ОмГУ, кафедра ПОЗИ

Схема ветвей и отсечений для задачи агрегации бинарных отношений

Р. Ю. Симанчев, И.В.Уразова

Омск
2020

Задача разбиения на клики (CPP)

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

$$c : E \to R.$$

Задача разбиения на клики (clique partitioning problem, CPP) заключается в нахождении M-графа, минимизирующего функционал c на множестве $\mu(V)$.

Задача разбиения на клики (CPP)

С графом K_n свяжем евклидово пространство R^E размерности $\frac{n^2-n}{2}$, поставив в соответствие каждому ребру ось координат в R^E. Это пространство может рассматриваться как пространство вектор-столбцов с компонентами, индексированными элементами из E. Если $x \in R^E$ и $S \subseteq E$, то через $x(S)$ обозначим линейную форму $\sum_{e \in S} x_e$.

Задача разбиения на клики (CPP)

Вектором инциденций произвольного остовного графа $D \subseteq K_n$ называется вектор $x^D \in R^E$ с компонентами $x_e^D = 1$ при $e \in ED$ и $x_e^D = 0$ при $e \notin ED$. Последнее правило, очевидно, задает взаимно однозначное соответствие между множеством всех остовных пографов графа K_n и множеством вершин единичного куба в R^E. В этих терминах мы определим многогранник M-графов или, что то же, многогранник задачи разбиения на клики как

$$P_n = conv\{x^H \in R^E | H \in \mu(V)\},$$

а саму задачу разбиения на клики, как

$$\min\{\rho(x) = c^T x \mid x \in P_n\}.$$

Задача разбиения на клики (CPP)

$(0,1)$-вектор $x \in R^E$ является вектором инциденций M-графа, если и только если он удовлетворяет системе

$$-x_{uv} + x_{uw} + x_{vw} \leq 1$$

$$x_{uv} - x_{uw} + x_{vw} \leq 1$$

$$x_{uv} + x_{uw} - x_{vw} \leq 1$$

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

$$x_{uv} \geq 0, \text{ для всех } uv \in E.$$

Полиэдр, определяемый системой обозначим через M_n.

Агрегация бинарных отношений (ABRP)

При заданных множестве n объектов V, p характеристиках и $(n \times p)$-матрице $D = (d_{ik})$, где d_{ik} представляет свойство объекта i по отношению к характеристике k, найти "лучшее" разбиение набора объектов на непересекающиеся классы "однородных" объектов.

$$(i, j) \in R_k \Leftrightarrow d_{ik} = d_{jk}.$$

Агрегация бинарных отношений (ABRP)

Каждая характеристика трактуется как бинарное отношение эквивалентности R_k на множестве V. Соответстсвенно, под "лучшим" понимается такое отношение эквивалентности R^* на V, которое минимизирует величину

$$\sum_{k=1}^{p} |R \triangle R_k|$$

, где

$$R \triangle R_k = \{(i,j) \mid (i,j) \in R \text{ и } (i,j) \notin R_k, \text{ или } (i,j) \notin R \text{ и } (i,j) \in R_k\}$$

.

Всякое отношение эквивалентности на множестве объектов можно понимать как M-граф $H \subset K_n$. Отсюда получается полиэдральная постановка задачи агрегации бинарных отношений

$$\min\{\rho_D(x) = c_0 + \sum_{e \in E} c_e x_e \mid x \in P_n\},$$

где константа c_0 и веса c_e определяются равенствами

$$c_0 = \sum_{e \in E} |\{k \in \{1, \dots, p\} \mid e \in EH_k\}|,$$

$$c_e = p - 2|\{k \in \{1, \dots, p\} \mid e \in EH_k\}|.$$

Задача является *NP*-трудной (1086). В 1989 получены точные решения большого числа реальных прикладных задач типа ABRP. Среди них максимальными по размерности стали задачи, данные для которых взяты из голосований государств-членов ООН по резолюциям, принятым на 39 Генеральной ассамблее в 1984 году по вопросам в области прав человека ($n = 158$, $p = 3$), отношения к ядерному оружию ($n = 158$, $p = 15$), помощи палестинским беженцам на Ближнем Востоке ($n = 158$, $p = 9$). Были также рассмотрены задачи со случайными данными, но в них число объектов не превышало 34, а число характеристик не превышало 13.

Алгоритм *BFC*

0-я (начальная) итерация.

Эта итерация заключается в применении алгоритма *FC* при $Q = B^E$ в течение времени, не превышающем T. На выходе этой 0-й итерации мы получим одну из следующих альтернатив:
1) "ОПТИМАЛЬНЫЙ *M*-ГРАФ" (см. шаг 2 алгоритма *FC*),
2) "α-ПРИБЛИЖЕННЫЙ *M*-ГРАФ" (см. шаг 4 алгоритма *FC*),
3) "ЗАДАЧА НЕ РЕШЕНА" (см. шаг 5 алгоритма *FC*),
4) время работы алгоритма *FC* достигло величины T.
В первых двух случаях алгоритм заканчивает свою работу. В третьем и четвертом случаях переходим на 1-ю итерацию алгоритма *BFC*.

Алгоритм ветвей и отсечений

Перед k-й итерацией алгоритма BFC имеем массив Ψ и рекорд (rec, x^*).

k-я итерация.

Шаг 1 (выбор очередного узла и ветвление). В массиве Ψ выбираем строку $(B, \bar{x}, \mu)$ с минимальным значением μ. Если $\bar{x}$ – нецелочисленная точка, то пусть e – имя любой ее дробной координаты, если $\bar{x}$ – целочисленная, то пусть e – имя любой ее координаты, которая в строке $(B, \bar{x}, \mu)$ массива Ψ не фиксирована. Формируем две новых грани

$$B' = \{x \in B \mid x_e = \lfloor \bar{x}_e \rfloor\} \text{ и } B'' = \{x \in B \mid x_e = \lfloor \bar{x}_e \rfloor + 1(\mathrm{mod} 2)\}.$$

Переходим на шаг 2.

Алгоритм ветвей и отсечений

Шаг 2 (нижняя оценка для B'). Применяем алгоритм FC' к задаче $\min\{c^T x \mid x \in Q\}$ при $Q = B'$. Если на некоторой FC'-итерации получаем ответ "α-ПРИБЛИЖЕННЫЙ M-ГРАФ" , то алгоритм BFC заканчивает работу. Если получен ответ "ЗАДАЧА НЕ РЕШЕНА" или время работы алгоритма FC' достигло величины T, то строку (B', x', μ), где x' – текущее решение, полученное алгоритмом FC', вносим в массив Ψ; если получен ответ "МНОЖЕСТВО ДОПУСТИМЫХ РЕШЕНИЙ ПУСТО" , то – не вносим. Переходим на шаг 3.

Шаг 3 (нижняя оценка для B''). Этот шаг полностью повторяет шаг 2 с той разницей, что $Q = B''$ и осуществляется переход на шаг 4.

Шаг 4 (фильтрация массива Ψ). Сравниваем строки массива Ψ по столбцу оценок с рекордом и удаляем из него те строки, для которых $\mu \geq rec$. Переходим на $k+1$-ю итерацию.

Алгоритм ветвей и отсечений

Шаг 3 (нижняя оценка для B''). Этот шаг полностью повторяет шаг 2 с той разницей, что $Q = B''$ и осуществляется переход на шаг 4.

Шаг 4 (фильтрация массива Ψ). Сравниваем строки массива Ψ по столбцу оценок с рекордом и удаляем из него те строки, для которых $\mu \geq rec$. Переходим на $k+1$-ю итерацию.

Решение $\bar{x}$ будем называть α-приближенным, если $\rho(\bar{x}) \leq (1+\alpha)\rho(x^{opt})$, где x^{opt} – оптимальное решение задачи, α – неотрицателный скаляр. В частности, если $\alpha = 0$, то α-приближенное решение является точным.

Таблица 1: ABRP со случайными данными

n	f_{opt}	Время (сек.)	Число итераций
160	-7175	1033	5
160	-7732	1399	7
200	-7174	1971	5
200	-8572	1525	7
300	-7175	1756	5
300	-3209	7210	19

Таблица 2: α-приближенное решение GAP

задача	плотность	α	FC (сек.)	BFC (сек.)	α-прибл. решение
$f5-30$	0,51	0,00	1595,42	1271,09	152
$f1-40$	0,49	0.05	–	1540,13	330
$f2-40$	0,51	0.05	–	1600,08	327
$f3-40$	0,51	0.05	–	1671,02	330
$f4-40$	0,49	0.05	–	1586,45	315
$f5-40$	0,51	0.05	–	1652,34	321
$f1-50$	0,26	0.10	–	3534,12	267
$f2-50$	0,25	0.10	–	2976,31	263
$f3-50$	0,27	0.10	–	4985,07	279
$f4-50$	0,25	0.10	–	3109,43	269
$f5-50$	0,27	0.10	–	3988,16	276
$f1-60$	0,25	0.15	–	9720,38	398
$f2-60$	0,26	0.15	–	10681,14	396
$f3-60$	0,26	0.15	–	9307,46	407
$f4-60$	0,26	0.15	–	10142,09	407
$f5-60$	0,24	0.15	–	9223,27	366
$f1-70$	0,27	0.15	–	13980,09	565
$f2-70$	0,25	0.15	–	13320,54	548
$f3-70$	0,26	0.15	–	14400,05	562
$f4-70$	0,24	0.15	–	11,520,31	517
$f5-70$	0,24	0.15	–	12180,24	519

Таблица 3: Соотношение ± 1 в GAP.

n	± 1	f_{opt}	Время (сек.)	FC	BFC
50	0,95	71	4	0	3
50	0,85	132	8	0	5
50	0,7	375	35	2	16
50	0,6	–			
50	0,3	–			
50	0,2	-242	569	7	344
50	0,15	174	868	21	350
50	0,1	111	1896	54	419
50	0,05	64	379	10	351
60	0,95	92	12	0	4
60	0,85	202	26	0	9
60	0,7	537	59	0	17
60	0,6	–			
60	0,2	–			
60	0,15	256	882	3	396
60	0,1	160	798	7	355
60	0,05	83	2840	31	220
70	0,95	161	20	0	4
70	0,85	338	73	0	13
70	0,7	788	347	0	26
70	0,6	–			

Симанчев Р.Ю., Уразова И.В., Кочетов Ю.А. Метод ветвей и отсечений для задачи разбиения на клики // Дискретный анализ и исследование операций. — 2019. — Т. 26. — №3. — С. 60–87.

(Simanchev R.Yu., Urazova I.V., Kochetov Yu.A. The branch and cut method for the clique partitioning problem, Journal of Applied and Industrial Mathematics, 2019)