

ПРИКЛАДНАЯ ТЕОРИЯ ГРАФОВ

DOI 10.17223/20710410/13/5

УДК 519.178

ВЫЧИСЛИТЕЛЬНЫЕ АСПЕКТЫ ДРЕВОВИДНОЙ ШИРИНЫ ГРАФА

В. В. Быкова

*Институт математики Сибирского федерального университета, г. Красноярск, Россия***E-mail:** bykvalen@mail.ru

Дан краткий обзор современных результатов по проблеме вычисления древовидной ширины. Представлены и исследованы некоторые нижние и верхние оценки данного числового параметра графа. Предложены алгоритмические методы улучшения этих оценок.

Ключевые слова: алгоритмы на графах, частичные k -деревья, древовидная ширина.

Древовидная ширина — числовой параметр, характеризующий меру древовидности графа. Графы с ограниченной древовидной шириной образуют специальный класс графов, называемых частичными k -деревьями. Этот класс графов был введен четверть века назад Н. Робертсоном и П. Д. Сеймуром [1]. Широкий интерес к изучению древовидной ширины графа вызван тем, что многие NP-трудные задачи теории графов, возникающие в различных приложениях, в том числе при моделировании надежности и безопасности компьютерных и коммуникационных систем, полиномиально разрешимы, если модельный граф — частичное k -дерево. В этих условиях задача может быть решена методом динамического программирования, причем алгоритмическая эффективность достигается за счет разложения модельного графа на части с помощью небольших (мощности не более k) сепараторов [2]. С вычислительной точки зрения весьма сложно установить, имеет ли заданный произвольный граф ограниченную древовидную ширину. Доказана NP-полнота этой задачи [3, 4]. Лишь для некоторых классов графов древовидная ширина известна заранее или находится без особого труда. Например, древовидная ширина любого ациклического графа не превышает 1, а у всякого последовательно-параллельного графа древовидная ширина не более 2. Хордаловые графы — типичный пример класса графов, для которых задача определения древовидной ширины разрешима за полиномиальное время. Поскольку в общем случае данную задачу решить трудно, то актуальны границы возможных значений древовидной ширины графа и методы их улучшения.

1. Основные числовые параметры графа

Введем необходимые обозначения и определения. Везде далее $G = (V, E)$ — неориентированный конечный связный граф без петель и кратных ребер, $n = |V|$ и $m = |E|$. Обозначим через $G(A)$ подграф графа G , индуцированный множеством $A \subseteq V$. Говорят, что множество вершин $S \subseteq V$ разделяет несмежные вершины a и b графа G , если в графе $G(V \setminus S)$ вершины a и b принадлежат различным компонентам связности. Множество S при этом называется сепаратором, или (a, b) -сепаратором. Сепаратор минимальный, если он не содержит другой сепаратор в качестве собствен-

ного подмножества. Наименьший по размеру сепаратор графа G — наименьший сепаратор. Сепаратор размера 1 называется точкой сочленения графа. Вершина $a \in V$ считается симплициальной вершиной графа G , если ее окрестность $N(a)$ индуцирует в G клику. Множество вершин A называется кликой графа G , если $G(A)$ — полный граф, и максимальной кликой, если она не содержится в клике с большим количеством вершин. Размер наибольшей клики графа G обозначается $\varphi(G)$ и называется плотностью графа G . Хроматическое число графа G определяется как наименьшее возможное количество цветов $\chi(G)$, в которые можно раскрасить вершины графа так, что любые смежные вершины будут окрашены в разные цвета. Для произвольного графа G всегда $\varphi(G) \leq \chi(G)$, а для совершенного графа $\varphi(G) = \chi(G)$ [5]. Совершенными, например, являются полные, двудольные и хордальные графы. Граф G называется хордальным, если ни один из его индуцированных подграфов не является простым циклом длины $l \geq 4$. Всякий индуцированный подграф хордального графа также является хордальным. Известны полиномиальные по времени алгоритмы распознавания хордальности графа G , основанные на симплициальных вершинах и сепараторах [6, 7]. Нахождение чисел $\varphi(G)$, $\chi(G)$ для графа G общего вида — NP-трудные задачи [2]. Между тем для хордального графа G числа $\varphi(G)$, $\chi(G)$ могут быть вычислены за полиномиальное время [7]. Любой граф можно превратить в хордальный, добавив в него некоторое множество ребер. Триангуляцией графа $G = (V, E)$ называется хордальный граф $H = (V, E')$, который содержит G в качестве остова подграфа ($E \subseteq E'$). Триангуляция H — минимальная триангуляция графа G , если для G не существует другой триангуляции, которая является собственным подграфом графа H . Триангуляция, имеющая наименьшее число ребер, называется наименьшей. Число ребер наименьшей триангуляции является числовым параметром графа G и обозначается $\text{ch}(G)$. Задача нахождения $\text{ch}(G)$ для произвольного графа (ее также называют задачей наименьшего пополнения графа до хордального) NP-трудная. Заметим, что если граф G хордальный, то $\text{ch}(G) = m$. Числом вершинной связности $\kappa(G)$ графа G называется наименьшее число вершин, удаление которых приводит к несвязному или одновершинному графу. Граф G считается l -связным, если $\kappa(G) \geq l$. Согласно теореме Менгера [5], чтобы граф G был l -связен, необходимо и достаточно, чтобы любые две его несмежные вершины a и b были соединены не менее чем l вершинно-непересекающимися (a, b) -цепями. Другими словами, граф l -связен тогда и только тогда, когда каждый его (a, b) -сепаратор содержит по крайней мере l вершин. Значит, число $\kappa(G)$ определяет размер наименьшего сепаратора графа G (исключение составляет лишь полный n -вершинный граф K_n , в котором нет сепараторов вообще, хотя $\kappa(K_n) = n - 1$). Если $\delta(G)$ — минимальная степень вершин графа G , то всегда $\kappa(G) \leq \delta(G)$. Примечательно, что задача вычисления числа $\kappa(G)$ для графа G сводится к задаче о максимальном потоке и минимальном разрезе и разрешима за полиномиальное время [8].

Перечисленные выше числовые параметры графа давно известны и хорошо изучены в теории графов. Все они с различных позиций характеризуют структуру графа и являются его инвариантами в аспекте изоморфизма. Дополняет этот набор инвариантов древовидная ширина как мера близости графа к дереву.

2. Дерево декомпозиции и древовидная ширина графа

Древовидная ширина графа вычисляется через специальную графовую структуру, которая называется деревом декомпозиции. Дерево декомпозиции графа $G = (V, E)$ представляет собой пару $(\mathcal{X}, T)$, где $\mathcal{X} = \{X_i : i \in I\}$ — семейство подмножеств мно-

жества V , называемых «мешками»; $T=(I, W)$ — дерево, узлам которого сопоставлены эти «мешки», и выполняются следующие условия [1, 9]:

- 1) $\bigcup_{i \in I} X_i = V$;
- 2) для всякого ребра графа G обязательно имеется хотя бы один «мешок», содержащий обе вершины этого ребра;
- 3) для любой вершины $v \in V$ графа G множество узлов $\{i \in I : v \in X_i\}$ индуцирует связный подграф, являющийся поддеревом дерева T .

Ширина дерева декомпозиции $(\mathcal{X}, T)$ равна $\max_{i \in I} \{|X_i| - 1\}$. Древовидная ширина (*treewidth*) графа G определяется как наименьшая ширина всех допустимых его деревьев декомпозиции и обозначается через $\text{tw}(G)$. Дерево декомпозиции $(\mathcal{X}, T)$ ширины $\text{tw}(G)$ называется оптимальным деревом декомпозиции графа G . Заметим, что если рассматривать только деревья декомпозиции без кратных и вложенных «мешков», то для каждого связного графа G множество возможных деревьев декомпозиции непусто и конечно. Например, всегда существует тривиальное дерево декомпозиции, которое состоит из одного «мешка», содержащего все вершины графа. Однако такое дерево декомпозиции в большинстве случаев не является оптимальным. Когда граф G несвязен, то полагают $\text{tw}(G) = 0$. Следовательно, числовой параметр $\text{tw}(G)$ может быть вычислен для любого конечного графа G .

Дерево декомпозиции графа — это весьма полезная графовая структура, которая определяет не только древовидную ширину графа, но и отражает информацию о его кликах и сепараторах. Это подтверждает

Утверждение 1 [10]. Пусть задано некоторое дерево декомпозиции $(\mathcal{X}, T)$ графа $G = (V, E)$. Тогда справедливы следующие высказывания:

- 1) если множество вершин $A \subseteq V$ образует в G клику, то в T существует узел $i \in I$, такой, что $A \subseteq X_i$, т. е. любая клика графа всегда целиком вложена в отдельный «мешок» дерева декомпозиции $(\mathcal{X}, T)$ и, возможно, не в один;
- 2) если множества A и B индуцируют в G полный двудольный подграф, то в T имеется узел $i \in I$, такой, что $A \subseteq X_i$ или $B \subseteq X_i$, т. е. все вершины хотя бы одной доли этого подграфа принадлежат отдельному «мешку» дерева декомпозиции $(\mathcal{X}, T)$;
- 3) пусть $w = \{i, j\}$ — произвольное ребро дерева T , а A и B — множества узлов двух компонент связности графа $T - w$, полученного удалением из дерева T ребра w . Тогда пересечение «мешков» $S = X_i \cap X_j$ образует (a, b) -сепаратор графа G при условии, что

$$a \in X_A = \bigcup_{l \in A} X_l \setminus S \neq \emptyset, \quad b \in X_B = \bigcup_{l \in B} X_l \setminus S \neq \emptyset.$$

Из данного утверждения следует, что если K_n — полный n -вершинный граф и K_{pq} — полный двудольный граф $(p + q = n)$, то $\text{tw}(K_n) = n - 1$, $\text{tw}(K_{pq}) = \min(p, q)$. В общем случае при условии связности графа G верны естественные границы значений его древовидной ширины

$$0 < \text{tw}(G) \leq n - 1. \quad (1)$$

Очевидны следующие леммы.

Лемма 1. Пусть граф $G - v$ получен из графа $G = (V, E)$ удалением вершины $v \in V$. Тогда $\text{tw}(G - v) \leq \text{tw}(G)$.

Лемма 2. Пусть граф $G + e$ образован из графа $G = (V, E)$ добавлением ребра $e = \{a, b\}$, где $a, b \in V$. Тогда $\text{tw}(G) \leq \text{tw}(G + e)$.

Данные леммы свидетельствуют о том, что древовидная ширина графа — числовой параметр, которому свойственна монотонность относительно операций удаления вершины и добавления ребра: древовидная ширина графа не возрастает при удалении из него произвольной вершины и не убывает при добавлении в граф дополнительного ребра.

Следствие 1. Для любого подграфа $G' = G(A)$ графа $G = (V, E)$, индуцированного множеством $A \subseteq V$, верно $\text{tw}(G') \leq \text{tw}(G)$.

Следствие 2. Для каждой триангуляции $H = (V, E')$ графа $G = (V, E)$, где $E \subseteq E'$, всегда $\text{tw}(G) \leq \text{tw}(H)$.

Минором графа $G = (V, E)$ принято называть граф G' , который получается в результате применения к G одной или нескольких операций удаления вершины, удаления ребра или стягивания ребра. Ясно, что всякий индуцированный подграф графа G является его минором, однако обратное не всегда верно. Следующая лемма обобщает леммы 1 и 2.

Лемма 3 [1]. Для всякого минора G' графа G всегда $\text{tw}(G') \leq \text{tw}(G)$.

3. Графы с ограниченной древовидной шириной

Древовидная ширина отражает, насколько близок граф G к дереву, и, согласно утверждению 1, определяет размеры его клик и сепараторов: чем меньше $\text{tw}(G)$, тем ближе граф G к дереву и тем меньше у него по мощности клики и сепараторы. Так, все n -вершинные деревья ($n \geq 2$) имеют единичную древовидную ширину, размер всякой клики такого дерева равен 2, а каждый сепаратор — точка сочленения. Считается, что граф G обладает ограниченной древовидной шириной, если $\text{tw}(G) \leq k$ и k — положительная целая константа, не зависящая от n [9]. Например, если G — последовательно-параллельный граф, то $\text{tw}(G) \leq 2$, а для всякого графа Халина неизменно $\text{tw}(G) = 3$. Однако ограниченная древовидная ширина свойственна не всем графам. В частности, $\text{tw}(K_n) = n - 1$. Кроме того, существуют совсем простые по структуре графы, для которых значение $\text{tw}(G)$ растет с увеличением числа вершин, т. е. при $n \rightarrow \infty$ подобные графы все больше отдаляются от дерева. Типичным примером являются «сетки» — графы, построенные из правильных многогранников так, что любые два соседних многогранника имеют лишь одно общее ребро. Для «сетки» размера $n_1 \times n_2$ древовидная ширина вычисляется по формуле $\text{tw}(G) = \min(n_1, n_2) + 1$. При увеличении размера «сетки» ее древовидная ширина увеличивается, а число вершинной связности, наименьшая степень вершины, плотность и хроматическое число графа остаются неизменными.

4. Проблема вычисления древовидной ширины

Установить, имеет ли заданный граф ограниченную древовидную ширину, не всегда просто. В [3, 4] доказана NP-полнота следующей задачи: для графа G и целого положительного числа k верно ли, что $\text{tw}(G) \leq k$? Существует другая формулировка данной задачи, основанная на следующем утверждении.

Утверждение 2 [10]. Граф G имеет древовидную ширину не больше k тогда и только тогда, когда он является частичным k -деревом.

Использование понятия частичного k -дерева позволяет выявить структуру графа с ограниченной древовидной шириной. Определение частичного k -дерева базируется

на понятии k -дерева, которое, в свою очередь, определяется рекурсивно по правилам: полный граф из $k + 1$ вершин есть k -дерево; k -дерево с $i + 1$ вершинами получается из k -дерева с i вершинами путем добавления в него вершины v и k ребер таким образом, чтобы v стала смежной со всеми вершинами некоторой клики размера k (k -клики). Характерные особенности k -деревьев непосредственно вытекают из их построения и отражены в следующем утверждении.

Утверждение 3. Если граф $G = (V, E)$ есть k -дерево, то

- 1) G — связный граф;
- 2) G — хордальный граф и $\text{ch}(G) = m = |E|$;
- 3) G всегда имеет k -клику, но в нем нет $(k + 2)$ -клики;
- 4) $\kappa(G) = \delta(G) = k$;
- 5) все максимальные клики в G есть $(k + 1)$ -клики;
- 6) $\varphi(G) = \chi(G) = k + 1$;
- 7) любой минимальный сепаратор в G является k -кликой;
- 8) $m = kn - k(k + 1)/2$.

Для того чтобы распознать k -дерево, нужно раскрутить рекурсию в обратном порядке. Это можно осуществить за полиномиальное время. Поскольку k -дерево — хордальный граф, то найти для него древовидную ширину не представляет труда (используя дерево клик).

Утверждение 4 [11]. Связный граф $G = (V, E)$ хордальный тогда и только тогда, когда для него существует дерево клик, определяемое следующим образом:

- 1) множество узлов дерева — множество $\{C_i : i \in I\}$ всех максимальных клик графа G ;
- 2) два узла C_i, C_j соединены ребром, если соответствующие им клики графа G имеют непустое пересечение, т. е. $C_i \cap C_j \neq \emptyset$;
- 3) для всякой вершины $v \in V$ графа G множество максимальных клик, содержащих эту вершину, индуцирует связный подграф, являющийся поддеревом дерева клик.

Известно, что любой хордальный граф имеет не более $n - 1$ максимальных клик [11]. Значит, дерево клик такого графа содержит не более $n - 1$ узлов и ребер. Хордальный граф может иметь несколько деревьев клик. Доказано [4], что каждое из них может быть найдено как остовное дерево наибольшего веса графа пересечений всех максимальных клик исходного графа. Здесь под весом ребра понимают число вершин, образующих пересечение подбоающих максимальных клик. Таким образом, дерево клик хордального графа можно всегда построить за время $O(n^2)$, применяя известные алгоритмы построения оптимального остова [5, 8]. Очевидно, что для связного хордального графа G дерево клик с точностью до изоморфизма определяет оптимальное дерево декомпозиции, а мощность крупнейшей клики — древовидную ширину графа G : $\text{tw}(G) = \varphi(G) - 1$. По утверждению 3 плотность k -дерева равна $k + 1$ и потому $\text{tw}(G) = k$.

Следствие 3. Всякое k -дерево — хордальный граф с древовидной шириной, равной k .

Принадлежность k -деревьев к классу хордальных графов позволяет использовать свойства хордальных графов в определении древовидной ширины и построении деревьев декомпозиции графа.

Частичным k -деревом называют остовный подграф k -дерева [10]. Поскольку всякий n -вершинный граф является остовным подграфом графа K_n , то можно утверждать,

что подобный граф есть частичное $(n - 1)$ -дерево. Между тем в аспекте ограниченной древовидной ширины интерес представляет наименьшее значение k , при котором граф G — частичное k -дерево. Отыскать такое значение k — это значит найти наименьшее k -дерево, в которое можно вложить граф G . Таким образом, по следствию 3 для частичного k -дерева верно неравенство $\text{tw}(G) \leq k$, которое отвечает утверждению 2. Справедливость этого факта вытекает из монотонности древовидной ширины относительно операции добавления ребра в граф G , ведь всякое погружение графа G в граф H , удовлетворяющий требованиям k -дерева или хордальности, сводится к насыщению G дополнительными ребрами. Для триангуляций графа G справедливо

Утверждение 5 [10]. Если H является наименьшей триангуляцией графа G , то $\text{tw}(G) = \text{tw}(H)$. Для произвольной триангуляции H всегда $\text{tw}(G) \leq \text{tw}(H)$.

На сегодня сформировались два основных подхода к построению триангуляций графа: через удаление вершин, окрестность которых предварительно насыщается ребрами до клики, и через минимальные сепараторы, которые также дополняются до клики [12, 13]. В рамках этих подходов разработано много алгоритмов со временем работы $O(n^3)$.

Несмотря на то, что задача распознавания частичного k -дерева NP-полная [14], это никак не мешает вычислять древовидную ширину графа во многих ситуациях. Во-первых, известны точные неполиномиальные по времени алгоритмы нахождения $\text{tw}(G)$, основанные на методе динамического программирования и методе ветвей и границ [15, 16]. Разработаны также FPT-алгоритмы (*Fixed-Parameter Tractable algorithms*) [9, 17], способные при фиксированном k за время $O(2^k n^{O(1)})$ дать ответ на вопрос: $\text{tw}(G) \leq k$? Во-вторых, разработаны точные алгоритмы полиномиальной сложности для различных специальных классов графов [18–20]. В-третьих, выявлены необходимые полиномиально проверяемые признаки частичного k -дерева [10]. Так, если G — частичное k -дерево, то количество его ребер ограничено сверху: $m \leq kn - k(k + 1)/2$. В-четвертых, для $k = 1, 2$ найдены легко проверяемые критерии распознавания частичного k -дерева: $\text{tw}(G) \leq 1$ тогда и только тогда, когда граф G ациклический; $\text{tw}(G) \leq 2$ тогда и только тогда, когда каждый блок графа G является последовательно-параллельным графом [21]. В-пятых, к настоящему времени предложены различные схемы приближений [22, 23] и эвристические алгоритмы [24], позволяющие за полиномиальное время находить значения, близкие к истинному значению древовидной ширины графа. К сожалению, пока остается открытым вопрос о существовании полиномиальных схем приближения с гарантированной ошибкой. И наконец, актуальны границы возможных значений древовидной ширины графа и методы их уточнения.

5. Нижние оценки древовидной ширины и метод их уточнения

Можно назвать несколько причин, по которым полезны нижние оценки для древовидной ширины графа. Прежде всего, они могут быть использованы как критерии отсека бесперспективных ветвей в методе ветвей и границ и в эвристических алгоритмах вычисления древовидной ширины графа. Кроме того, если в конкретной прикладной задаче для модельного графа G нижняя оценка $\text{tw}(G)$ слишком велика, то целесообразно отказаться от решения этой задачи методом динамического программирования на основе дерева декомпозиции. Иначе не исключена ситуация, когда время решения задачи может выйти за разумные пределы.

Рассмотрим нижние границы значений древовидной ширины графа G , связывающие $\text{tw}(G)$ с другими числовыми параметрами этого графа. Из свойств k -деревьев и

частичных k -деревьев (утверждений 2, 3 и следствия 3) непосредственно вытекают такие нижние оценки:

$$\delta(G) \leq \text{tw}(G); \quad (2)$$

$$\varkappa(G) \leq \text{tw}(G); \quad (3)$$

$$\varphi(G) - 1 \leq \text{tw}(G); \quad (4)$$

$$\chi(H) - 1 \leq \text{tw}(G); \quad (5)$$

$$\frac{2n - 1 - \sqrt{(2n - 1)^2 - 8m}}{2} \leq \text{tw}(G); \quad (6)$$

$$\frac{2n - 1 - \sqrt{(2n - 1)^2 - 8\text{ch}(G)}}{2} \leq \text{tw}(G). \quad (7)$$

Оценки (2)–(5) свидетельствуют о том, что если граф G имеет ограниченную древовидную ширину, т. е. $\text{tw}(G) \leq k$, то для него также ограничены сверху $\delta(G)$, $\varkappa(G)$, $\varphi(G)$, $\chi(H)$. Однако обратное, вообще говоря, неверно. Важно, что оценки (2), (3), (6) полиномиально вычисляемые. Но как раз они не очень хорошие.

Применение оценки (7) требует знания числа $\text{ch}(G)$, которое является результатом решения NP-трудной задачи наименьшего пополнения графа G до хордального. Для произвольной триангуляции H графа G всегда $\text{ch}(G) \leq \text{ch}(H)$. Поскольку всякая триангуляция H графа G по построению есть хордальный граф, то $\text{ch}(H)$ — число ребер в H . С учетом этого из (7) следует оценка

$$\frac{2n - 1 - \sqrt{(2n - 1)^2 - 8\text{ch}(H)}}{2} \leq \text{tw}(G). \quad (8)$$

Конечно, чем меньше использовано ребер для пополнения графа G до H , тем выше нижняя граница значений $\text{tw}(G)$ в (8). Поэтому при вычислении оценки (8) следует использовать триангуляции хорошего качества, для построения которых могут быть применены различные известные полиномиальные по времени приближенные и эвристические алгоритмы [22, 24].

Оценки (4) и (5), выраженные через плотность и хроматическое число графа, трудновычисляемые. На первый взгляд может показаться, что именно плотность и хроматическое число тесно связаны с древовидной шириной графа, и если значения $\varphi(G)$ и $\chi(G)$ невелики, то невелико и значение $\text{tw}(G)$. Однако на самом деле это не так: разности $\text{tw}(G) - \varphi(G)$, $\text{tw}(G) - \chi(G)$ могут быть величинами, зависящими от n . Например, «сетка» из треугольников размера $n_1 \times n_2$ неизменно имеет плотность и хроматическое число, равные 3, при этом значение древовидной ширины зависит от размера «сетки». Существуют графы без треугольников с произвольно большим хроматическим числом [5, с. 242]. Правила построения последовательности $G_2, G_3, \dots, G_i, \dots$ таких графов указал А. А. Зыков в 1949 г. В этой последовательности $G_2 = K_2$, а каждый граф G_i , $i = 3, 4, \dots$, имеет $n_i = 3 \cdot 2^{i-2} - 1$ вершин, $\varphi(G_i) = 2$, $\chi(G_i) = i$, $\text{tw}(G_i) = (n_i - 1)/2$. Поэтому значения разностей $\text{tw}(G_i) - \varphi(G_i)$, $\text{tw}(G_i) - \chi(G_i)$ увеличиваются с ростом i .

Полиномиально вычисляемые оценки (2), (3), связанные со степенями вершин графа, можно усилить. Для этого следует воспользоваться монотонностью $\text{tw}(G)$ и немонотонностью $\delta(G)$ и $\varkappa(G)$ относительно операции удаления вершины.

Лемма 4. Пусть $\Psi(G)$ — множество всех индуцированных подграфов графа G . Тогда

$$\max_{G' \in \Psi(G)} \delta(G') \leq \text{tw}(G); \quad (9)$$

$$\max_{G' \in \Psi(G)} \varkappa(G') \leq \text{tw}(G). \quad (10)$$

Доказательство. В самом деле, согласно следствию 1 и оценке (2), для каждого индуцированного подграфа $G' \in \Psi(G)$ графа G справедливы неравенства

$$\delta(G') \leq \text{tw}(G') \leq \text{tw}(G).$$

Отсюда вытекают соотношения

$$\max_{G' \in \Psi(G)} \delta(G') \leq \max_{G' \in \Psi(G)} \text{tw}(G') \leq \text{tw}(G)$$

и правильность (9). Аналогичным образом с использованием (3) доказывается (10). ■

В общем случае множество $\Psi(G)$ содержит порядка 2^n индуцированных подграфов графа G , и их перебор может потребовать слишком много времени. Поэтому при больших значениях n в (9) и (10) вместо множества $\Psi(G)$ целесообразно использовать некоторое его подмножество, которое имеет небольшую мощность и отражает структурные особенности графа G через его части. Таким подмножеством, например, может быть множество $\mathfrak{P}(G) \subseteq \Psi(G)$, называемое колодой графа G . Колода n -вершинного графа $G = (V, E)$ состоит из n его подграфов $G - v$, полученных из G удалением только одной вершины $v \in V$. Для $\mathfrak{P}(G)$ оценки (9) и (10) полиномиально вычислимы. При формировании колоды $\mathfrak{P}(G)$ удалению подлежит каждая вершина графа G , включая специальные вершины, такие, как симплициальные вершины и точки сочленения, если они имеются в G . Примечательно, что чем больше специальных вершин в графе G , тем больше шансов улучшить нижние оценки (2), (3) на основе $\mathfrak{P}(G)$. Наибольший эффект колоды обеспечивают симплициальные вершины.

Лемма 5. Если $v \in V$ есть симплициальная вершина графа $G = (V, E)$ степени $\deg(v)$, то

$$\max(\deg(v), \text{tw}(G - v)) = \text{tw}(G). \quad (11)$$

Справедливость данной леммы следует из утверждения 7, которое приведено и доказано ниже. Равенство (11) дает для симплициальной вершины v соотношения

$$\max(\deg(v), \delta(G - v)) \leq \text{tw}(G), \quad \max(\deg(v), \varkappa(G - v)) \leq \text{tw}(G).$$

Таким образом, при вычислении оценок (9), (10) по $\mathfrak{P}(G)$ для каждой симплициальной вершины v вместо $\delta(G - v)$ и $\varkappa(G - v)$ необходимо применять соответствующие величины

$$\max(\deg(v), \delta(G - v)), \quad \max(\deg(v), \varkappa(G - v)).$$

Распознается симплициальная вершина за время $O(n^2)$.

Для уточнения оценок (2) и (3) в качестве представительного подмножества множества $\Psi(G)$ можно рассматривать также множество $\mathfrak{B}(G)$ блоков графа G . Это множество получается разделением графа G на части с помощью точек сочленения. Напомним, что блок графа G — максимальный относительно включения связный его подграф

без точек сочленения. Здесь важно отметить, что каждый блок является индуцированным подграфом графа G (значит, к нему применимо следствие 1), $|\mathfrak{B}(G)| \leq n$ и всякий блок можно отыскать в G за полиномиальное время. Конечно, если граф G состоит только из одного блока (таким, например, является «сетка» из треугольников), то предложенный выше алгоритмический метод улучшения нижних оценок (2), (3) не дает эффекта. В подобных случаях следует искать другие правила учета специальных вершин графа и выделения в $\Psi(G)$ представительного подмножества или оценивать значения древовидной ширины сверху.

6. Верхнее конструктивное оценивание и предобработка графа

Явных верхних оценок сравнительно немного. К ним можно отнести тривиальную оценку (1). Известны также верхние оценки для некоторых специальных классов графов. Например, для всякого планарного графа G с диаметром $d(G)$ выполняется соотношение [25] $\text{tw}(G) \leq 3d(G) - 2$. В общем случае на практике для определения верхних границ древовидной ширины графа чаще всего используют конструктивное оценивание на основе неравенства

$$\text{tw}(G) \leq \text{tw}(H), \quad (12)$$

которое, согласно следствию 2 и утверждению 5, верно для любой триангуляции H графа G . Здесь, как и в оценке (8), нужны качественные триангуляции.

Тривиальную оценку (1) и верхние оценки, найденные с помощью триангуляций, можно уточнить с помощью процедуры предобработки, основанной на принципе «разделяй и властвуй». Цель предобработки — разделить исходный граф G на части таким образом, чтобы задача нахождения наименьшей триангуляции H для графа G свелась к решению той же самой задачи, но только для графов меньшей размерности. Затем в зависимости от размера полученных частей графа использовать для построения качественных триангуляций точные, приближенные или эвристические алгоритмы. Один из возможных подходов надлежащего разбиения графа — выделение частей графа с помощью сепараторов.

Пусть $\mathfrak{S}(G) = \{S_j : j \in I_s\}$ — множество сепараторов графа $G = (V, E)$. Предположим сначала, что всякий сепаратор состоит лишь из одной вершины — точки сочленения графа G . Рассмотрим семейство $\mathfrak{B}(G) = \{B_i : i \in I_b\}$ блоков, полученных путем разделения графа точками сочленения. Пусть $\{X_i : i \in I_b\}$ — семейство блоковых множеств, где $X_i \subseteq V$ состоит из вершин блока B_i , и только из них ($i \in I_b$). Будем опираться на известные свойства блоков [5]:

- 1) любые два блока имеют не более одной общей вершины, а каждое ребро графа входит только в один его блок;
- 2) если блок графа содержит вершины a и b , то он также содержит и всякую простую (a, b) -цепь этого графа;
- 3) семейство блоковых множеств является покрытием множества вершин графа. Каждая пара блоковых множеств либо не пересекается, либо имеет единственную общую вершину, и эта вершина — точка сочленения графа.

Соотнесем графу G двудольный граф $T = (I, W)$ с множеством узлов $I = I_b \cup I_s$, в котором два узла i и j смежны, если узел i соответствует блоку B_i , а узел j — сепаратору S_j и $S_j \subset X_i$ ($i \in I_b, j \in I_s$). Такой граф описывает структуру графа G с точки зрения взаимного расположения его блоков и точек сочленения. Исходя из перечисленных выше свойств блоков, нетрудно убедиться, что если граф G связан, то T —

всегда дерево [5]. Это дерево принято называть деревом блоков и точек сочленения графа G . На его основе построим дерево декомпозиции графа G . Для этого сопоставим каждому узлу дерева T «мешок» по следующим правилам: пусть для узла $i \in I_b$ в роли «мешка» выступает блоковое множество X_i , а для узла $j \in I_s$ — множество вершин сепаратора S_j . Таким образом, «мешки» для узлов-сепараторов будут содержать только одну вершину — соответствующую точку сочленения графа G .

Утверждение 6. Пара $(\mathcal{X}, T)$, где $\mathcal{X} = \{X_i : i \in I_b\} \cup \{S_j : j \in I_s\}$ и $T = (I, W)$ — дерево блоков и точек сочленения графа G , определяет дерево декомпозиции этого графа.

Доказательство. Достаточно проверить выполнимость трех требований, которым должно удовлетворять всякое дерево декомпозиции. Доказательство опирается на свойства блоков. В самом деле, в качестве «мешков» здесь выступают блоковые множества и одновершинные сепараторы. Каждый из таких сепараторов принадлежит двум и более блоковым множествам. Согласно свойству 3, семейство блоковых множеств покрывает все вершины графа G . Исходя из свойства 1, всякое ребро $\{a, b\}$ графа G входит в некоторый блок, а значит, оба конца данного ребра принадлежат «мешку», отвечающему этому блоку. На основании свойств 1 и 3 вершина $v \in V$ графа G , если она не является точкой сочленения этого графа, всегда принадлежит лишь одному блоковому множеству (одному «мешку») и, следовательно, образует поддереву дерева T , состоящее из одного узла. Если вершина v — точка сочленения графа G , то по построению дерева блоков и точек сочленения в T всегда существует узел $j \in I_s$, смежный с узлами $i_1, i_2, \dots, i_\tau \in I_b$, которые соответствуют некоторому подмножеству блоков $B' \subseteq \mathfrak{B}(G)$ графа G , причем точка сочленения v принадлежит «мешкам» только этих блоков и никаким другим. Таким образом, все узлы дерева, «мешки» которых содержат точку сочленения v , индуцируют связный подграф, являющийся поддеревом дерева T . Все три требования, предъявляемые к дереву декомпозиции, выполнены. ■

Заметим, что дерево декомпозиции, определенное через блоковые множества и точки сочленения графа G , не имеет кратных «мешков», но обязательно имеет вложенные «мешки». Непустые пересечения «мешков» этого дерева задают все одновершинные сепараторы графа G . Ширина данного дерева устанавливает верхнюю границу значений $\text{tw}(G)$. К сожалению, если граф G является 2-связным, то эта граница не отличается от границы, заданной неравенством (1). Если граф G содержит точки сочленения, т. е. является 1-связным, то для $\mathfrak{B}(G)$ верна формула

$$\text{tw}(G) = \max\{\text{tw}(B_i) : i \in I_b\}, \quad (13)$$

которая означает, что древовидная ширина графа G может быть найдена через значения древовидной ширины всех его блоков. Пусть для каждого блока B_i определена некоторая его триангуляция H_i ($i \in I_b$). Тогда очевидна верхняя оценка

$$\text{tw}(G) \leq \max\{\text{tw}(H_i) : i \in I_b\}. \quad (14)$$

Поскольку при разложении графа на блоки размерность задачи построения наименьшей триангуляции снижается, то область действия погрешностей, вносимых схемами приближений и эвристиками, сокращается. Возможно также применение точных методов для блоков с небольшим числом вершин. Улучшение качества триангуляций H_i позволяет в силу (14) совершенствовать оценку (12).

Деревья декомпозиции, построенные для блоков, можно «склеить» через соответствующие сепараторы в графовую структуру, которая по утверждению 6 является

деревом декомпозиции исходного графа G . Заметим, что в данном случае все сепараторы, образующие $\mathfrak{S}(G)$, являются минимальными и попарно непересекающимися, так как они одновершинные. Семейство $\mathfrak{B}(G) = \{B_i : i \in I_b\}$ можно создавать поэтапно — путем нахождения очередной точки сочленения в какой-либо компоненте связности графа, полученной на предыдущем этапе разбиения. При этом состав блоков в $\mathfrak{B}(G)$ не зависит от порядка выбора точек сочленения.

Рассмотренный подход к предобработке графа можно обобщить на произвольные сепараторы. Если граф $G = (V, E)$ является l -связным, то $\chi(G) \geq l$ и в G существуют сепараторы размера l . Пусть $\mathfrak{S}(G)$ — множество сепараторов графа G и $S \in \mathfrak{S}(G)$ — некоторый сепаратор размера $l = |S|$. Обозначим через $Y_1, Y_2, \dots, Y_\tau$ области связности графа $G(V \setminus S)$. Части $G_1, G_2, \dots, G_\tau$ графа G , как результат разбиения его сепаратором S , определим сначала как подграфы, индуцированные множествами вершин $X_i = Y_i \cup S$ ($i \in I = \{1, 2, \dots, \tau\}$). Для сохранения l -связности к каждой части добавим всевозможные ребра между вершинами множества S . Таким образом, если через $K(S)$ обозначить полный граф на множестве вершин S , то окончательно $G_i = G(Y_i \cup S) + K(S)$, т. е. в каждом подграфе G_i вершины сепаратора S образуют клику ($i \in I$). Сепаратор S графа G назовем безопасным сепаратором относительно $\text{tw}(G)$, если

$$\text{tw}(G) = \max\{\text{tw}(G_i) : i \in I\}. \quad (15)$$

Лемма 6. Для произвольного сепаратора S графа G всегда

$$\text{tw}(G) \leq \max\{\text{tw}(G_i) : i \in I\}. \quad (16)$$

Доказательство. Предположим, что значение максимума в правой части (16) равно μ . Тогда для каждого подграфа G_i можно построить дерево декомпозиции ширины не более μ ($i \in I$). По высказыванию 1 утверждения 1 в этом дереве обязательно найдется узел, «мешок» которого содержит сепаратор S , так как вершины этого сепаратора образуют в G_i клику. Дерево декомпозиции графа G может быть сформировано путем добавления в каждое дерево декомпозиции, построенное для G_i ($i \in I$), дополнительного узла с «мешком», равным S , и «склеиванием» данных деревьев посредством этого дополнительного узла. Очевидно, что ширина результирующего дерева декомпозиции не будет превышать μ . ■

Утверждение 7. Если сепаратор S графа G — клика, то он безопасный относительно $\text{tw}(G)$.

Доказательство. Если $G(S)$ — клика, то для любого подграфа G_i , как части разбиения G сепаратором S , справедливо равенство

$$G_i = G(Y_i \cup S) + K(S) = G(Y_i \cup S), \quad i \in I,$$

которое означает, что G_i — индуцированный подграф графа G . Исходя из следствия 1, $\text{tw}(G_i) \leq \text{tw}(G)$. Отсюда

$$\max\{\text{tw}(G_i) : i \in I\} \leq \text{tw}(G). \quad (17)$$

Из неравенств (16), (17) вытекает равенство (15). ■

Из утверждения 7 следует справедливость леммы 5, так как окрестность всякой симплициальной вершины — сепаратор, являющийся кликой и отделяющий эту вершину от других вершин графа. Утверждение 7 определяет достаточные условия безопасности сепаратора относительно древовидной ширины графа. Одновершинные сепараторы отвечают данному условию как одновершинные полные графы. Это является подтверждением равенства (13).

Следствие 4. Все одновершинные сепараторы графа G являются безопасными относительно $\text{tw}(G)$.

Условие утверждения 7 можно расширить. Сепаратор S графа G есть почти клика, если существует вершина $v \in S$, такая, что множество $S - v$ образует клику графа G .

Лемма 7 [3]. Сепаратор S графа G является минимальным тогда и только тогда, когда для каждой вершины $v \in S$ и всякой области связности Y_i ($i \in I$) графа $G(V \setminus S)$ всегда найдется вершина $w \in Y_i$, смежная с v в $G(Y_i \cup S)$.

Утверждение 8. Если S — почти клика и минимальный сепаратор графа G , то он безопасный относительно $\text{tw}(G)$.

Доказательство. Пусть вершина $v \in S$ такова, что множество $S - v$ образует клику графа G . Покажем, что каждая компонента связности $G(Y_i \cup S)$ графа $G(V \setminus S)$ содержит $K(S)$ в качестве минора ($i \in I$). Рассмотрим вершину v минимального сепаратора S , которая не входит в клику этого сепаратора. Согласно лемме 7, существует вершина $w \in Y_i$, смежная с v в $G(Y_i \cup S)$. Для всякой другой вершины из $S - v$ также имеется смежная вершина в $G(Y_i \cup S)$. Исходя из этого, все вершины области связности Y_i можно стянуть к v . В результате стягивания получим граф $K(S)$. Убедимся теперь в справедливости равенства (15). Кроме $G(Y_i \cup S)$, проанализируем еще одну компоненту связности графа $G(V \setminus S)$. Пусть это будет компонента $G(Y'_i \cup S)$. Она так же, как и $G(Y_i \cup S)$, включает $K(S)$ в качестве минора. Между тем граф $G - Y_i$ содержит $G(Y'_i \cup S)$ в качестве подграфа, а значит, $K(S)$ в качестве минора. Отсюда следует, что граф $G_i = G(Y_i \cup S) + K(S)$ является минором для G . Поэтому по лемме 3 верно неравенство (17) и, учитывая лемму 6, справедливо равенство (15). ■

Следствие 5. Все минимальные двухвершинные сепараторы графа G безопасные относительно $\text{tw}(G)$.

Справедливость следствия 5 непосредственно вытекает из утверждения 8, так как всякое двухэлементное множество вершин графа G образует почти клику.

Результаты утверждений 6–8 и следствий 4, 5 могут быть использованы для уточнения верхних оценок древовидной ширины исходного графа G следующим образом:

- 1) первоначально семейство $\mathfrak{B}(G)$ состоит только из G ;
- 2) для каждого элемента из $\mathfrak{B}(G)$ — некоторой части исходного графа — находим безопасный сепаратор. Поиск начинаем с сепараторов размера 1, затем, если таких нет, выполняем поиск минимальных сепараторов размера 2 и т. д.;
- 3) если безопасный сепаратор найден, то рассматриваемую часть подвергаем разбиению и полученные более мелкие части помещаем в $\mathfrak{B}(G)$. Если безопасный сепаратор не найден, то этот элемент из $\mathfrak{B}(G)$ далее уже не подвергается обработке;
- 4) шаги 2 и 3 повторяем до тех пор, пока в $\mathfrak{B}(G)$ не окажется частей с безопасными сепараторами;
- 5) далее применяем алгоритмы триангуляции к элементам из $\mathfrak{B}(G)$, если это необходимо, и затем соотношение (14).

Для того чтобы описанный процесс можно было реализовать за полиномиальное время, необходимо на шаге 2 осуществлять поиск только тех сепараторов, которые легко вычисляются. К ним, например, относятся сепараторы размера 1 и 2. Они вычисляются с помощью поиска в глубину [5] за время $O(n + m)$. Для нахождения сепараторов размера 3 и более можно привлекать потоковые алгоритмы (Форда — Фалкерсона, Эд-

монса — Карпа и др.) [8]. Однако эти алгоритмы более затратные по времени, хотя и являются полиномиальными.

Следует отметить, что результат разбиения исходного графа на части безопасными сепараторами, вообще говоря, зависит от порядка выбора этих сепараторов, поскольку при $l > 1$ в l -связном графе допустимы зависимые (разделяющие друг друга) сепараторы мощности l , и после разбиения по одному из них невозможно уже произвести разбиение по второму [26]. Тем не менее если использовать только безопасные сепараторы, то на каждом шаге разбиения верно (15) для любого построенного варианта $\mathfrak{B}(G)$. Многовариантность может сказаться только на построении триангуляций для элементов из $\mathfrak{B}(G)$ и, следовательно, на значении верхней границы древовидной ширины графа G в (14). Конечно, существуют графы, в которых нет минимальных сепараторов, являющихся кликами или почти кликами. Для них предложенный выше алгоритмический метод улучшения верхней оценки древовидной ширины не дает эффекта. В таких случаях требуются другие безопасные сепараторы и иные методы предварительной обработки графа.

Заключение

В настоящее время проблеме вычисления древовидной ширины графа и построения дерева декомпозиции в зарубежных научных изданиях уделяется много внимания. Использование дерева декомпозиции при решении NP-трудных графовых задач — один из современных подходов преодоления высокой вычислительной сложности этих задач. В российских изданиях тематика древовидной ширины графов освещена весьма скромно. Имеются работы по исследованию структуры сепараторов в многосвязных графах [26, 27]. Частично введена соответствующая терминология в словарь по теории графов [28]. Изучаются хордальные графы [29] и ведется разработка новых алгоритмов построения минимальных триангуляций [30]. Имеются работы по применению древовидной ширины графа в решении задач дискретной оптимизации [31]. Однако эти работы не охватывают весь спектр открытых вопросов, связанных с древовидной шириной графа.

В данной работе дан краткий обзор современных результатов по проблеме вычисления древовидной ширины. Исследованы нижние и верхние оценки, связывающие древовидную ширину с другими числовыми параметрами графа, проанализировано их качество и сложность вычисления. Предложены и теоретически обоснованы алгоритмические методы улучшения этих оценок, основанные на немонотонности отдельных числовых параметров графа относительно операции удаления вершин графа и разложении графа сепараторами. Данные методы указывают направления исследований, перспективные для развития алгоритмических и прикладных аспектов теории графов.

ЛИТЕРАТУРА

1. Robertson N. and Seymour P. D. Graph minors. II. Algorithmic aspects of treewidth // J. Algorithms. 1986. V. 7. P. 309–322.
2. Kleinberg J. and Tardos E. Algorithm Design. Boston: Addison-Wesley, 2005.
3. Bodlaender H. L. and Thilikos D. M. Treewidth for graphs with small chordality // Disc. Appl. Math. 1997. V. 79. P. 45–61.
4. Parra A. and Scheffler P. Characterizations and algorithmic applications of chordal graph embeddings // Disc. Appl. Math. 1997. V. 79. P. 171–188.
5. Емеличев В. А., Мельников О. И., Сарванов В. И., Тышкевич Р. И. Лекции по теории графов. М.: Наука, 1990.

6. *Rose D., Tarjan R. E., and Lueker G.* Algorithmic aspects of vertex elimination on graphs // SIAM J. Comput. 1976. V. 5. P. 146–160.
7. *Buneman P.* A characterization of rigid circuit graphs // Disc. Math. 1974. V. 9. P. 205–212.
8. *Быкова В. В.* Дискретная математика с использованием ЭВМ. Красноярск: РИО КрасГУ, 2006.
9. *Bodlaender H. L.* Some classes of graphs with bounded treewidth // Technical Report RUU-CS-76-22, Dept. Comput. Science, Univ. Utrecht, 1998.
10. *Bodlaender H. L.* A partial k -arboretum of graphs with bounded treewidth // Theor. Comput. Sci. 1998. V. 209. P. 1–45.
11. *Blair J. R. S. and Peyton B.* An introduction to chordal graphs and clique trees // Graph theory and sparse matrix computation. New York: Springer, 1993. P. 1–29.
12. *Berry A.* A wide-range efficient algorithm for minimal triangulation // Proc. Tenth Annual ACM-SIAM Symposium on Disc. Algorithms. Philadelphia: SIAM, 1999. P. 860–861.
13. *Berry A., Heggernes P., and Simonet G.* The minimum degree heuristic and the minimal triangulation process // LNCS. 2003. V. 2880. P. 58–70.
14. *Arnborg S., Corneil D. G., and Proskurowski A.* Complexity of finding embeddings in a k -tree // SIAM J. Alg. Disc. Meth. 1987. V. 8. P. 277–284.
15. *Gogate V. and Dechter R.* A complete anytime algorithm for treewidth // Proc. 20Th Conference on Uncertainty in Artificial Intelligence. Arlington, Virginia, USA: AUAI Press, 2004. P. 201–208.
16. *Bodlaender H. L., Fomin F. V., Koster A. M. C. A., et al.* On exact algorithms for treewidth // LNCS. 2006. V. 4168. P. 672–683.
17. *Bodlaender H. L. and Kloks T.* Efficient and constructive algorithms for the pathwidth and treewidth of graphs // J. Algorithms. 1996. V. 21. P. 358–402.
18. *Bodlaender H. L. and Rotics U.* Computing the treewidth and the minimum fill-in with the modular decomposition // Algorithmica. 2003. V. 36. P. 375–408.
19. *Broersma H., Dahlhaus E., and Kloks T.* A linear time algorithm for minimum fill-in and tree width for distance hereditary graphs // Disc. Appl. Math. 2000. V. 99. P. 367–400.
20. *Broersma H., Kloks T., Kratsch D., and Muller H.* A generalization of AT-free graphs and a generic algorithm for solving triangulation problems // Algorithmica. 2002. V. 32. P. 594–610.
21. *Bodlaender H. L. and Fluiter B.* Parallel algorithms for series parallel graphs and graphs with treewidth two // Algorithmica. 2001. V. 29. P. 543–559.
22. *Amir E.* Efficient approximations for triangulation of minimum treewidth // Proc. 17Th Conference on Uncertainty in Artificial Intelligence. CA, USA: Morgan Kaufmann Publishers Inc. San Francisco, 2001. P. 7–15.
23. *Bouchitte V., Kratsch D., Muller H., and Todinca I.* On treewidth approximations // Disc. Appl. Math. 2004. V. 6. P. 183–196.
24. *Clautiaux F., Moukrim A., Negre S., and Carlier J.* Heuristic and meta-heuristic methods for computing graph treewidth // RAIRO Oper. Res. 2004. V. 38. P. 13–26.
25. *Eppstein D.* Diameter and treewidth in minor-closed families // Algorithmica. 2000. V. 27. P. 275–291.
26. *Карпов Д. В.* Разделяющие множества в k -связном графе // Зап. научн. семин. ПОМИ. 2006. Т. 340. С. 33–60.
27. *Карпов Д. В., Пастор А. В.* О структуре k -связного графа // Зап. научн. семин. ПОМИ. 2000. Т. 266. С. 76–106.
28. *Евстигнеев А. А., Касьянов В. Н.* Словарь по графам в информатике. Новосибирск: ООО «Сибир. научн. изд-во», 2000.

-
29. *Турсунбай кызы Б.* Деревья клик хордального графа и деревья подграфов // Конструирование и оптимизация программ. Новосибирск: Институт систем информатики СО РАН, 2008. Вып. 16. С. 314–321.
 30. *Быкова В. В., Никульская Н. А.* Алгоритмические аспекты минимальных триангуляций графа // Труды XIV Межд. конф. по эвент. матем. и смежным вопросам. Красноярск: КГТЭИ, СФУ, 2010. С. 26–32.
 31. *Щербина О. А.* Локальные элиминационные алгоритмы решения разреженных дискретных задач // Журн. вычисл. матем. и матем. физ. 2008. Т. 48. № 1. С. 159–175.