

Эвристики для идентификации 1-парашютов в задаче аппроксимации графа

И.В. Уразова
urazovainn@mail.ru

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация

Задача аппроксимации графа заключается в нахождении в полном реберно-взвешенном графе семейства попарно вершинно-непересекающихся клик минимального веса. В общем случае задача является NP -трудной. В работе [1] был предложен класс неравенств, опорных к многограннику данной задачи. Найдены условия, при которых построенные неравенства являются фасетными. При использовании этих неравенств в алгоритмах отсечений, становится актуальной задача идентификации (Separation problem). В работе [2] было показано, что задача идентификации предложенных неравенств NP -трудна. В настоящей работе для идентификации отсечений разработана процедура локального поиска. Для анализа эффективности предлагаемых методов проведен вычислительный эксперимент.

Работа выполнена при поддержке РФФИ (проект 18-07-00599)

1 Основные понятия и факты

Впервые задача аппроксимации графа была сформулирована Харари в 1955 году. В 60-70-х годах прошлого столетия в ряде работ были найдены нетривиальные классы графов, на которых задача является полиномиально разрешимой. В 1986 г. Криванек и Моравек рассмотрели задачу аппроксимации графа как частный случай задачи кластеризации деревьев и доказали, что она является NP -трудной. Систематическое изучение задачи аппроксимации графа началось в прошлом десятилетии, когда задача была переоткрыта под разными именами (Correlation clustering, Cluster editing) различными группами авторов. В частности была установлена NP -трудность различных ее вариантов и для их решения предложены первые приближенные алгоритмы с гарантированной оценкой точности. Лучший из известных на сегодняшний день приближенный алгоритм для задачи аппроксимации графа гарантировано находит решение не более чем в 2.5 раза хуже оптимального.

Введем следующие обозначения и понятия. Для любого графа $D \subset K_n$ через VD и ED будем обозначать множества его вершин и ребер соответственно. Для ребра $e \in E$ будем также использовать запись uv , где u и v — вершины из V , инцидентные ребру e . Для $D \subseteq K_n$ и $u \in V$ через $\delta_D(u)$ обозначим множество ребер графа D , инцидентных вершине u . Если $D = K_n$, то в этой записи индекс D будем опускать. Каждое множество ребер $R \subset E$ индуцирует в K_n некоторый подграф T , в котром $ET = R$ и VT — множество вершин,

Copyright © by the paper's authors. Copying permitted for private and academic purposes.

In: Sergey V. Belim, Nadezda F. Bogachenko (eds.): Proceedings of the Workshop on Data, Modeling and Security 2018 (DMS-2018), Omsk, Russia, October 2018, published at <http://ceur-ws.org>

инцидентных ребрам из R . Там, где не возникает двусмысленности, граф, индуцированный множеством ребер R , будем обозначать через R . Для подграфов D, F и K_n положим

$$D \cup F = ED \cup EF, D \cap F = ED \cap EF.$$

Пусть $\mathbf{K}_n = (V, E)$ — полный неориентированный граф без петель и кратных ребер. Остовный подграф $H \subset \mathbf{K}_n$ называется M -графом, если каждая его компонента связности является кликой или одновершинным графом. Множество всех M -графов в $\mathbf{K}_n$ обозначим через $\mu(V)$.

Пусть $G \subset \mathbf{K}_n$ некоторый априори заданный остовный подграф. Задача аппроксимации графа G заключается в нахождении M -графа H , минимизирующего на множестве $\mu(V)$ функционал

$$\rho_G(H) = |G \cup H| - |G \cap H|.$$

Иными словами, требуется найти такое множество попарно непересекающихся клик на V , которое как можно меньше (в реберном смысле) отличается от графа G .

В работе рассматривается полиэдральная постановка задачи аппроксимации графа. Полиэдральный подход к решению экстремальных комбинаторных задач заключается в сопоставлении задаче специального многогранника, заданного как выпуклая оболочка векторов инцидентий допустимых решений, и, как следствие, использование аппарата выпуклого анализа и целочисленного программирования. Особое место на этом пути занимает задача описания многогранника в виде множества решений системы линейных уравнений и неравенств. При наличии полного линейного описания многогранника экстремальная комбинаторная задача сводится к задаче линейного программирования (возможно с экспоненциальным числом ограничений), что нередко позволяет получить эффективные алгоритмы ее решения.

Множество $P \subset R^E$ называется многогранником, если оно является выпуклой оболочкой конечного числа точек. Линейное неравенство $a^T x \leq a_0$ ($a, x \in R^E, a \neq 0, a_0 \in R, "T"$ — знак транспонирования) называется правильным к многограннику P , оно не нарушается ни одной из течек многогранника. Правильное неравенство называется опорным к многограннику P , если оно выполняется для любой точки из P и существует, по крайней мере, одна точка из P , обращающая его в равенство. Всякое опорное к P неравенство порождает множество $\{x \in P | a^T x = a_0\}$, которое называется гранью многогранника P . Максимальные по включению грани многогранника называются фасетами. Ясно, что фасетой является та и только та грань, размерность которой на 1 меньше размерности самого многогранника. Опорное неравенство, порождающее фасету, назовем, соответственно, фасетным.

Полиэдром в R^E называется множество решений конечной системы линейных уравнений и неравенств с переменными $x_e, e \in E$, если оно ограничено. Согласно теореме Вейля-Минковского для всякого многогранника существует совпадающий с ним в теоретико-множественном смысле полиэдр и наоборот.

В [1, 4] было показано, что $(0, 1)$ -вектор $x \in R^E$ является вектором инцидентий M -графа, если и только если он удовлетворяет системе

$$\begin{aligned} -x_{uv} + x_{uw} + x_{vw} &\leq 1 \\ x_{uv} - x_{uw} + x_{vw} &\leq 1 \\ x_{uv} + x_{uw} - x_{vw} &\leq 1 \end{aligned} \quad (3)$$

где $u, v, w \in V$ — всевозможные тройки попарно различных вершин,

$$x_{uv} \geq 0, \text{ для всех } uv \in E \quad (4)$$

Полиэдр, определяемый системой (3)-(4) обозначим через M_n . Таким образом, $P_n \subset M_n$, причем включение именно стогое (в [1] приведены примеры нецелочисленных вершин полиэдра M_n) и каждая целочисленная вершина M_n является вектором инцидентий некоторого M -графа. Следуя работе [3], ограничения вида (3) будем называть треугольниками.

В [1] был введен и изучен новый класс опорных к P_n неравенств. Пусть $U = \{u_1, u_2, \dots, u_k\}$ и $W = \{v_1, v_2, \dots, v_p\}$ — непустые подмножества множества V , $U \cap W = \emptyset, k \geq 1, p \geq 2$. Через $T_i, i = 1, 2, \dots, k$, обозначим звезду в K_n с центром в вершине u_i и лучами $u_i v_j, j = 1, 2, \dots, p$ (см. рис. 1).

Через K_p обозначим клику на множестве вершин W . Положим $T = \bigcup_{i=1}^k T_i$. Граф $T \cup K_p$ называется k -парашютом (см. Рис. 2).

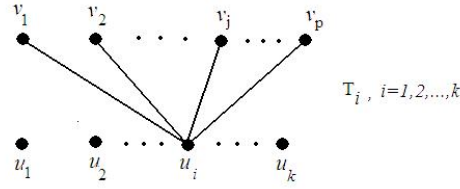


Рис. 1: Звезда в K_n с центром в вершине u_i и лучами $u_i v_j, j = 1, 2, \dots, p$

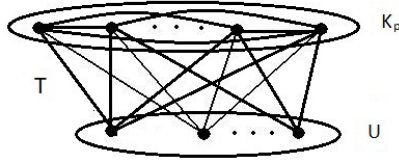


Рис. 2: k -парашют

С этим графом свяжем неравенство

$$x(ET) - x(EK_p) \leq \frac{k^2 + k}{2}.$$

В [1] было доказано, что такое неравенство, индуцированное k -парашютом $T \cup K_p$, является опорным к многограннику P_n тогда и только тогда, когда $p \geq k$, а фасетным — тогда и только тогда, когда $k = 1$. В этой связи особый интерес для нас будут представлять именно 1-парашюты (см. Рис. 3).

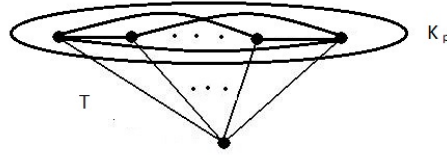


Рис. 3: 1-парашют.

В [3, 4] был предложен класс опорных неравенств (2-дольных), отличающихся от k -парашютов тем, что на множестве $U = \{u_1, u_2, \dots, u_k\}$ строится клика K_2 (см. Рис. 4). Класс 2-дольных неравенств состоит из неравенств вида

$$x(ET) - x(EK_p \cup EK_2) \leq \min(p, k).$$

Кроме того, нетрудно заметить, что пересечением класса 2-дольных неравенств и класса k -парашютов является класс 1-парашютов.

Для заданного 1-парашюта $T \cup K_p$ вершину графа K_n , образующую множество U , будем иногда называть парашютистом, а клику K_p — куполом 1- парашюта.

При разработке процедур отсекающих, использующих опорные неравенства как отсекающие плоскости, на передний план выходит задача идентификации (separation problem), которая заключается в следующем. Даны некоторый класс L неравенств, опорных к многограннику P , и точка $\bar{x} \in R^E$. Требуется найти в классе L неравенство, строго отделяющее точку $\bar{x}$ от многогранника P , либо доказать, что в L такого неравенства нет. В 1982 году в работах Karp R.M. и Papadimitriou C.H. было показано, что при $NP \neq co-NP$ для полиэдра общего вида (не обязательно целочисленного) и класса неравенств, определяющих все фасеты выпуклой оболочки его целочисленных точек, не существует полиномиального алгоритма решения задачи идентификации. В связи с этим целесообразно рассматривать задачу идентификации применительно

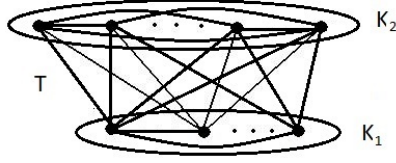


Рис. 4: 2-дольное неравенство

к конкретным классам неравенств относительно конкретных задач. Например, идентификация кликовых фасет для многогранника симметричной задачи коммивояжера полиномиально разрешима, а идентификация гребневых неравенств и неравенств деревьев клик для той же задачи NP-трудна (см. также [2, 7]).

Очевидно, что поскольку число попарно различных троек вершин в K_n полиномиально по n , задача идентификации треугольников полиномиально разрешима. В [3] для поиска 2-дольного неравенства, отсекающего заданную точку, использовалась эвристическая процедура, поскольку сложностной статус задачи идентификации 2-дольных неравенств на тот момент был не известен. В 2001 году была показана NP-трудность задачи идентификации для этого класса. Однако важно иметь в виду следующее. Поскольку многогранник P_n целиком лежит в единичном кубе пространства R^E , то отсекаемую точку можно полагать лежащей в единичном кубе. Это несколько усиливает постановку задачи идентификации. Возможно еще большее усиление. Поиск неравенства, отсекающего текущий нецелочисленный оптимум, начинается, как правило, с класса неравенств с полиномиально разрешимой задачей идентификации. Поэтому постановка задачи идентификации для класса 2-дольных неравенств может быть усилена до следующей: найдется ли 2-дольное неравенство, отсекающее точку, удовлетворяющую всем треугольникам? В [2] дано новое строгое доказательство этого факта для класса 1-парашютов и, как следствие, 2-дольных неравенств и k -парашютов.

2 Идентификация 1-парашютов

Сформулируем задачу идентификации 1-парашютов относительно многогранника M -графов. При заданной точке $\bar{x} \in R^E$, $0 \leq \bar{x} \leq 1$, среди всех 1-парашютов $T \cup K_p$ графа K_n требуется найти такой, для которого $\sum_{e \in ET} \bar{x}_e - \sum_{e \in EK_p} \bar{x}_e > 1$. Эта задача может быть формализована в виде задачи целочисленного линейного программирования. Определим булевы переменные: $x_u \in \{0, 1\}$, $u \in V$, – выбор парашютиста; $y_u \in \{0, 1\}$, $u \in V$, – выбор вершин купола; $z_{uv} \in \{0, 1\}$, $u, v \in V$, – выбор ребер купола; $t_{uv} \in \{0, 1\}$, $u, v \in V$, – выбор ребер между парашютистом и куполом. Требуется максимизировать функцию

$$\frac{1}{2} \left(\sum_{u, v \in V} \bar{x}_{uv} t_{uv} - \sum_{u, v \in V} \bar{x}_{uv} z_{uv} \right) \quad (5)$$

при условиях

$$z_{uv} \geq y_u + y_v - 1, \quad u, v \in V; \quad (6)$$

$$z_{uv} \leq y_u, \quad z_{uv} \leq y_v, \quad u, v \in V; \quad (7)$$

$$t_{uv} \geq x_u + y_v - 1, \quad u, v \in V; \quad (8)$$

$$t_{uv} \leq x_u, \quad t_{uv} \leq y_v, \quad u, v \in V; \quad (9)$$

$$x_u + y_u \leq 1, \quad u \in V; \quad (10)$$

$$\sum_{u \in V} x_u = 1; \quad (11)$$

$$\sum_{u \in V} y_u \geq 1; \quad (12)$$

$$x_u, y_u, z_{uv}, t_{uv} \in \{0, 1\}, \quad u, v \in V. \quad (13)$$

Ограничения (6), (7) определяют ребра внутри купола и фактически гарантируют равенство $z_{uv} = y_u y_v$. Ограничения (8), (9) определяют ребра между куполом и парашютистом и гарантируют равенство

$t_{uv} = x_u y_v$. Ограничения (10) запрещают пересечение парашютиста с куполом. Ограничение (11) требует выбора ровно одного парашютиста. Ограничение (12) гарантирует непустой купол. В действительности для решения задачи идентификации достаточно проверить на совместность систему, образованную ограничениями (6)-(13) и неравенством $\sum_{u,v \in V} \bar{x}_{uv} t_{uv} - \sum_{u,v \in V} \bar{x}_{uv} z_{uv} > 1$, что в целом не облегчает задачу.

Как уже говорилось, задачу идентификации целесообразно рассматривать применительно к конкретным классам неравенств относительно конкретных задач. При такой стратегии помимо точных процедур решения часто используются эвристики (см., например, [3]). Так как задача идентификации 1-парашютов NP-трудна, мы разработали эвристическую процедуру для ее решения.

Итак, рассматривается следующая задача. Дана точка $\bar{x} \in M_n \setminus P_n$. В графе K_n нужно найти такой 1-парашют $T_u \cup K_p$, где K_p , $p \geq 3$, – клика на множестве вершин $\{v_1, v_2, \dots, v_p\}$ и T_u – звезда с центром в вершине $u \notin \{v_1, v_2, \dots, v_p\}$ и лучами uv_j , $j = 1, 2, \dots, p$, что выполняется неравенство

$$\bar{x}(ET_u) - \bar{x}(EK_p) > 1.$$

Полагая величины $\bar{x}_e$ весами ребер $e \in E$, а величину $\bar{x}(ET_u) - \bar{x}(EK_p)$ – весом 1-парашюта $T_u \cup K_p$, мы можем сформулировать рассматриваемую задачу идентификации, как задачу поиска в полном реберно-взвешенном графе 1-парашюта с весом больше 1.

Всякий 1-парашют $T_u \cup K_p$ является кликой порядка $p + 1$. В связи с этим, зафиксировав какое-либо множество вершин, мы можем простым перебором найти на этом множестве самый тяжелый 1-парашют. Формально эта процедура выглядит так.

Процедура $\gamma(W)$. Эта процедура просматривает 1-парашюты на фиксированном множестве вершин $W = \{u_1, u_2, \dots, u_l\} \subset V$, $l > 3$. Положим $\gamma(W) = 0$, T и K – графы с пустыми множествами вершин и ребер.

Шаг i ($i = 1, 2, \dots, l$). Пусть T_i – звезда с центром в вершине u_i и лучами $u_i u_j$, $j = 1, \dots, i-1, i+1, \dots, l$, и K_i – клика на вершинах $u_1, \dots, u_{i-1}, u_{i+1}, \dots, u_l$. Если $\bar{x}(ET_i) - \bar{x}(EK_i) > \gamma(W)$, то полагаем $\gamma(W) := \bar{x}(ET_i) - \bar{x}(EK_i)$, $T = T_i$ и $K = K_i$. Иначе оставляем все неизменным. Переходим на шаг $(i + 1)$.

Если в результате этой процедуры получим $\gamma(W) > 1$, то $T \cup K$ – искомый 1-парашют на множестве вершин W . В процедуре $\gamma(W)$ не обязательно просматривать все вершины $u_1, u_2, \dots, u_l$. Можно остановиться сразу, как только $\gamma(W) > 1$. Однако если размерность задачи относительно невелика, то просмотр лучше довести до конца, так как величину $\bar{x}(ET) - \bar{x}(EK) > \gamma$ можно в определенном смысле считать "глубиной" отсечения.

Вообще говоря, в результате применения процедуры γ к конкретному множеству W может получиться $\gamma(W) \leq 1$. В этой связи имеет смысл перейти к другому W и повторить процедуру. Мы рассматриваем три способа перехода к новому множеству вершин.

1) Множество W' получается из W удалением одной вершины $u \in W$, то есть $W' \subset W$ и $|W'| = |W| - 1$. Такой переход будем обозначать $W' = \varphi_1(W, u)$.

2) Множество W' получается из W добавлением одной вершины $u' \notin W$, то есть $W \subset W'$ и $|W'| = |W| + 1$. Такой переход будем обозначать $W' = \varphi_2(W, u')$.

3) Множество W' получается из W удалением одной вершины $u \in W$ и добавлением одной вершины $u' \notin W$, то есть $|W'| = |W|$ и $|W \cup W'| - |W \cap W'| = 2$. Для обозначения такого перехода будем использовать запись $W' = \varphi_3(W, u, u')$.

К случайно выбранному множеству W применяется каждый из указанных способов в рандомизированном режиме. А именно, переход φ_1 осуществляется последовательным просмотром вершин множества W , вершина $u \in W$ выбирается с вероятностью p_1 , запоминается такое W_1 , что $\gamma(W_1) = \max\{\gamma(\varphi_1(W, u)), u \in W\}$; переход φ_2 – просмотр множества $V \setminus W$, вершина u' выбирается с вероятностью p_2 , запоминается такое W_2 , что $\gamma(W_2) = \max\{\gamma(\varphi_2(W, u')), u' \notin W\}$; переход φ_3 заключается в просмотре множества $W \times (V \setminus W)$, пара (u, u') выбирается с вероятностью p_3 , запоминается W_3 , для которого $\gamma(W_3) = \max\{\gamma(\varphi_3(W, u, u')), u \in W, u' \notin W\}$. В качестве нового множества вершин W' выбирается то из множеств W_1 , W_2 и W_3 , на котором значение функции $\gamma(W)$ наибольшее.

Описанная процедура является одной итерацией эвристики, используемой нами для решения задачи идентификации 1-парашютов. Обозначим через $\varphi(W)$ множество вершин, которое получается из множества W в результате этой итерации. При построении отсекающих 1-парашютов итерации продолжаютсся до тех пор, пока не прекратится заметный рост функции γ . На рисунках 4 и 5 показан пример работы описанной процедуры для задач с $n = 100$.

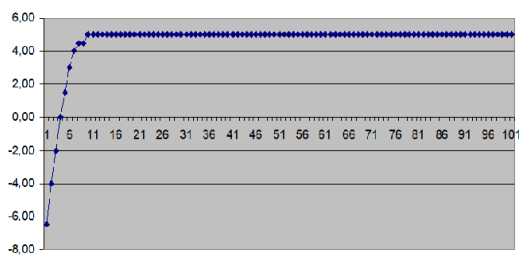


Рис. 5: Число вершин в клике 1-парашюта $|VK| = 15$, итераций 100, наибольшее значение $\gamma(W)=5$

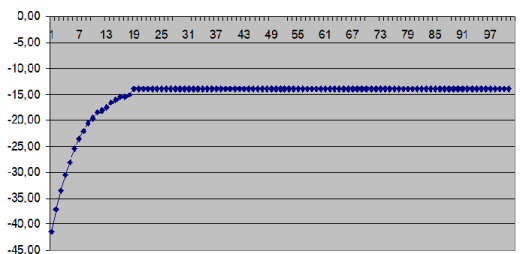


Рис. 6: Число вершин в клике 1-парашюта $|VK| = 30$, итераций 100, наибольшее значение $\gamma(W)=-14$

В следующем параграфе показаны результаты вычислительного эксперимента, цель которого заключалась в проверке эффективности предложенной процедуры решения задачи идентификации для 1-парашютов. Следует отметить, что анализ результатов вычислительного эксперимента существенно осложняется отсутствием на сегодняшний день тестовых задач.

3 Вычислительный эксперимент

Алгоритм локального поиска 1-парашюта был реализован на языке Java и запускался на компьютере Intel Pentium (2,4 GHz). Сам эксперимент проводился по трем направлениям и был организован следующим образом. На первом этапе исследуется эвристика для построения 1-парашютов при одинаковых значениях параметров p_1 , p_2 и p_3 . Время было фиксированным и составляло 10 мин. Оценивалось значение целевой функции, полученное на заданных параметрах за 10 мин. Целью этого этапа было выделить при каких параметрах рост значений целевой функции будет быстрее. В эксперименте рассматривались задачи размерности от 20 до 100, всего было по 10 задач на каждую размерность. Ниже представлен фрагмент таблицы 1 с результатами эксперимента. В первом столбце указано имя файла с решаемой задачей, в последнем – количество ребер аппроксимируемого графа, в остальных столбцах стоят значения целевой функции, полученные за 10 мин при заданной вероятности.

Задача	p1=p2=p3					EG
	0,25	0,40	0,50	0,60	0,75	
f1_40	216,64	216,84	216,89	218,49	218,23	385
f2_40	223,81	225,28	223,54	222,70	225,35	394
f3_40	227,33	225,82	225,86	228,16	226,55	400
f4_40	217,30	218,20	218,27	219,05	219,42	379
f5_40	227,38	225,50	225,14	225,78	227,38	400
f1_50	185,78	180,30	178,89	179,01	190,46	313
f2_50	177,15	175,05	174,32	177,01	177,19	306

Рис. 7: Таблица 1

Из таблицы 1 видно, что при выбранной вероятности 0.6 0.75 рост значения целевой функции за фиксированное время происходит быстрее.

Часто, найденных с помощью локального поиска неравенств 1-парашютов оказывается очень много, и добавление их всех в качестве отсечений может привести к быстрому росту размерности задачи и, как следствие, увеличению времени на каждой итерации. Целью данного эксперимента было выяснить, сколько лучше добавить найденных 1-парашютов, чтобы рост значений функции происходил быстрее. Вероятности были взяты одиноковыми и равными, а именно $p_1 = p_2 = p_3 = 0.6$. Из всех найденных 1-парашютов

выбирались неравенства с наибольшим значением $|a^T \bar{x} - 1|$. Здесь $\bar{x}$ – решение задачи ЛП, полученное на очередной итерации, $a^T x \leq 1$ – неравенство 1-парашюта, найденное с помощью эвристики. Время работы программы было фиксированным и составляло 10 минут. На каждой итерации добавлялись найденные неравенства 1-парашютов в количестве 1, $n/10$, $n/5$ и $n/2$, где n – это число вершин исходного графа G . В эксперименте рассматривались задачи размерности от 20 до 100, всего было по 10 задач на каждую размерность. Ниже представлен фрагмент таблицы 2 с результатами эксперимента.

Задача	q=1	n/10		n/5		n/2		EG
	f	ax-1	f	ax-1	f	ax-1	f	
f1 20	56,90	1,05	56,89	1,06	56,91	1,06	56,92	89
f2 20	62,88	1,063	62,89	1,07	62,90	1,07	62,90	106
f4 20	58,65	1,0578	58,66	1,09	58,64	1,06	58,65	99
f5 20	57,84	1,08	57,81	1,05	57,85	1,06	57,84	92
f1 30	131,00	1,17	139,02	1,21	144,50	1,11	146,00	232
f2 30	126,28	1,31	129,97	1,29	131,58	1,14	135,23	200
f3 30	137,06	1,2659	139,17	1,23	142,58	1,11	145,19	228
f4 30	132,85	1,32	135,95	1,32	139,39	1,18	141,75	213
f5 30	135,19	1,31	138,32	1,29	140,99	1,12	145,30	220
f1 40	210,00	2,50	216,30	2,50	217,53	2,68	222,38	385
f2 40	221,07	2,50	227,50	2,50	228,45	2,21	234,69	394
f3 40	221,75	2,50	225,81	2,50	230,39	2,00	234,15	400
f4 40	211,06	2,50	217,27	2,50	220,41	2,19	225,44	379
f5 40	218,92	2,50	220,00	2,50	229,15	2,24	232,18	400
f1 50	174,93	3,89	173,99	4,88	175,42	2,50	177,04	313
f2 50	171,49	4,62	172,98	4,50	172,24	4,00	173,56	306
f3 50	180,50	3,50	178,70	2,50	181,25	3,50	183,58	325
f4 50	173,84	4,58	175,96	3,47	177,81	4,79	177,72	312

Рис. 8: Таблица 2

Из таблицы 2 видно, что если на каждой итерации добавлять $n/2$ найденных 1-парашютов, то за заданное время рост значений функции происходил быстрее.

Последний эксперимент повторяет предыдущий за исключением одного шага. Вместо значения $|a^T \bar{x} - 1|$ рассматривалась величина $d = \frac{|a^T \bar{x} - 1|}{|a|}$. Результаты представлены в таблице 3.

Задача	q=1	n/10		n/5		n/2		EG
	f	d	f	d	f	d	f	
f1 20	56,91	0,02	56,91	0,02	56,91	0,02	56,90	89
f2 20	62,90	0,02	62,90	0,04	62,90	0,03	62,90	106
f3 20	71,00	0,08	71,00	0,08	71,00	0,03	71,00	88
f4 20	58,61	0,03	58,64	0,03	58,69	0,03	58,66	99
f5 20	57,82	0,03	57,83	0,03	57,84	0,02	57,83	92
f1 30	141,99	0,04	145,74	0,03	146,00	0,06	146,00	232
f2 30	130,04	0,07	134,57	0,02	135,66	0,02	135,94	200
f3 30	139,78	0,05	144,38	0,02	145,62	0,02	145,68	228
f4 30	135,76	0,10	140,86	0,02	142,25	0,02	142,46	213
f5 30	137,06	0,07	141,55	0,05	144,31	0,02	145,49	220
f1 40	214,27	0,30	222,32	0,30	225,04	0,40	231,78	385
f2 40	225,68	0,30	229,33	0,29	240,59	0,15	244,23	394
f3 40	225,25	0,36	232,12	0,21	236,52	0,15	241,29	400
f4 40	216,18	0,32	224,61	0,26	230,36	0,21	234,73	379
f5 40	222,00	0,32	229,82	0,27	234,86	0,14	240,82	400
f1 50	182,18	0,45	186,14	0,37	185,23	0,43	186,43	313

Рис. 9: Таблица 3

Из таблицы 3 видно, что если выбирать найденные 1-парашюты по значению $d = \frac{|a^T \bar{x} - 1|}{|a|}$ при параметрах $p_1 = p_2 = p_3 = 0.6$ и $n/2$, то значения целевой функции выше, чем при $|a^T \bar{x} - 1|$.

Заключение

В заключении автор выражает благодарность Р.Ю. Симанчеву и Ю.А. Кочетову за ценные идеи и замечания в работе над данной статьей.

Список литературы

- [1] Simanchev R.Yu., Urazova I.V. On the polytope faces of the graph approximation problem *Journal of Applied and Industrial Mathematics*, April 2015, Volume 9, Issue 2, pp 283–291.

- [2] Simanchev R.Yu., Urazova I.V. Separation Problem for k-parashutes *Proc. DOOR 2016, Vladivostok, Russia, September 19-23, CEUR-WS*, 1623:109–114, 2016. URL: <http://ceur-ws.org/Vol-1623/paperco16.pdf>
- [3] Grotschel M., Wakabayashi Y. A cutting plane algorithm for a clustering problem *Mathematical Programming*, (Series B) 45:59–96. 1989.
- [4] Grotschel M., Wakabayashi Y. Facets of the clique partitioning polytope *Mathematical Programming*, 47:367–387, 1990.
- [5] Simanchev R.Yu., Urazova I.V. Cutting Planes Algorithm for the Connected k-factor Problem Using the Facet Inequalities. *Petrovac, Montenegro, October 2-7, CEUR-WS*, 1987, 2017. URL: <http://ceur-ws.org/Vol-1987/paper74.pdf>
- [6] Schrijver A. Combinatorial Optimization. Polyhedra and Efficiency. *Springer, Vol. A. 2003*
- [7] Simanchev R.Yu. On facet-inducing inequalities for combinatorial polytopes *Journal of Applied and Industrial Mathematics*, October 2017, Volume 11, Issue 4, pp 564–571.

Heuristics for the Separation Problem of 1-Parachutes in the Graph Approximation Problem

Inna Vl. Urazova

In work [1] a class of inequalities supporting to the polyhedron of the given problem was proposed. Conditions are found under which the constructed inequalities are faceted. When these inequalities are used in cutting algorithms, the Separation problem becomes an actual problem. In the paper [2] it was shown that the problem of identifying the proposed inequalities is *NP*-hard. In this paper, a local search procedure has been developed to identify clipping. To analysis the effectiveness of the proposed methods, a computer experiment was carried out.