

О СХЕМЕ ЛОКАЛЬНОГО ПОИСКА ДЛЯ ЗАДАЧИ МАКСИМИЗАЦИИ РАЗРЕЗА, ПОРОЖДЕННОГО МИНИМАЛЬНЫМ ДОМИНИРУЮЩИМ МНОЖЕСТВОМ

А.А. Тоцкий

аспирант, e-mail: TotckiiAA@omsu.ru

Р.Ю. Симанчёв

д.ф.-м.н., профессор, e-mail: simanchevriu@omsu.ru

И.В. Уразова

к.ф.-м.н., доцент, e-mail: urazovaiv@omsu.ru

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

Аннотация. В статье описывается схема локального поиска для задачи максимизации разреза, порождённого минимальным доминирующим множеством. Приведены результаты численного эксперимента.

Ключевые слова: граф, доминирующее множество, локальный поиск, максимизация разреза.

Введение

Широко известна задача о поиске в простом неориентированном графе доминирующего множества минимального размера. Она приведена как задача ТГ 2 в [1, с. 235]. Задача является достаточно широко изученной NP-полной задачей. Ей и её различным вариациям посвящено множество публикаций [2, 3]. Мы рассмотрим её зеркальный (и обобщённый) вариант.

В настоящей работе рассматривается следующая задача. Дан обыкновенный граф $G = (V, E)$ с множеством вершин V и множеством рёбер E . Рёбра графа G будем обозначать как неупорядоченную пару вершин ij , где $i, j \in V$. Каждому ребру $ij \in E$ сопоставлен положительный вес c_{ij} . Подмножество $W \subseteq V$ называется доминирующим, если для любой вершины $j \in V \setminus W$ найдётся такая вершина $i \in W$, что $ij \in E$. Семейство всех доминирующих множеств в графе G обозначим через $D(G)$. Доминирующее множество называется минимальным (по включению), если оно не содержит в себе доминирующих множеств меньшего размера. Соответственно семейство всех минимальных доминирующих множеств будем обозначать через $D_{\min}(G)$. Всякое подмножество $W \subset V$ порождает в графе G разрез $(W, \bar{W}) = \{ij \in E \mid i \in W, j \in \bar{W}\}$, где $\bar{W} = V \setminus W$. Весом разреза назовём сумму весов входящих в него рёбер $c(W, \bar{W}) = \sum_{ij \in (W, \bar{W})} c_{ij}$. Среди всех минимальных доминирующих множеств требуется найти такое, которое порождает наибольший по весу разрез, т. е.

$$\max\{c(W, \bar{W}) \mid W \in D_{\min}(G)\}. \quad (1)$$

Эта задача возникает из большого числа прикладных вопросов системного анализа, когда для мониторинга системы требуется найти насколько возможно малое количество ключевых признаков, имеющих при этом наибольшее влияние на состояние и поведение системы в целом (см., например, [4–6]). В работе [6] для этой задачи предложено обозначение MAX MDS CUT. Там же доказывалась принадлежность этой задачи классу NP-трудных, обосновывается её полиномиальная разрешимость для деревьев и двудольных графов. Кроме того, в [6] показывается, что в случае $P \neq NP$ не существует приближённых полиномиальных алгоритмов её решения с гарантированной оценкой точности $|V|^{\varepsilon - \frac{1}{2}}$ для любого $\varepsilon > 0$.

Помимо приведённых результатов по этой задаче, отметим, что в [7] построена модель целочисленного линейного программирования для поиска точного решения задачи MAX MDS CUT в случае ориентированного графа G . Эта модель естественным образом переносится на неориентированный граф, правда, с существенным увеличением числа переменных.

В настоящей работе мы предлагаем схему локального поиска для нахождения приближенного решения задачи (1).

1. Схема локального поиска

Процедура локального поиска предполагает два основных этапа: генерирование начального минимального доминирующего множества и правило перехода к соседнему множеству (окрестность решения). Вообще говоря, множество всех вершин графа G является доминирующим множеством, но не обязательно минимальным по включению.

Для вершины $i \in V$ введём обозначение $N(i) = \{j \in V \setminus \{i\} \mid ij \in E\}$ – множество вершин, смежных с i .

Отметим несколько простых свойств доминирующих множеств, которые лежат в основе предлагаемых процедур:

Свойство 1. Множество $W \subset V$ является доминирующим тогда и только тогда, когда для всякой вершины $j \in \bar{W}$ имеет место $W \cap N(j) \neq \emptyset$.

Свойство 2. Если $W \in D(G)$, то для любого $j \in \bar{W}$ имеем $W \cup \{j\} \in D(G)$. Как следствие, $W \cup U \in D(G)$ для любого $U \subseteq \bar{W}$.

Свойство 3. Доминирующее множество $W \in D_{min}(G)$ является минимальным по включению тогда и только тогда, когда для каждой $i \in W$ имеем $W \setminus \{i\} \notin D(G)$.

Действительно, если $W \in D_{min}(G)$, то требуемое следует из определения, так как $W \setminus \{i\} \subset W$ при $i \in W$. Обратно, пусть $W \in D(G)$ и $W \setminus \{i\} \notin D(G)$ при любом $i \in W$. Возьмём произвольно $U \subset W$ и покажем, что $U \notin D(G)$. Пусть $i \in W \setminus U$. Тогда $U \subseteq W \setminus \{i\} \subset W$, и, по условию, $W \setminus \{i\} \notin D(G)$. Значит, найдётся вершина $j \in V \setminus (W \setminus \{i\})$, которая не достижима из $W \setminus \{i\}$ за один шаг. Но тогда она не достижима и из U , так как $U \subseteq W \setminus \{i\}$. Следовательно, $U \notin D(G)$.

Начальное решение генерируется следующим образом:

Процедура MD

Полагаем, $V = \{i_1, i_2, \dots, i_n\}$ – произвольное упорядочение множества вершин графа G ; $W = V$.

ШАГ 0. Полагаем $W := V, s = 1$.

ШАГ s . При $s < n$. Если $W \setminus \{i_s\} \in D(G)W$, то полагаем $W := W \setminus \{i_s\}$ и переходим на шаг $s + 1$. Если $W \setminus \{i_s\} \notin D(G)W$, то множество W не меняем и переходим на шаг $s + 1$. Если $s = n$, то конец: W – минимальное доминирующее множество в графе G .

Проверка на доминируемость осуществляется на основании свойства 1. Минимальность полученного в итоге доминирующего множества следует из свойства 3.

Переход к соседнему минимальному доминирующему множеству осуществляем посредством Процедуры NMD, которая заключается в следующем. Пусть $W \in D_{min}(G)$. Выбираем случайным образом множество $U \subset \bar{W}$, $|U| = q$, $q = f * |\bar{W}|$, $0 < f < 1$, и используем процедуру MD применительно к множеству $W := W \cup U$. В результате получим $W' \in D_{min}(G)$, которое назовём соседом множества W . Основанием для такой конструкции является свойство 2.

Теперь схема локального поиска может быть представлена так:

Процедура LSMD

Шаг 0. Выбираем максимальное количество просматриваемых соседей t и величину f .

Шаг 1. С помощью MD строим начальное множество $W \in D_{min}(G)$.

Шаг 2. С помощью NMD просматриваем не более t соседей множества W , выбирая множество U случайным образом. Если $c(W, \bar{W}) \leq c(W', \bar{W}')$, то полагаем $W := W'$ и повторяем шаг 2. Если за t просмотров множество с большим весом не найдено, то среди просмотренных соседей выбираем W' с наибольшим весом $c(W', \bar{W}')$, полагаем $W := W'$ и повторяем шаг 2.

Замечания.

1. Целесообразно применение процедуры LSMD для различных начальных решений (см. процедуру MD). Новое начальное решение легко получить, изменив порядок вершин графа G .
2. Просмотр соседей на шаге 2 также можно производить, изменяя упорядочение вершин в $W \cup U$.
3. Величину f , задающую мощность $q = f * |\bar{W}|$ множества U и количество просматриваемых соседей t , также можно считать пусковым параметром наряду с начальным решением.

Ситуацией, достаточной для окончания работы алгоритма, можно считать:

- стабильный затухающий рост весов получаемых минимальных доминирующих множеств;
- априорное ограничение времени работы;
- априорное ограничение числа полученных множеств W .

В качестве приближённого решения выбирается множество W с наибольшим весом порождённого им разреза.

2. Вычислительный эксперимент

С помощью построенного алгоритма был проведён численный эксперимент на наборах случайно сгенерированных задач, содержавших по десять графов с 10, 20, 30 и 40 вершин, 20 %-й заполненностью и целочисленными весами рёбер. Точные

решения для графов с количеством вершин менее 50 были получены решением задачи ЦЛП [7] с помощью демонстрационной версии пакета IBM ILOG CPLEX Optimization Studio 12.10. Получение с помощью этого пакета точных решений для графов с 50 вершинами за разумное время уже представляется невозможным.

Для приближённого алгоритма в качестве критерия остановки было выбрано число s переходов к соседнему минимальному доминирующему множеству и заданное равным 30 для всех задач. Время работы точного и приближённого алгоритмов t_{app} и t_{opt} указано в секундах, относительная погрешность – в процентах. f_{app} и f_{opt} – веса разрезов, найденных приближённым и точным алгоритмами соответственно. $\epsilon = \frac{f_{opt} - f_{app}}{f_{opt}}$ – относительная погрешность, указана в процентах (табл. 1).

Таблица 1. Результаты вычислительного эксперимента

№	n	f_{opt}	f_{app}	$\epsilon, \%$	t_{app}	t_{opt}	№	n	f_{opt}	f_{app}	$\epsilon, \%$	t_{app}	t_{opt}
0	30	3918	3586	8,4	9	4	0	40	6183	4887	20,9	17	189
1	30	4014	3620	9,8	12	3	1	40	6148	5662	7,9	18	84
2	30	3625	3456	4,6	7	2	2	40	5827	5554	4,7	17	228
3	30	3808	3501	8,1	11	3	3	40	6810	5308	22,1	16	154
4	30	3628	3590	1,1	8	9	4	40	6329	6069	4,1	15	246
5	30	4071	3646	10,4	6	11	5	40	5628	5362	4,7	15	141
6	30	3777	3682	2,5	7	12	6	40	6016	5139	14,5	15	101
7	30	3900	3434	11,9	7	2	7	40	5828	5130	11,9	14	114
8	30	3678	3488	5,1	6	12	8	40	6916	6043	12,6	17	102
9	30	3564	3296	7,5	6	11	9	40	6304	6082	3,5	15	108
av:	30	-	-	7	8	7	-	40	-	-	11	16	147

На наборе задач с десятью вершинами в семи случаях из десяти решение, найденное приближённым алгоритмом, совпадало с точным.

Заключение

Для NP-трудной задачи минимизации в графе разреза, порождённого минимальным доминирующим множеством, составлен алгоритм локального поиска. Окрестность задана с помощью определения процедуры перехода от одного минимального доминирующего множества к соседнему. На случайных данных произведён численный эксперимент, приведены его результаты.

Дальнейшим развитием данной работы может стать внесение случайности в выбор окрестности, когда после определённого количества просмотров соседних доминирующих множеств с некоторой вероятностью происходит переход к другой процедуре перехода, т. е. меняется тип окрестности.

Благодарности

Работа выполнена в рамках государственного задания Омского научного центра СО РАН (номер госрегистрации проекта 121022000112-2).

Литература

1. Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи. М.: Мир, 1982. 420 с.
2. Boria N., Croce F. D., Paschosdef V. T. On the max min vertex cover problem // Discrete Applied Mathematics. 2015. Т. 196. Р. 62–71.
3. Lee J., Nagarajan V., Shen X. Max-Cut under Graph Constraints. 2015.
4. Агалаков С.А., Симанчев Р.Ю., Уразова И.В. Об одном подходе к построению системы ключевых показателей экономической безопасности // Вестник Омского университета. Серия «Экономика». 2018. № 4. С. 5–12.
5. Simanchev R. Yu., Urazova I. V., Voroshilov V. V. The minimal dominating sets in a directed graph and the key indicators set of socio-economic system // Ural Mathematical Journal. 2023. Vol. 9, No. 1. Р. 153–161.
6. Ворошилов В. В. Сложность задачи о максимальном разрезе с условием минимального доминирования // Дискретный анализ и исследование операций. 2022. Т. 29, № 1. С. 5–17.
7. Ворошилов В. В. Разрез наибольшего веса в орграфе, порождённый минимальным доминирующим множеством // Дискретный анализ и исследование операций. 2020. Т. 27, № 4. С. 5–20.

ON LOCAL SEARCH SCHEME FOR A PROBLEM OF MAXIMIZATION OF DICUT, GENERATED BY MINIMAL DOMINATING SET

A.A. Totskij

PhD student, e-mail: TotckiiAA@omsu.ru

R.Y. Simanchev

Dr.Sc. (Phys.-Math.), Professor, e-mail: simanchevriu@omsu.ru

I.V. Urazova

Ph.D. (Phys.-Math.), Associate Professor, e-mail: urazovaiv@omsu.ru

Dostoevsky Omsk State University, Omsk, Russia

Abstract. The article describes local search scheme for a problem of maximization of dicut, generated by minimal dominating set. Results of a numerical experiment are presented.

Keywords: graph, dominating set, local search, cut maximization.

Дата поступления в редакцию: 12.07.2025