О сложности восстановления частичного порядка

А. А. Абдель Маджид (МГУ имени М. В. Ломоносова, Москва)

Работа посвящена вопросу сложности восстановления частичного порядка на конечном множестве, если известно, что частичный порядок обладает какими-то наперед заданными свойствами. Рассматривается модель вычислений в которой под сложностью понимается количество сравнений, необходимое для восстановления частичного порядка. Исследуются классы частично упорядоченных множеств, для которых задача восстановления порядка решается меньше чем за квадратичное число сравнений относительно количества элементов в множестве.

Ключевые слова: частичный порядок, конечное множество, восстановление порядка, сортировка, сравнение.

Основные понятия и постановка задачи

Нас будут интересовать конечные множества с заданным на нем частичным порядком $(P, \preceq)$. Частичный порядок $\preceq$ является бинарным отношением, который удовлетворяет $\forall x, y, z \in P$ следующим аксиомам:

1) $x \preceq x$ (рефлексивность)
2) $x \preceq y \wedge y \preceq x \Rightarrow x = y$ (антисимметричность)
3) $x \preceq y \wedge y \preceq z \Rightarrow x \preceq z$ (транзитивность)

Два элемента и называются сравнимыми, если имеет место $x \preceq y$ или $y \preceq x$. В противном случае — несравнимыми. Графическим способом задания частично упорядоченного множества служит диаграмма Хассе. Это ориентированный граф, в котором вершинам соответствуют элементы множества P, а две вершины x и y соединены ориентированным ребром точно тогда, когда $x \preceq y$ и не существует элемента z такого, что $x \preceq z$ и $z \preceq y$, то есть является транзитивно редуцированным графом. В случае конечных множеств частичный порядок однозначно задается диаграммой Хассе, а диаграмма Хассе однозначно строится по частичному порядку.

Общая постановка задачи

На множестве задан частичный порядок. Такие множества будет называть частично упорядоченными множествами, далее просто ЧУМ. Пусть ЧУМ обладает некоторыми свойствами. Мы можем задавать вопросы вида $x \preceq y$ на который получим ответ «да», если это верно и «нет» иначе. Восстановить частичный порядок значит для любых двух элементов x, y понять верно ли, что $x \preceq y$, $y \preceq x$ или несравнимы. Сколько сравнений необходимо выполнить, чтобы восстановить этот частичный порядок? Под сложностью будем понимать количество сравнений, которое необходимо выполнить, чтобы восстановить частичный порядок. Будем считать, что $\log n$ обозначает логарифм по основанию 2.

Случай линейного порядка

Частным случаем частичного порядка является линейный порядок — случай, когда любые два элемента сравнимы. Если частичный порядок — линейный порядок, то задача сводится к задаче сортировки.

Лемма 1. *В случае линейного порядка для $(P, \leqslant), n = |P|$ достаточно выполнить не более $n \log n$ сравнений.*

Общий случай частичного порядка

Теперь ответим на вопрос, сколько сравнений понадобится в общем случае в худшем случае.

Лемма 2. *Для восстановления частичного порядка на $(P, \leqslant), n = |P|$ достаточно $n(n-1)$ сравнений.*

Пусть нам нужно восстановить частичный порядок на множестве, где либо все элементы несравнимы, либо только какие-то два сравнимы. На таком множестве достигается оценка из леммы.

Деревья решений и нижние оценки

Определение 1. Дерево решений — это бинарное дерево, в котором представлены операции сравнений элементов. В нашем случае в узлах дерева будет производиться сравнение элементов, и левая ветвь будет обозначать что ответ на сравнение «да», а правая ветвь — ответ «нет». Каждый лист представляет собой окончательное упорядочение элементов.

Выполнение алгоритма соответствует прохождению от корня дерева до одного из его листьев. Все алгоритмы для восстановления частичного порядка представляют собой деревья решений.

Определение 2. Числом конфигураций назовем количество частичных порядков на множестве, изоморфных заданному.

Заметим, что для $(P, \leqslant), n = |P|$ количество конфигураций не превосходит $n!$. Мощностную нижнюю оценку можно получить как $\log n$ от числа конфигураций для данного ЧУМ. Приведем некоторые примеры, будем считать что во всех примерах количество элементов равно n:

1) Для линейного порядка положение каждого элемента является уникальным, поэтому число конфигураций равняется $n!$.
2) Рассмотрим порядок когда все n элементов несравнимы. Тогда количество конфигураций равно одному.
3) Порядок где либо все элементы несравнимы, либо только какие-то два сравнимы. Необходимо соединить какие-то 2 элемента ребром. n для выбора первого элемента и $n-1$ для второго. Итого $n(n-1)$ конфигураций.

Алгоритм, основывающийся на величине антицепи

Определение 3. [1] Множество в котором любые два элемента несравнимы называется антицепью.

Пусть дано ЧУМ $(P, \leqslant), n = |P|$ и известно что максимальная антицепь не превосходит k.

Теорема 1. *Алгоритм $\mathcal{ACP}$(Anti Chain Poset) восстанавливает порядок не больше чем за $2nk \log \frac{n}{k}$ сравнений.*

Следствие 1. *Если максимальная антицепь k в $(P, \leqslant), n = |P|$ ограничена $o(\frac{n}{\log n})$, то порядок алгоритмом $\mathcal{ACP}$ восстанавливается за $o(n^2)$ сравнений.*

В силу этого следствия мы можем восстанавливать частичный порядок для довольного большого класса частично упорядоченных множества меньше чем за квадратичное количество сравнений.

Деревья

В данном разделе будем рассматривать ЧУМ, диаграммы которых имеют вид корневых деревьев с ограниченным ветвлением. Будем считать, что ветвление ограничено двумя. Случай деревьев интересен тем, что, например, для регулярного дерева, его антицепь есть $O(n)$ и для алгоритма $AntichainPoset$ получается квадратичное число сравнений.

Наивный алгоритм для деревьев

Пусть $n = |T|$. Тогда количество элементов в самой длинной цепи обозначим через m, которое будем выражать через n.

Теорема 2. *Количество сравнений для восстановления порядка на дереве для наивного алгоритма $\mathcal{NTP}(Naive\ Tree\ Poset)$ не превосходит $3nm$.*

Следствие 2. *Если количество элементов самой длинной цепи m в дереве с n элементами ограничено $o(n)$. Тогда количество сравнений для восстановления порядка на нем составляет $o(n^2)$.*

Следствие 3. *Если T регулярное дерево длина цепи в нем равна $\log n$ и порядок на нем восстанавливается не больше чем за $3n \log n$ сравнений.*

Деревья, имеющие цепи одинаковой длины

Будем рассматривать алгоритм для класса корневых деревьев, имеющих все цепи одинаковой длины.

Пусть T ЧУМ в виде корневого дерева где всякая цепь имеет размер m, а количество элементов в дереве n.

Теорема 3. *Количество сравнений алгоритма $\mathcal{ETP}(Equal\ Tree\ Poset)$ есть $O(n^{3/2} \log^2 m)$.*

Следствие 4. *Пусть ветвление дерева не превосходит $O(\log^k n)$, тогда порядок на нем восстанавливается за $O(n^{3/2} \log^{2+k} m)$.*

Деревья, имеющие регулярные константные деревья

Пусть $n = |T|$. Рассмотрим случай, когда из цепи длины $m = \frac{n}{c}$ и из некторых элементов цепи выходят деревья константной длины. В этом пункте имеет дело со случаем, когда из каждого элемента цепи выходит регулярное дерево одной и той же высоты k. Тогда общее число элементов есть $n = md$, где $d = 2^k - 1$.

Теорема 4. *Количество сравнений алгоритма $\mathcal{RTCP}$ (Regular Tree Chain Poset) есть $O(n \log n)$.*

Деревья с константными отростками

В этом пункте имеет дело со случаем, когда из каждого элемента цепи выходит произвольное дерево, число элементов которого есть константа.

Теорема 5. *Количество сравнений алгоритма $\mathcal{CTCP}$ (Const Tree Chain Poset) есть $O(n \log n)$.*

Древовидное семейство решеток

Рассмотрим семейство решеток, которое строится из одного элемента и следующих операций, примененных к уже построенным P и Q:

1) Последовательное соединение P и Q. Проводится ребро от наименьшего элемента P к наибольшему элементу Q.
2) Параллельное соединение P и Q. P и Q должны иметь максимальной цепи одной длины. Добавляется наибольший и наименьший элемент. От наибольшего выходят два ребра, одно идет к наибольшему элементу P, другое к наибольшему элементу Q. От наименьших элементов P и Q проводится ребро к наименьшему элементу.

Теорема 6. *Алгоритм $\mathcal{TLL}(Tree\ Like\ Lattice)$ восстанавливает порядок на $P, n = |P|$, за $O(n^{3/2} \log n)$ сравнений.*

Декартово произвдение

Определение 4. [1] Декартовым произведением $(P, \preceq), n = |P|$ и $(Q, \preceq), m = |Q|$, $P \times Q$ называется ЧУМ $R = \{(x, y) | x \in P, y \in Q\}$, что $(x_1, y_1) \preceq (x_2, y_2)$, когда $x_1 \preceq x_2$ в P и $y_1 \preceq y_2$ в Q.

$P \times Q$ и $Q \times P$ изоморфны, хотя их диаграммы могут выглядеть по-разному.

Рассмотрим P — цепь из двух элементов и антицепь Q из n элементов. Тогда $P \times Q$ и мы получим диаграмму для восстановления порядка которой потребуется $O(n^2)$ сравнений.

Теорема 7. *Если* $(P, \leqslant), n = |P|$, *P — цепь, а* $(Q, \leqslant), m = |P|$ *и порядок на Q восстанавливается за $O(n \log n)$ и Q имеет наибольший элемент, то порядок на $P \times Q$ восстанавливается за $O(nm \log nm)$ сравнений.*

Заключение

Как указано в [2], множество всех ЧУМ структурно необозримо и неизвестно никакой удобной его характеризации. Поэтому исследуются классы, обладающими какими-то наперед заданным свойствами. И как было отмечено в работе класс, для которых порядок не восстанавливается меньше чем за квадратичное число сравнений не пуст. Но кое-что про произвольный частичный порядок сказать можно. Например, из [3] известно, что максимальные цепи почти каждого частичного порядка имеют одну и туже длину три.

Возможен подход, когда мы применяем к классам ЧУМ на которых мы умеем восстанваливать порядок меньше чем за квадратичное время какие-то операции. Такой подход был продемонстрирован на примере декартового произведения.

Пусть дерево содержит n элементов. Остается открытым вопрос: «можно ли восстановить порядок меньше чем за квадратичное число сравнений для класса деревьев с ограниченным ветвлением?» Гипотеза автора заключается в том, что это возможно сделать. В данной работе эта задача решена для частного случая, когда максимальная антицепь ограничена $o(\frac{n}{\log n})$ и для случая когда число элементов в самой длиной цепи дерева ограничено $o(n)$. Если же в дереве только одна цепь растет с ростом n, то задача решается даже за $O(n \log n)$.

Автор благодарит своего научного руководителя с.н.с Жука Д. Н. за постановку задачи и помощь в работе. Также выражает благодарность к.ф.-м.н Галатенко А. В. за помощь в работе и ценные замечания.

Список литературы

[1] Стенли Р. Перечислительная комбинаторика. Том. 1. — М.: Мир, 1990.

[2] Гуров С. И. Булевы алгебры, упорядоченные множества, решетки: определения, свойства, примеры. — М.: Либроком, 2013.

[3] Kleitman D. J., Rothschild B. L., Kelly B. L. Asymptotic enumeration of partial orders on a finite set // Trans. Amer. Math. Soc. — 1975. — 205. — P. 205–220.