Об одном алгоритме для решения задачи восстановления последовательности исторических событий

Левшунов М. А., Миронов С. В., Сидоров С. П.,
Суворов А. Д., Файзлиев А. Р.

Саратовский национальный исследовательский
государственный университет
им. Н. Г. Чернышевского

2018г.

В древнегреческих полисах существовала выборная должность магистрата полиса, который занимался контролем производства амфор. Её избранный занимал на фиксированный срок (обычно один год) и не более одного раза.
В период античности, в некоторых древнегреческих полисах для перевозки на кораблях вина и оливкового масла использовались керамические амфоры. Чаще всего эта тара была одноразовой и, благодаря этому, ее производили в огромных количествах.
Многие амфоры маркировались одним или двумя клеймами, в которых фигурируют имена магистрата, занимавшего в это время описанную выше должность, и фабриканта, владевшего мастерской, в которой эта тара была произведена.

Одной из задач исторической науки является задача восстановления хронологической последовательности избрания магистратов. Рассмотрим ее.
Известно, что некоторые из них занимали должность раньше других, а также предполагая, что один магистрат исполнял обязанности только один срок и не был переизбран.
Также будем предполагать, что все найденные изделия предоставляют полную информацию, т.е. если некоторый фабрикант изготовлял амфоры в период выполнения обязанностей некоторым магистратом, то хотя бы одна такая амфора обнаружена археологами и отражена в базе данных.

Пусть задана бинарная матрица $M \cdot N$. Столбцы этой матрицы характеризуют магистратов, а строки – фабрикантов. Каждому чиновнику и каждом изготовителю тары поставим в соответствие некоторое число, при этом количество идентификаторов фабрикантов будет M, а количество идентификаторов магистратов – N. Если на пересечении строки i и столбца j стоит единица, то была найдена тара, изготовленная мастером с номером i тогда, когда у должности находился магистрат с номером j.

Пример

Пусть было найдено 6 амфор со следующими клеймами (первое число – номер фабриканта, второе – номер магистрата): 1-3, 1-1, 2-1, 2-2, 3-1, 4-3.

Магистраты / Фабриканты	1	2	3
1	1	0	1
2	1	1	0
3	1	0	0
4	0	0	1

Кроме того, задан массив пар, который характеризует уже известные отношения по времени избрания между некоторыми магистратами. Если задана пара (u, v), значит чиновник с номером u занимал должность раньше, чем чиновник с номером v.
Необходимо переставить столбцы матрицы в порядке, соответствующем предполагаемому порядку избрания магистратов.

Для решения поставленной задачи будем отталкиваться от предположения, что каждый фабрикант работал в некоторый непрерывный промежуток времени, который мог охватить сроки нескольких магистратов. Таким образом, в матрице необходимо добиться того, что в каждой строке все единицы образуют один непрерывный отрезок, а столбцы удовлетворяют дополнительным условиям.

Из этого предположения вытекает, что два магистрата занимали должность в соседние промежутки времени тогда и только тогда, когда максимизировано количество фабрикантов, которые работали при обоих этих чиновниках. Обозначим это количество как функцию $f(u, v)$. Для общности будем считать, что для $\forall u$ $f(u, u) = 0$.

Тогда наиболее оптимальный ответ возникает в том случае, когда максимизирована сумма этой функции по всем парам соседних магистратов. То есть среди всех последовательностей избрания $u_1, u_2, \ldots, u_N$, выбирается такая, для которой выполняется следующее:

$$\sum_{i=1}^{N-1} f(u_i, u_{i+1}) \to \max \tag{1}$$

Для большей наглядности, представим данные в виде полного взвешенного графа из N вершин, где вершинами будут метки магистратов, а вес ребра между вершинами u и v – значение функции $f(u, v)$. Петли нулевого веса в этот граф добавлять не будем, на ответ они все равно не повлияют.
В таком графе ответом будет путь, который проходит через каждую вершину графа ровно один раз и при этом сумма весов ребер на этом пути максимальна.

В теории графов такой путь называется гамильтоновым, а задача максимизации суммы весов ребер на нем является частным случаем задачи коммивояжера. Эта задача относится к классу NP-трудных задач. Для ее решения существует множество быстрых, но вероятностных алгоритмов, однако мы воспользуемся медленным, но точным алгоритмом на основе динамического программирования.

Ограничение на заданные отношения

Кроме всего перечисленного, необходимо учесть, что для некоторых пар магистратов порядок избрания уже известен. При этом очевидно, что заданный массив пар должен определять строгий частичный порядок на множестве магистратов. Соответственно, должны иметь место антирефлексивность, антисимметричность и транзитивность. В случае невыполнения любого из этих условий, решения для заданных входных данных не существует.

Для решения полученной задачи используем алгоритм Беллмана-Хелда-Карпа с некоторыми дополнениями:

- Алгоритм решает задачу коммивояжера для случая, когда стартовая вершина известна. У нас же она не задана, поэтому необходимо перебрать все N вершин и запустить алгоритм для каждой.
- Перед каждым запуском алгоритма будем проверять, что в заданном отношении порядка нет ни одной пары, в которой текущая стартовая вершина стоит на втором месте. Если это условие не выполняется, сразу переходим к рассмотрению следующей вершины.
- Во время работы алгоритма при каждом переходе в очередную вершину v необходимо проверить, что нет ни одной пары, в которой на первом месте содержится еще не посещенная вершина, а на втором месте – вершина v.

Пусть были найдены клейма 15 различных магистратов и 13 фабрикантов.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1		1	1					1					1		
2			1					1		1	1				
3		1						1					1		
4	1					1	1							1	1
5									1			1			
6	1						1		1						
7						1								1	1
8			1							1	1				
9										1	1			1	
10						1					1			1	1
11				1	1		1		1			1			
12	1					1									
13	1					1	1		1						1

Пример работы

Результат после работы программы и перестановки магистратов в найденном порядке.

	2	13	8	3	10	11	14	15	6	1	7	9	12	4	5
3	1	1	1												
1	1	1	1	1											
2			1	1	1	1									
8				1	1	1									
9					1	1	1								
10						1	1	1	1						
7							1	1	1						
4							1	1	1	1	1				
13								1	1	1	1	1			
12									1	1					
6										1	1	1			
11											1	1	1	1	1
5												1	1		

На каждом шаге алгоритма необходимо решить набор из $N \cdot 2^N$ подзадач, на решение каждой из которых необходимо время, пропорциональное $O(N+k)$, где k – количество начальных условий. Шагов всего так же N, таким образом решение работает за асимптотику $O(N^2 \cdot (N+k) \cdot 2^N)$. Однако за счет начальных условий, очень большое количество шагов алгоритма отсекается и на реальных данных алгоритм работает на порядок быстрее.
Кроме того, решение требует $O(N^2)$ дополнительной памяти на сохранение графа и еще $O(N \cdot 2^N)$ на сохранение массива состояний. Это приводит к тому, что уже при $N = 22$ на решение поставленной задачи требуется чуть больше одного гигабайта памяти. И при увеличении N это число растет на порядок.

Итоги

Было получено точное решение поставленной задачи, время работы и требуемая дополнительная память которого зависят в первую очередь от количества магистратов.
В свою очередь на время работы также влияет количество начальных условий. Несмотря на то, что в асимптотике указана прямая зависимость от их количества, на деле она является обратной, так как каждое дополнительное условие позволяет отсекать большое количество возможных вариантов.
Количество фабрикантов никак не влияет на итоговое время работы. Оно важно только при считывании данных и преобразовании их в граф.
Разработанный алгоритм позволяет решать не только аналогичные задачи, но и задачи из любой области, где необходима хронологизация данных.

Список использованных источников

Н. Ф. Федосеев
О хронологии синопских керамических клейм
Античный мир и археология, Саратов. — 2015 — № 17. — С. 352–364.

N. F. Fedoseev
Classification des timbres astynomiques de Sinope
Production et Commerce des Amphores anciennes en Mer Noire, Provence. — 1999 — Pp. 27–48.

Ф. Харари, В. П. Козырев, Г. П. Гаврилов
Теория графов
М.: Мир: 1973.

Т. Кормен,Ч. Лейзерсон, Р. Ривест, К. Штайн
Алгоритмы. Построение и анализ.
М.: Издательский дом Вильямс: 2009.

СПАСИБО ЗА ВНИМАНИЕ!