Секция 6. ТЕОРИЯ И МЕТОДИКА ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

УДК 519.85

Использование строковых матриц смежности в алгоритмах на графах

Ю.А. Алябышева[1], *А.А. Веряев*[2]
[1]*АлтГУ, Барнаул;* [2]*АлтГПУ, Барнаул*

Истоки постановки решаемой задачи следующие. В литературе, посвященной применению матриц смежности, описывающих графы, доказывается теорема о поиске количества путей заданной длины (см. например [1]): Пусть A=$a_{i,j}$ – матрица смежности ориентированного графа без петель и $A^l = \gamma_{i,j}$, где l <=N . Тогда $\gamma_{i,j}$ равно количеству путей $v_i \rightarrow v_j$ длины l.

Возник вопрос, почему при возведении матриц в степень, элементы получающихся матриц несут информацию только о числе путей и теряется информация, относящаяся к описанию самих путей? Можно ли что-то предпринять, чтобы сохранялась информация о способах перехода из вершины i в вершину j? Ответам на эти вопросы и посвящены настоящие материалы.

На первый взгляд, вопросы могут показаться не актуальными, однако в ряде авторитетных курсов по разработке и использованию алгоритмов (см., например, [2]) справедливо утверждается, что одной из самых фундаментальных и выразительных структур в математике являются графы, что чем больше работаешь с графами, тем скорее начинаешь замечать их повсюду.

Первое, на что обратим внимание, это то, что при работе программистов с графами происходит, как правило, редукция базовых представлений об этой структуре и манипуляция упрощенными представлениями о графе в памяти компьютера, вернее описанием графа в задачах. При этом сознательную потерю информации о графе при программировании приходится компенсировать введением дополнительных структур данных, параллельно возникающих при решении задачи и чаще всего связанных со спецификой решаемой задачи.

Поясним сказанное. Пусть задано N вершин, которые перенумерованы от 1 до N. Некоторые из вершин соединены между собой дугами/ребрами. Если из вершины i в вершину j идет дуга (ребро), то этот факт описывают величиной $a_{i,j}$. Совокупность всех величин $a_{i,j}$ порождает так называемую матрицу смежности, используемую для задания графов при работе на компьютерах. Таким образом, матрица смежности графа G с числом вершин N представляет собой квадратную матрицу A размера N, в которой значение элемента $a_{i,j}$ равно числу рёбер из i-й вершины графа в j-ю вершину. Если задан граф не взвешенный без петель, то в этом случае в матрицу на место элементов, соответствующих отсутствующим ребрам ставят 0. Если из вершины в вершину идет одно ребро, на месте элемента для невзвешенных графов ставят 1, диагональные элементы при этом все равны нулю. Во взвешенных графах, на месте единиц, стоят веса ребер. В этом коротком отрезке, поясняющем введение матрицы смежности и присутствующем практически во всех пособиях, посвященных алгоритмам на графах, произошла указанная выше редукция. Действительно, число, стоящее в матрице смежности и элемент $a_{i,j}$ несут разную информацию. В записи элемента $a_{i,j}$ подразумевается и числовое значение, и местоположение в матричной структуре, числа же из матрицы уже не содержат вторую составляющую информации, они имеют смысл не как отдельные объекты, а как составные части структуры, называемой «матрица». При дальнейших манипуляциях с матрицами смежности это различие начинает проявлять себя. Если числа из матрицы смежности «рассыпать», затем перемешать, то граф G уже не будет подлежать восстановлению с помощью этого описания. Набор же элементов a_{ij}, превращенный в простое множество элементов, позволит восстановить G полностью.

Для того, чтобы потери информации при задании матриц смежности не происходило, нужно рассматривать $a_{i,j}$ как строковые переменные или слова над алфавитом. Как правило, такой алфавит состоит из множества цифр. Можно использовать любую систему счисления, но заранее необходимо оговорить, сколько разрядов будет использовано для задания индексов и веса (весов) ребра. Логика решения поставленной задачи требует введения операций над словами, а затем модификации некоторых операций над матрицами. Дальнейшее изложение хоть и обладает некоторой общностью, тем не менее ориентировано на применение сказанного к алгоритмам над графами и осуществления операций над матрицами смежности.

1. Слова в алфавите имеют структуру, содержат три поля: первое поле – первый индекс i, второе поле – вес ребра (при работе с мультиграфами во втором поле может быть закодированы несколько весов), третье поле – второй индекс j. Удобно даже записывать слова в виде ${}_ia_j$ (что мы, впрочем, далее делать не будем, но мысленное использование такой записи упрощает понимание дальнейшего). Назовем слово с такой структурой атомарным. Можно рассматривать атомарное слово

как структуру данных, называемую в языках программирования или в теории баз данных «запись». Атомарное слово – двунаправленный связный список, в котором i и j – указатели. «Средняя» часть записи описывает вес ребра.

2. Необходимо ввести операции извлечения веса (весов) из слова $a_{i,j}$, и пары индексов i, j, последовательность которых важна. Для неориентированных графов вес ребер одинаков, сами же элементы $a_{i,j}$, $a_{j,i}$ – различны.

3. Далее вводится две базовые операции над словами: сложение и конкатенация. Конкатенация возможна, только в том случае, если рядом стоящие индексы у слов одинаковы: $a_{ij}a_{jk}$. В итоге получается составное слово. Операция конкатенации необходима для того, чтобы в дальнейшем сделать возможным переход от матриц смежности к матрицам достижимости, которые описывают не только число путей из вершины i в вершину j, но и будут описывать сами эти пути. Возможны составные слова, состоящие из произвольного количества атомарных слов. Для графа с числом вершин N, как правило, имеет смысл ограничиваться словами, состоящими из N атомарных слов. Возможными оказываются слова, у которых первый и последний индексы одинаковы. Такие слова описывают циклы. В матрицах достижимости они стоят на диагоналях. Конкатенация напоминает традиционную запись умножения двух элементов матрицы, однако веса при работе со словами нужно не умножать, а складывать. Таким образом, извлечение веса из составного слова сводится к фрагментации составного слова на атомарные слова, извлечение весов из атомарных слов и их последующее сложение.

4. Сложение же слов предлагается рассматривать как объединение элементов: $a_{i,j}+b_{i,j}$. Граф может содержать, например, два разных ребра с различающимися весами, соединяющие одни и те же вершины. Если же ребро одно, то есть $a_{i,j}=b_{i,j}$, объединение дает единственный указанный элемент. Объединяться могут слова с разным количеством атомарных слов, содержащие одинаковые первый и последний индексы. Такое объединение описывает разные способы попадания из одной вершины графа в другую.

5. Базовая операция над рассматриваемыми словами – конкатенация – некоммутативна. В общем случае для двух слов a и b $ab \neq ba$.

6. Нужно считать, что на месте нулевых элементов (в матрице смежности) стоит пустое слово Λ, для которого справедливо $\Lambda a=a\Lambda=a$.

7. Теперь о матричных операциях. Первая матричная операция – сложение. Сложение матриц есть, фактически, объединение слов с учетом места слова в матричной структуре. Обратим внимание на ин-

тересный момент. В матричной алгебре всегда указывается, что размерности матриц при сложении должны быть одинаковы. От этого ограничения можно отказаться. Пусть B и C два подграфа, описывающие части графа A. Подграфы могут содержать разное количество вершин, висячие ребра в соответствующих матрицах смежности подграфов не отражены и не описаны. Но, тем не менее, никто не запрещает после согласования и приведения обозначений вершин к единой метрике объединить матрицы B и C. Такое обобщение матричного сложения позволяет в дальнейшем использовать программистскую идею «разделяй и властвуй» в матричных операциях. Никто не запрещает в одной матрице описывать, например, два несвязных графа, в не работать порознь с двумя матрицами.

8. Вторая матричная операция – умножение. Умножение матриц определяется традиционно, как это описано в учебной литературе, но вместо произведения элементов матриц нужно использовать конкатенацию слов, не меняя их порядка следования.

Предложенное рассмотрение матриц смежности, состоящих из строковых данных, позволяет переформулировать встречающиеся в учебной литературе алгоритмы (поиск в глубину, нахождение циклов, определение связности и др.). Нужно последовательно умножать матрицу смежности, состоящую из строковых данных, саму на себя (находить ее степени) и анализировать результаты. Такое умножение есть, фактически, конструирование объектов, требуемых в условиях задачи. Отказ от реальных умножений элементов матриц (в традиционной матричной алгебре, где элементами матриц являются числа) дает некоторый вычислительный выигрыш, однако при этом происходит возрастание количества конструируемых объектов (способов попасть из вершины i в вершину j за число шагов, равное показателю степени матрицы смежности). Если параллельно отфильтровывать ненужные слова (повторяющиеся или иные, ранее учтенные в алгоритмах), можно сократить затраты и на сложениях (объединениях). Известно, что матричные операции распараллеливаются. Поэтому, несмотря на проигрыш в одних операциях, можно получить выигрыш в других.

В заключение обратим внимание на добавление 1 к монографии [3]. Добавление носит название «Булевы методы в теории графов». В нем автор рассматривает ряд задач теории графов и использует не числовое, а буквенное обозначение ребер. Интересно его замечание при рассмотрении задачи о поиске путей в графе [3, с. 499]: «Львиная доля маршрутов в графе обусловлена не особенностями его строения, а неограниченной возможностью банальных «хождений взад-вперед», и можно ожидать, что переход к задаче выявления лишь тех маршрутов, которые являются простыми цепями, приведет к значительному упрощению элементов в матрицах». Сказанное согласуется со сделанным нами в предыдущем абзаце замечанием.

Материалы, представленные на конференцию, могут оказаться полезными как для организации учебного процесса по дискретной математике (поскольку предоставляют преподавателям возможность постановки разнообразных задач для практических занятий), так и для занятий по информатике, на которых отрабатываются известные алгоритмы на графах, а предложенные способы представления матриц смежности позволяют реализовывать их с некоторыми вариациями, отличающимися от предлагаемых в учебной литературе.

Библиографический список

1. Матрица смежности графа // Заголовок с экрана [Электронный ресурс] // Режим доступа: https://neerc.ifmo.ru/wiki/index.php?title=Матрица_смежности_графа.

2. Клейнберг Д., Тардос Е. Алгоритмы: разработка и применение. – СПб., 2016. – 800 с.

3. Зыков А.А. Основы теории графов. – М.: Вузовская книга, 2004. – 664 с.

УДК 372.851

Методы развития мышления студентов на занятиях по математике: технологии и переводы

Е.В. Богарова[1], Г.В. Кравченко[1], В.Н. Токарев[2]
[1]АлтГУ, г. Барнаул; [2]АлтГТУ, г. Барнаул

Цель развития мышления – научиться ставить правильные вопросы, переходить от одних вопросов к другим и таким образом получать ответ. Правильно ли решена задача нельзя определить по конечному ответу, т.к. путь к верному ответу может быть пройден и через ложные посылки, но логическое построение математики такого не допускает. У многих студентов данный вопрос вызывает недоумение, поэтому результаты тестовых работ не могут служить подтверждением сформированности математического мышления. Тесты, ориентируя на внешний результат, демотивируют развитие мыслительных операций анализа.

Задача преподавателя – помочь в овладении методами поиска информации, решения задач, а также привить вкус к решению сложных задач. Без предварительной подготовки подойти к решению задач не-