Комбинаторные алгоритмы

Задача о полном паросочетании. Алгоритм Куна

Гальперин Александр Леонидович

2018 г.

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

В этом разделе мы будем рассматривать двудольные графы $G + (X, Y, E)$, в которых множества X и Y имеют одинаковое число вершин. Пусть $|E| = m$, $|X| = |Y| = n$.

Определение

Паросочетание, насыщающее все вершины данного двудольного графа, называется *полным* или *совершенным*.

В этом разделе мы будем рассматривать двудольные графы $G + (X, Y, E)$, в которых множества X и Y имеют одинаковое число вершин. Пусть $|E| = m$, $|X| = |Y| = n$.

Определение

Паросочетание, насыщающее все вершины данного двудольного графа, называется *полным* или *совершенным*.

Определение

Задача, в которой требуется построить полное паросочетание, если оно существует, называется *задачей о полном паросочетании*.

Теорема 6 (Холл, 1935)

В двудольном графе $G = (X, Y, E)$ *полное паросочетание существует тогда и только тогда, когда для любого* $S \subseteq X$ *справедливо неравенство*

$$|S| \leqslant |E(S)|,$$

где $E(S)$ *— множество всех вершин из* Y*, смежных с некоторыми вершинами из* X*.*

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

С точки зрения алгоритмов ценность этой теоремы невелика. Действительно, для проверки выполнения условия существования полного паросочетания требуется просмотреть 2^n подмножество множества X.

С точки зрения алгоритмов ценность этой теоремы невелика. Действительно, для проверки выполнения условия существования полного паросочетания требуется просмотреть 2^n подмножество множества X.

Более того, теорема не дает никакого конструктивного метода построения полного паросочетания.

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

Одним из возможных методов решения задачи о полном паросочетании могло бы быть применение алгоритма Хопкрофта–Карпа и выделение полного паросочетания: если это паросочетание состоит из n ребер — оно полное, а если в нем меньше, чем n ребер, полного паросочетания в этом двудольном графе нет.

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

Рассмотрим теперь алгоритм, который завершает работу либо построением полного паросочетания, либо в тот момент (а он может наступить достаточно рано), когда станет ясно, что полного паросочетания в данном графе нет.

Этот алгоритм составляет существенную часть метода, разработанного Харольдом Уильямом Куном в 1955 году для решения более общей задачи — задачи о назначениях.

Кун назвал свой метод *Венгерским алгоритмом*. Алгоритм построения полного паросочетания мы будем называть *алгоритмом Куна*.

Неформально алгоритм Куна можро изложить следующим образом.

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

Неформально алгоритм Куна можро изложить следующим образом.

1. Пустое паросочетание объявить текущим.

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

Неформально алгоритм Куна можро изложить следующим образом.

1. Пустое паросочетание объявить текущим.
2. Если все вершины из X насыщены в M, тогда СТОП (M — полное паросочетание).

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

Неформально алгоритм Куна можро изложить следующим образом.

1. Пустое паросочетание объявить текущим.
2. Если все вершины из X насыщены в M, тогда СТОП (M — полное паросочетание).
3. Иначе выбрать произвольную свободную вершину $x \in X$ и искать M–чередующуюся цепь, начинающуюся в x.

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

Неформально алгоритм Куна можро изложить следующим образом.

1. Пустое паросочетание объявить текущим.
2. Если все вершины из X насыщены в M, тогда СТОП (M — полное паросочетание).
3. Иначе выбрать произвольную свободную вершину $x \in X$ и искать M–чередующуюся цепь, начинающуюся в x.
4. если такая цепь P найдена, то положить $M = M \oplus P$ и вернуться на шаг 2.

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

Неформально алгоритм Куна можро изложить следующим образом.

1. Пустое паросочетание объявить текущим.
2. Если все вершины из X насыщены в M, тогда СТОП (M — полное паросочетание).
3. Иначе выбрать произвольную свободную вершину $x \in X$ и искать M–чередующуюся цепь, начинающуюся в x.
4. если такая цепь P найдена, то положить $M = M \oplus P$ и вернуться на шаг 2.
5. Иначе СТОП (полного паросочетания в заданном графе не существует).

Разберем предложенный алгоритм подробнее.

- Для поиска M-чередующейся цепи можно использовать как поиск в глубину, так и поиск в ширину. В данном случае удобнее поиск в глубину.
- Правила поиска те же самые, что и в алгоритмах Форда–Фалкерсона и Хопкрофта–Карпа: переход из вершин $x \in X$ к вершинам $y \in Y$ осуществляется по светлым ребрам, а от $y \in Y$ к $x \in X$ — по темным.

Как всегда возможны два исхода поиска.

- либо будет найдена свободная вершина $y \in Y$, т.е. найдена M–чередующаяся цепь, Как всегда возможны два исхода поиска.

- либо будет найдена свободная вершина $y \in Y$, т.е. найдена M–чередующаяся цепь,
- либо чередующейся цепи с началом в корневой вершине поиска не существует.

В первом случае действия стандартны:

- увеличиваем текущее паросочетание с помощью найденной цепи;

В первом случае действия стандартны:

- увеличиваем текущее паросочетание с помощью найденной цепи;
- начинаем искать M–чередующуюся цепь из другой свободной вершины.

Второй исход интереснее.

Второй исход интереснее.

Пусть поиск в глубину начался из вершины x и M–чередующаяся цепь не была найдена.

Второй исход интереснее.

Пусть поиск в глубину начался из вершины x и M–чередующаяся цепь не была найдена.

Тогда дерево поиска выглядит следующим образом.

1. Вершина x находится в корне дерева поиска.

Второй исход интереснее.

Пусть поиск в глубину начался из вершины x и M–чередующаяся цепь не была найдена.

Тогда дерево поиска выглядит следующим образом.

1. Вершина x находится в корне дерева поиска.
2. Все вершины, уровень которых в дереве поиска есть число нечетное, принадлежат Y, а все вершины с четным уровнем — X. Причем, все вершины в дереве за исключением x насыщены в M.

Второй исход интереснее.

Пусть поиск в глубину начался из вершины x и M–чередующаяся цепь не была найдена.

Тогда дерево поиска выглядит следующим образом.

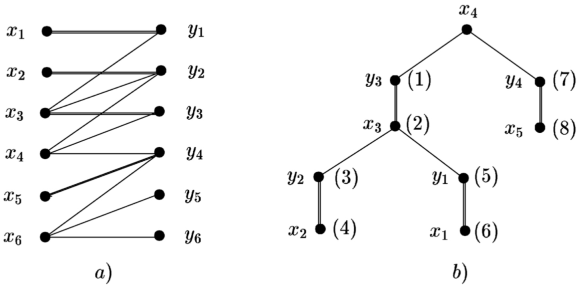
1. Вершина x находится в корне дерева поиска.
2. Все вершины, уровень которых в дереве поиска есть число нечетное, принадлежат Y, а все вершины с четным уровнем — X. Причем, все вершины в дереве за исключением x насыщены в M.
3. Кроме того, все ребра, исходящие из вершин с нечетным уровнем, соответствуют ребрам паросочетания M, а все прочие — ребрам, не входящим в M.

Такое дерево часто называют *венгерским* или *чередующимся* деревом.

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

На рисунке приведен пример графа и паросочетания в нем, а также дерево поиска в глубину из вершины x_4. Ребра паросочетания изображены утолщенными линиями. Числа в скобках соответствуют порядку просмотра в ходе поиска.



Обозначим через S множество всех вершин дерева поиска, уровень k которых является четным числом. Тогда $S \subseteq X$.

Обозначим через S множество всех вершин дерева поиска, уровень k которых является четным числом. Тогда $S \subseteq X$.

Пусть $E(S)$ — множество всех $y \in Y$, смежных с вершинами из S. Покажем, что все вершины $E(S)$ попадают в дерево поиска.

Пусть $\tilde{y} \in E(S)$. Если вершина $\tilde{y}$ смежна с корневой вершиной поиска x, то соединяющее их ребро является светлым, т.к. x — свободная вершина.

Пусть $\tilde{y} \in E(S)$. Если вершина $\tilde{y}$ смежна с корневой вершиной поиска x, то соединяющее их ребро является светлым, т.к. x — свободная вершина.

Тогда по правилам поиска вершина $\tilde{y}$ непременно будет помечена и, следовательно, $\tilde{y}$ входит в дерево поиска.

Пусть $\tilde{y} \in E(S)$. Если вершина $\tilde{y}$ смежна с корневой вершиной поиска x, то соединяющее их ребро является светлым, т.к. x — свободная вершина.

Тогда по правилам поиска вершина $\tilde{y}$ непременно будет помечена и, следовательно, $\tilde{y}$ входит в дерево поиска.

Если $\tilde{y}$ смежна с насыщенной вершиной $\tilde{x} \in S$, то $\tilde{x}$ попадает в дерево поиска только после того, как туда попадает смежная с ней по темному ребру вершина y^*.

Пусть $\tilde{y} \in E(S)$. Если вершина $\tilde{y}$ смежна с корневой вершиной поиска x, то соединяющее их ребро является светлым, т.к. x — свободная вершина.

Тогда по правилам поиска вершина $\tilde{y}$ непременно будет помечена и, следовательно, $\tilde{y}$ входит в дерево поиска.

Если $\tilde{y}$ смежна с насыщенной вершиной $\tilde{x} \in S$, то $\tilde{x}$ попадает в дерево поиска только после того, как туда попадает смежная с ней по темному ребру вершина y^*.

Поскольку M — паросочетание, имеем $\tilde{y} = y^*$.

Отсюда вытекает, что $\tilde{y}$ была помечена раньше чем x.

Отсюда вытекает, что $\tilde{y}$ была помечена раньше чем x.

Тем самым проверено, что $E(S)$ содержится в дереве поиска.

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

Пусть k — нечетно. Тогда из каждой вершины y уровня k в дерево поиска входит ровно одна дуга.

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

Пусть k — нечетно. Тогда из каждой вершины y уровня k в дерево поиска входит ровно одна дуга.

Следовательно, если число k нечетно, то количество вершин, имеющих уровень k, равно количеству вершин уровня $k + 1$.

Пусть k — нечетно. Тогда из каждой вершины y уровня k в дерево поиска входит ровно одна дуга.

Следовательно, если число k нечетно, то количество вершин, имеющих уровень k, равно количеству вершин уровня $k+1$.

Из приведенных рассуждений следует, что

$$|S| = |E(S)| + 1,$$

т.к. корневая вершина дерева поиска является «лишней».

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

Пусть k — нечетно. Тогда из каждой вершины y уровня k в дерево поиска входит ровно одна дуга.

Следовательно, если число k нечетно, то количество вершин, имеющих уровень k, равно количеству вершин уровня $k+1$.

Из приведенных рассуждений следует, что

$$|S| = |E(S)| + 1,$$

т.к. корневая вершина дерева поиска является «лишней».

Отсюда получаем, что $|S| > E(S)$. Следовательно, граф G не имеет полного паросочетания в силу теоремы Холла.

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

Проведенный анализ показывает, что, осуществляя поиск в глубину, мы находимся в беспроигрышной ситуации:

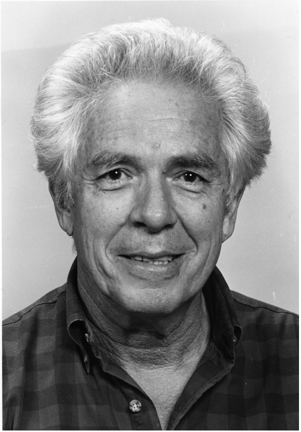
- либо поиск завершится нахождением чередующейся цепи, и тогда текущее паросочетание можно увеличить;

Проведенный анализ показывает, что, осуществляя поиск в глубину, мы находимся в беспроигрышной ситуации:

- либо поиск завершится нахождением чередующейся цепи, и тогда текущее паросочетание можно увеличить;
- либо такая цепь не будет найдена, и тогда можно остановить работу алгоритма, ибо полного паросочетания не существует.

Харольд Уиллиам Кун

(29 июля 1925 года – 2 июля 2014 года)



Американский математик. Специалист в области теории игр. Профессор Принстонского университета. Автор Венгерского алгоритма (**1955 год**) и соавтор теоремы Куна–Таккера о необходимых условиях оптимальности в задаче математического программирования. В 1980 году удостоен приза Джона фон Неймана.

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

- При формальной записи алгоритма предполагается, что двудольный граф $G = (X, Y, E)$ задан матрицей смежности $A[1 \dots n, 1 \dots n]$.
- Текущее паросочетания описывается двумя массивами $Xdouble$ и $Ydouble$ длины n каждый.

Напомним, что $Xdouble[x] = y$, если x сочетается с y.

$Xdouble[x] = nil$, если x — свободная вершина относительно паросочетания M.

Структура алгоритма Куна следующая.

- T — множество свободных вершин относительно текущего паросочетания.

Структура алгоритма Куна следующая.

- T — множество свободных вершин относительно текущего паросочетания.
- $Start(T)$ — функция, которая возвращает для множества $T \neq \varnothing$ произвольный элемент.

Структура алгоритма Куна следующая.

- T — множество свободных вершин относительно текущего паросочетания.
- $Start(T)$ — функция, которая возвращает для множества $T \neq \varnothing$ произвольный элемент.
- Переменная *indication* равна нулю, если свободная вершина $y \in Y$ еще не встретилась, и равна единице, как только достигается какая–нибудь свободная вершина $y \in Y$.

Структура алгоритма Куна следующая.

- T — множество свободных вершин относительно текущего паросочетания.
- $Start(T)$ — функция, которая возвращает для множества $T \neq \varnothing$ произвольный элемент.
- Переменная *indication* равна нулю, если свободная вершина $y \in Y$ еще не встретилась, и равна единице, как только достигается какая–нибудь свободная вершина $y \in Y$.
- Функция $Choice(x)$ возвращает произвольную смежную с x вершину $y \in Y$, не посещавшуюся в ходе поиска из вершины x. Если такой вершины нет, то $Choice(x) = nil$. Отметим, что при реализации этой функции не обойтись без переменной, указывающей на то, посещалась или нет данная вершина.

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

Вход: двудольный граф $G=(X, E, Y)$, заданный матрицей $A[1 \ldots n, 1 \ldots n]$, где $n = |X| = |Y|$.

Выход: полное паросочетание, задаваемое массивами $X double$ и $Y double$, либо сообщение о том, что такого паросочетания не существует.

```
begin
  T := X;
  for x ∈ X  do  Xdouble[x] := nil;
  for y ∈ Y  do  Ydouble[y] := nil;
  repeat
    S := nil; x := Start(T); S ⇐ x; indication := 0;
    while (S ≠ nil) and (indication = 0) do
      begin
        x ⇐ S; y := Choice(x);
        if y ≠ nil then
          begin
            S ⇐ y; z := Ydouble[y];
            if z ≠ nil then S ⇐ z;
            else indication = 1;
          end
        else
          begin
             ⇐ S;
            if S ≠ ∅ then   ⇐ S;
          end
      end;
    if indication = 1 then
      while S ≠ nil do
        begin
          x ⇐ S; y ⇐ S; T := T \ {x};
          Xdouble[x] := y; Ydouble[y] := x;
        end
  until (indication = 0) or (T = ∅);
end.
```

Проанализируем работу алгоритма.

Проанализируем работу алгоритма.

В строках 3–4 инициализируется пустое паросочетание

3. **for** $x \in X$ **do** $Xdouble[x] := nil$;
4. **for** $y \in Y$ **do** $Ydouble[y] := nil$;

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

В сроках 6–21 осуществляется поиск в глубину из свободной вершины x.

```
6.   S := nil; x := Start(T); S ⇐ x; indication := 0;
7.   while (S ≠ nil) and (indication = 0) do
8.     begin
9.       x ⇐ S; y := Choice(x);
10.      if y ≠ nil then
11.        begin
12.          S ⇐ y; z := Ydouble[y];
13.          if z ≠ nil then S ⇐ z;
14.          else indication = 1;
15.        end
16.      else
17.        begin
18.            ⇐ S;
19.          if S ≠ ∅ then   ⇐ S;
20.        end
21.    end;
```

$\mathcal{NB}$

Эти строки являются почти точной копией аналогичных строк в процедуре *Increase*(M) из предыдущего раздела.

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

Строки 6–21 показывают, что в стек вершины добавляются парами.

```
6.  S := nil; x := Start(T); S ⇐ x; indication := 0;
7.  while (S ≠ nil) and (indication = 0) do
8.    begin
9.      x ⇐ S; y := Choice(x);
10.     if y ≠ nil then
11.       begin
12.         S ⇐ y; z := Ydouble[y];
13.         if z ≠ nil then S ⇐ z;
14.         else indication = 1;
15.       end
16.     else
17.       begin
18.          ⇐ S;
19.         if S ≠ ∅ then  ⇐ S;
20.       end
21.   end;
```

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

Использованные в ходе поиска вершины удаляются из S тоже парами.

$\Leftarrow S$;
if $S \neq \emptyset$ **then** $\Leftarrow S$;

Строка 14 показывает, что сразу по достижении свободной вершины (т.е. такой вершины, для которой $Ydouble = nil$), переменная *indication* становится равной единице.

$$14. \qquad \textbf{else } indication = 1;$$

После этого (условие в строке 7) процесс поиска из вершины x сразу останавливается.

7. **while** $(S \neq nil)$ and $(indication = 0)$ **do**

После завершения поиска в строке 22 анализируется, чем именно закончился поиск из данной вершины.

$$22. \qquad \textbf{if } indication = 1 \textbf{ then}$$

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

В том случае, если поиск завершился достижением свободной вершины, в строках 23–27 происходит увеличение паросочетания.

```
23.     while S ≠ nil do
24.       begin
25.         x ⇐ S; y ⇐ S; T := T \ {x};
26.         Xdouble[x] := y; Ydouble[y] := x;
27.       end
```

В строке 28 приведены два возможных исхода работы алгоритма Куна.

28. **until** ($indication = 0$) or ($T = \emptyset$);

В строке 28 приведены два возможных исхода работы алгоритма Куна.

28. **until** $(indication = 0)$ **or** $(T = \emptyset)$;

Понятно, что условие $T = \varnothing$ означает, что все вершины множества X насыщены, т.е. текущее паросочетание является полным.

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

В строке 28 приведены два возможных исхода работы алгоритма Куна.

$$28. \qquad \textbf{until } (indication = 0) \text{ or } (T = \emptyset);$$

Понятно, что условие $T = \varnothing$ означает, что все вершины множества X насыщены, т.е. текущее паросочетание является полным.

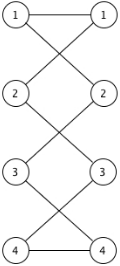
Если поиск из какой-либо вершины не завершился нахождением свободной вершины y, т.е. случай $indication = 0$, то первое условие в сроке 28 обеспечивает остановку алгоритма (полного паросочетания не существует).

Т.к. поиск в глубину из заданной вершины имеет сложность $O(n^2)$ и основной цикл 5–28 алгоритма выполняется не более n раз, справедлива

Теорема 7

Алгоритм Куна имеет сложность $O(n^3)$.

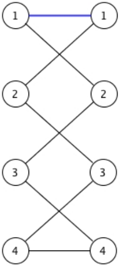
Рассмотрим работу алгоритма Куна на примере. Пусть дан двудольный граф G



Объявляем пустое паросочетание текущим:

$$M = \varnothing$$

Будем просматривать вершины из доли X в естественном порядке.

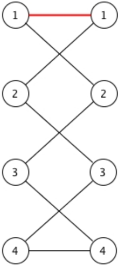


Находим первую M–чередующуюся цепь $P : 1 - 1$.

Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

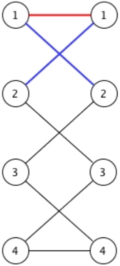
Выполним операцию $M \oplus P$ (переключаем цвета).



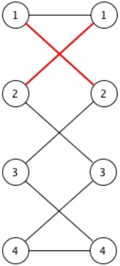
Новое текущее паросочетание

$$M = \{1 - 1\}.$$

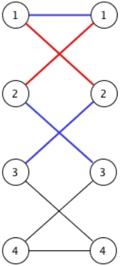
Находим M–чередующуюся цепь P: $2 - 1 - 1 - 2$.



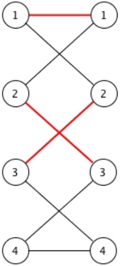
Выполним операцию $M \oplus P$ (переключаем цвета).
Новое текущее паросочетание: $M = \{2 - 1, 1 - 2\}$.



Находим M-чередующуюся цепь P: $3 - 2 - 1 - 1 - 2 - 3$.



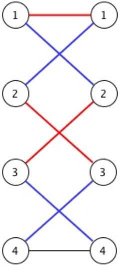
Выполним операцию $M \oplus P$ (переключаем цвета).
Новое текущее паросочетание $M = \{1 - 1, 2 - 3, 3 - 2\}$.



Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

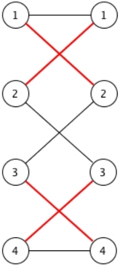
Находим М-чередующуюся цепь P: $4 - 3 - 2 - 1 - 1 - 2 - 3 - 4$.



Паросочетания в двудольных графах

Задача о полном паросочетании. Алгоритм Куна

Выполним операцию $M \oplus P$ (переключаем цвета).
Новое текущее паросочетание $M = \{1-2, 2-1, 3-4, 4-3\}$.



NB

Алгоритм Куна может быть реализован и на основе поиска в ширину. Все принципиальные моменты построения алгоритма при этом сохраняются.