, решение которой будет определять множество J' с наимень-

Таблица 1. Зависимость времени вычислений и числа просмотренных вершин в методе ветвей и границ (среднее по 5 примерам) от количества мест возможного размещения предприятий

$\|I\|$	Время, с	Вершины
5	1	3
10	28	15
15	86	4
20	387	46
25	4322	55

шим суммарным числом элементов во множествах $\alpha_{J'}(x')$ и $N_{J'}(x')$. Построенное таким образом с-отсечение назовем *с-отсечением решения* (X'_s, z'_s)*, сгенерированным номером* $k \in N_{J_{0s}}(x')$.

Суммируя сказанное, получим следующую схему вычисления на каждой основной итерации алгоритма вычисления верхней границы $UB(y)$. После вычисления оптимального решения (X', z') текущей задачи SEP для задачи $(\mathcal{L}(y), \mathcal{F})$ последовательно для всякого $s \in S$ исследуем элементы $k \in N_{J_{0s}}(x')$ с целью построения с-отсечений решения (X'_s, z'_s), сгенерированных элементом k. Для этого для заданного k проверяем выполнение критерия существования требуемого с-отсечения. Если критерий выполняется, то вычисляем соответствующее множество J' и получаем требуемое с-отсечение. Это неравенство добавляем к ограничениям рассматриваемой задачи SEP. Если в процессе просмотра элементов $s \in S$ и $k \in N_{J_{0s}}(x')$ удалось построить хотя бы одно с-отсечение, то начинаем следующую итерацию. В противном случае величину верхней границы $UB(y)$ полагаем равной оптимальному значению $L(X', z')$ целевой функции задачи SEP и завершаем процедуру вычисления верхней границы.

Для оценки эффективности указанной процедуры были сгенерированы тестовые примеры, в которых расположение предприятий и потребителей задавалось случайно размещенными на единичном квадрате точками. Предпочтения потребителей определяются евклидовыми расстояниями между точками. Сценарии спроса в примерах упорядочены по убыванию потенциальной доходности, и в каждом последующем сценарии рассматривается лишь часть множества потребителей предыдущего.

В табл. 1 представлены средние по 5 примерам каждой размерности показатели времени вычисления и числа просмотренных вершин для метода ветвей и границ, использующего описанный способ подсчета верхних границ. Отметим, что рост размерности задачи сопровождается лишь незначительным ростом числа просматриваемых вершин. Это, по-видимому, объясняется способностью алгоритма быстро обнаруживать оптимальное решение, а также особенностью модели: в силу существенных отличий между сценариями спроса, число решений, способных хорошо себя проявлять во всех сценариях, крайне ограничено, и отклонения от оптимального решения приводят к ухудшению верхних оценок, позволяющему на начальных этапах ветвления определить подобные отклонения как бесперспективные. При этом с ростом размерности задачи увеличивается время решения задач SEP в вершинах дерева ветвления, а также число генерируемых с-отсечений, что приводит к быстрому росту общего времени работы. Несмотря на указанное обстоятельство, алгоритм показывает высокую эффективность, позволяя точно решать задачи с 25 местами возможного размещения предприятий со средним временем работы менее двух часов.

ИСТОЧНИКИ ФИНАНСИРОВАНИЯ

Исследование выполнено за счет гранта Российского научного фонда (проект № 21-41-09017).

СПИСОК ЛИТЕРАТУРЫ

1. *Береснев В.Л., Мельников А.А.* Алгоритм генерации отсечений для дискретной задачи конкурентного размещения предприятий. Доклады Академии наук. 2018. V. 480. № 5. P. 515–518. https://doi.org/10.1134/S1064562418030183
2. *Beresnev V., Melnikov A.* Approximation of the competitive facility location problem with MIPs. *Computers & Operations Research*. 2019. V. 104. P. 139–148, https://doi.org/10.1016/j.cor.2018.12.010
3. *Ashtiani M.* Competitive location: A state-of-art review. International Journal of Industrial Engineering Computations. 2016. V. 7. № 1. P. 1–18. https://doi.org/10.5267/j.ijiec.2015.8.002
4. *Aras N., Küçükaydın H.* Bilevel Models on the Competitive Facility Location Problem. In: Mallozzi L., D'Amato E., Pardalos P. (eds) Spatial Interaction Models. Springer Optimization and Its Applications, vol 118. Springer, Cham. 2017. P. 1–19. https://doi.org/10.1007/978-3-319-52654-6_1
5. *Karakitsiou A.* Modeling discrete competitive facility location. Springer Cham, 2015, SpringerBriefs in Optimization, 54.
6. *Mishra M., Singh S.P., Gupta M.P.* Location of competitive facilities: a comprehensive review and future research agenda. Benchmarking, 2022. https://doi.org/10.1108/BIJ-11-2021-0638
7. *Dempe S.* Bilevel Optimization: Theory, Algorithms, Applications and a Bibliography, In: S. Dempe, A. Zemkoho (eds) Bilevel Optimization: Advances and Next Challenges, Springer International Publishing, Cham. 2020. P. 581–672. https://doi.org/10.1007/978-3-030-52119-6_20

UPPER BOUND FOR THE COMPETITIVE FACILITY LOCATION PROBLEM WITH DEMAND UNCERTAINTY

V. Beresnev[a,b] and A. Melnikov[a,b]

[a]*Sobolev Institute of Mathematics, Novosibirsk, Russian Federation*

[b]*Novosibirsk State University, Novosibirsk, Russian Federation*

Presented by Acadimician of the RAS V.G. Romanov

We consider a competitive facility location problem with two competing parties operating in a situation of uncertain demand scenario. The problem to find the best solutions for the parties is formulated as a discrete bi-level mathematical programming problem. In the paper, we suggest a procedure to compute an upper bound for the objective function on subsets. The procedure could be employed in implicit enumeration schemes capable to compute an optimal solution for the problem under study. Within the procedure, additional constraints iteratively augment the high-point relaxation of the initial bi-level problem, what strengthens the relaxation and improves the upper bound's quality. New procedure to generate such cuts allows to construct the strongest cuts without enumerating the parameters encoding them.

Keywords: Bi-level programming, Stackelberg game, competitive facility location, pessimistic optimal