

АЛГОРИТМЫ И ПРИЛОЖЕНИЯ ЗАДАЧИ О НАИБОЛЬШЕЙ КЛИКЕ ГРАФА

Илларионов Р.Е.,

научный руководитель д-р физ. мат. наук Быкова В.В.

Сибирский федеральный университет

Задача о наибольшей клике (Maximum Clique Problem, MCP) является одной из известных NP-трудных задач теории графов, для которой пока не найдено алгоритмов, разрешающих эту задачу за полиномиальное время. Между тем, данная задача имеет многочисленные приложения. В биоинформатике MCP используется при компьютерном анализе геномных баз данных, например при поиске потенциальных регуляторных структур рибонуклеиновых кислот. В социальных сетях MCP применяется при кластеризации данных – при разделении различных сообществ на группы (кластеры), обладающие общими свойствами. Выделение кластеров позволяет обрабатывать каждый из них отдельным вспомогательным сервером. В химии задача MCP лежит в основе поиска «максимальной общей подструктуры» в графе, описывающем структуру химического соединения. Кроме того, MCP является математической моделью ряда задач, возникающих при автоматизации проектирования радиоэлектронной аппаратуры.

В указанных приложениях, как правило, необходимы точные решения для MCP. При этом объем входных данных огромный (входные графы могут содержать до миллиона вершин). Таким образом, актуальным направлением исследования MCP является разработка новых подходов нахождения точных решений с учетом особенностей графов, возникающих в приложениях. В настоящем докладе предлагается модификация известного классического алгоритма Уилфа, позволяющего находить решение задачи MCP для разреженных графов за приемлемое время.

Пусть $G = (V, E)$ – неориентированный конечный граф с множеством вершин V и множеством ребер E , $|V| \geq 1$ и $|E| \geq 0$. Множество всех вершин графа G , смежных с некоторой вершиной $x \in V$, образует в G окрестность вершины x , которая обозначается через $N(x)$. Множество вершин $V' \subseteq V$ называется кликой графа G , если в графе $G(V')$ всякие две вершины смежны, и максимальной кликой, если она не содержится в клике с большим количеством вершин. Размер наибольшей (по числу вершин) клики графа G обозначается $\varphi(G)$ и называется плотностью (или кликовым числом) графа G . Граф $G = (V, E)$, являющийся кликой, называется полным графом и обозначается через K_n , где $n = |V|$. Для него $\varphi(K_n) = |V|$. В общем случае для графа $G = (V, E)$ всегда $1 \leq \varphi(G) \leq |V|$. Подмножество $V' \subseteq V$ полагается независимым в графе $G = (V, E)$, если подграф $G(V')$ не содержит ребер. Наибольшее по мощности независимое множество называется наибольшим. Число вершин в наибольшем независимом множестве – это число независимости графа G , которое обозначается через $\alpha_0(G)$. Число независимости графа G и плотность дополнительного графа \bar{G} связаны между собой формулами: $\alpha_0(G) = \varphi(\bar{G})$, $\varphi(G) = \alpha_0(\bar{G})$.

Распознавательный вариант задачи MCP можно сформулировать следующим образом.

УСЛОВИЕ. Заданы граф $G = (V, E)$ и положительное число $K \leq |V|$.

ВОПРОС. Верно ли, что G содержит клику размера не менее K ? Иными словами, существует ли подмножество $V' \subseteq V$, такое, что $|V'| \geq K$ и любые две вершины в V' соединены ребром из E ?

Легко показать, что необходимым и достаточным условием для существования в G клики размера K является наличие независимого множества размера не менее K в дополнительном графе. Таким образом, зная точное решение задачи о независимом множестве вершин, можно указать точное решение задачи МСР, и наоборот.

Наиболее известными алгоритмами нахождения точного решения задачи МСР являются алгоритм Брона–Кербоша и алгоритм Уилфа. Алгоритм Брона–Кербоша является рекурсивной процедурой, которая наращивает кандидатскую клику, рассматривая на каждом шаге одну вершину. Алгоритм добавляет эту вершину в клику или включает ее в набор исключенных вершин – вершин, которые не могут быть в клике. Показано, что время работы алгоритма Брона–Кербоша составляет $O(\text{poly}(n) \cdot 3^{n/3}) = O(\text{poly}(n) \cdot 1,4422^n)$, где $\text{poly}(n)$ – некоторый полином от $n = |V|$. Алгоритм Брона–Кербоша многократно совершенствовался. Самая быстрая модификация данного алгоритма находит точное решение МСР за время $O(\text{poly}(n) \cdot 2^{0,249n}) = O(\text{poly}(n) \cdot 1,1888^n)$.

Алгоритм Уилфа также является рекурсивной процедурой. Однако он находит не наибольшую клику, а наибольшее независимое множество вершин графа. Алгоритм Уилфа базируется на следующем очевидном факте: для любой фиксированной вершины x , есть два вида независимых множеств: те, которые содержат x и те, которые не содержат x . Если независимое множество содержит x , то вершины, которые находятся в $N(x)$, не могут находиться в этом независимом множестве. Так что поиск надо продолжать в графе $G - \{x\} - N(x)$. Если независимое множество не содержит x , то поиск следует продолжать в графе $G - \{x\}$.

Показано, что время работы алгоритма Уилфа составляет $O(\text{poly}(n) \cdot 1,39^n)$. Данное время может быть улучшено, если входной граф предварительно разбить на атомы – подграфы, не содержащие кликовых минимальных сепараторов. Говорят, что множество вершин $S \subseteq V$ разделяет несмежные вершины a и b графа $G = (V, E)$, если в графе $G(V \setminus S)$ вершины a и b принадлежат различным компонентам связности. Множество S при этом называется (a, b) -сепаратором и минимальным (a, b) -сепаратором, если S есть (a, b) -сепаратор, и в нем нет собственного подмножества, являющегося (a, b) -сепаратором. Сепаратор S считают минимальным, если в графе G существует такая пара вершин a и b , что S – минимальный (a, b) -сепаратор. В частности, одновершинный сепаратор – точка сочленения графа. Разделение графа на атомы является обобщением разложения графа на блоки точками сочленения. Такое разделение сохраняет все клики исходного графа и не порождает новых клик. Для разреженного графа $G = (V, E)$ размер атома (число вершин подграфа, образующего атом), как правило, ограничен сверху некоторой константой $k < n$, а число атомов не превышает $n = |V|$. Разложить граф $G = (V, E)$ на атомы можно за время $O(n^3)$.

Применение алгоритма Уилфа к каждому атому позволяет найти максимальную клику входного графа в границах отдельного атома, при этом алгоритм затрачивает время $O(\text{poly}(k) \cdot 1,39^k)$, где $\text{poly}(k)$ – некоторый полином от числа вершин атома. Наибольшая клика определяется как наибольшая по мощности клика из всех найденных максимальных клик. Таким образом, для вычисления наибольшей клики графа $G = (V, E)$ необходимо время $O(n \cdot \text{poly}(k) \cdot 1,39^k)$. Заметим, что это время линейно зависит от n и экспоненциально от k . Чем меньше значение k (чем более разреженным является граф), тем быстрее работает алгоритм.

Модифицированный алгоритм Уилфа был программно реализован в среде Delphi 7. Входными данными программы является описание графа в виде списка ребер. Выход программы – плотность входного графа и его наибольшая клика.

Вычислительные эксперименты подтвердили эффективность предложенной модификации алгоритма Уилфа на разреженных графах.