Закономерности десятичной кодировки деревьев
и числа-деревья

Кирсанов М.Н.

Деревья представляют собой важный вид графов. С помощью деревьев описываются базы данных, деревья моделируют алгоритмы и программы, их используют в электротехнике, химии. Одной из актуальных задач в эпоху компьютерных и телекоммуникационных сетей является задача сжатия информации. Сюда входит и кодировка деревьев. Компактная запись дерева, полностью описывающая его структуру, может существенно упростить как передачу информации о дереве, так и работу с ним. Различные виды кодировки деревьев подробно описаны в [1].

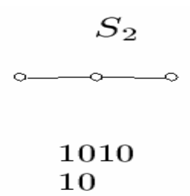
Приведем одну из простейших кодировок помеченных деревьев с выделенным корнем — десятичную. Кодируя дерево, придерживаемся следующих правил.

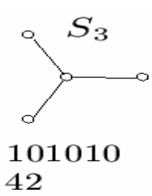
1. Кодировка начинается с корня и заканчивается в корне.
2. Каждый шаг на одну дугу от корня кодируется единицей.
3. В узле выбираем направление на вершину с меньшим номером.
4. Достигнув листа, идем назад, кодируя каждый шаг нулем.
5. При движении назад в узле всегда выбираем направление на не пройденную вершину с меньшим номером.

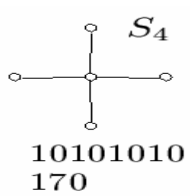
Полученное двоичное число переводим в десятичное. Очевидно, двоичная форма кода должна заканчиваться нулем (возвращение в корень), поэтому десятичный код дерева всегда четный.

Кодировка в такой форме получается достаточно компактной, однако она не несет в себе информации о номерах вершин дерева. Существуют аналогичные кодировки, где вместо единиц в таком же порядке проставляются номера или названия вершин.

Есть деревья, для которых несложно вывести формулу десятичной кодировки. Рассмотрим, например, графы-звезды $K_{1,n}$, являющиеся полными двудольными графами, одна из долей которых состоит из одной вершины. Эта же вершина является центром и центроидом дерева. Наименьший вес вершин звезды равен 1. Другое обозначение звезд — $K_{1,n} = S_n$.







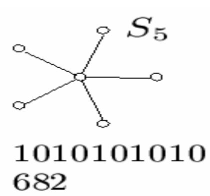


Рис. 1 Рис. 2 Рис. 3 Рис. 4

На рисунках 1-4 приведены звезды и их двоичные и десятичные кодировки. Корень дерева располагается в центральной вершине звезды. Легко получить общую формулу [2]:

$$Z(S_n) = 2(4^n - 1)/3$$

Если корень поместить в любую висячую вершину, то код Z’ такого дерева будет выражаться большим числом. Более того, существует зависимость

$$Z(S_n) - Z'(S_n) = Z(S_{n-1})$$

Аналогично рассматриваются цепи. У цепей C_{2n} и C_{2n+1} наименьший вес вершин равен *n*. Центр и центроид цепей совпадают. (рис. 5).

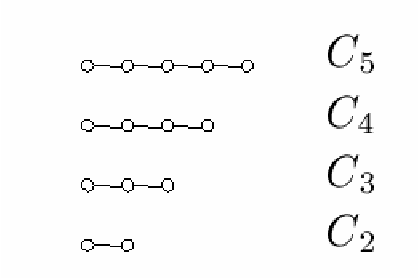


Рис. 5

В звездах только два варианта расположения корня с различными десятичными кодировками. В цепи же число вариантов кодировок в зависимости от положения корня растет с увеличением *n*. Рассмотрим самый простой вариант, расположив корень в концевой вершине (листе). Для C_2 получим десятичную кодировку 10 и двоичную 2. Точно так же для остальных цепей: 1100 и 12, 111000 и 56, 11110000 и 240. Общая формула для десятичной кодировки цепи с корнем в концевой вершине имеет вид

$$Z(C_n) = 2^{n-1}(2^{n-1} - 1)$$

Итак, каждому дереву можно поставить в соответствие десятичное число. Очевидно, обратное соответствие не существует. Выпишем наименьшие числа, которым соответствуют деревья. Для этого воспользуемся диаграммами деревьев Принса и Харари [3]

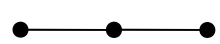


Рис. 6. G_3

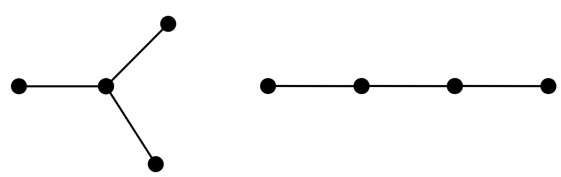


Рис. 7. G_4

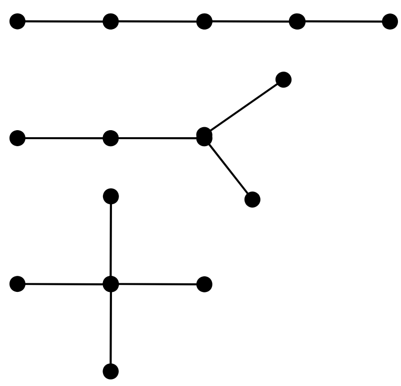


Рис. 8. G_5

Дерево G_2 состоит из ребра и двух вершин и имеет двоичный код 10 или десятичный 2. Для деревьев G_3 (рис. 6) имеем две десятичные кодировки в зависимости от положения корня 10 и 12. В деревьях G_4 кодировка зависит не только от положения корня, но и от нумерации вершин. Первое дерево на рис. 7 (цепь) имеет две различные кодировки 42 и 44, второе – три: 50, 52, 56. Число деревьев G_5 с различными кодировками еще больше. Имеем для цепи G_5 коды 184, 204, 226, 240. Второе на рис. 8 дерево имеет 8 вариантов кодов: 172, 178, 180, 202, 204, 210, 216, 228. Коды для G_5 - звезды S_4 : 170, 212.

Аналогично, для деревьев G_6 имеем 42 кода:

$Z(G_6)=$ 682, 684, 690, 692, 696, 714, 716, 722, 724, 728, 738, 740, 744, 752, 810, 812, 818, 820, 824, 842, 844, 850, 852, 856, 866, 868, 872, 880, 906, 908, 914, 916, 920, 930, 932, 936, 944, 962, 964, 968, 976, 992.

Замечена следующая закономерность

$$Z(G_k) = Z(G_{k-1}) + 2^{2k-3}\text{, k=3,4...}$$

Для кодировки деревьев использована следующая программа [2] системы символьной математики Maple:

```
> restart:with(networks):S11:={}:
> n:=5:           # Порядок графа
> V:={$1..n}:     # Все вершины
> for vk to 5 do # Корень
> for a1 in V do
> for a2 in V minus{a1} do
```

```
> for a3 in V minus{a1,a2} do
> for a4 in V minus{a1,a2,a3} do
> for a5 in V minus{a1,a2,a3,a4} do
> v1:=vk:
> new(G):
> n2:=2*n-2: # Число цифр в коде
> addvertex(V,G):
>  addedge(Path(a1,a2,a3,a4),G):addedge(Path(a3,a5),G):
> # addedge(Path(a1,a2,a3,a4,a5),G):
> # addedge(Path(a1,a2,a3),G): addedge(Path(a5,a2,a4),G):
> draw(G):
> V0:={v1}:# Множество вершин
> i:=0:    # Счетчик элементов кода
> S:={v1}: # Множество пройденных вершин
> while i<n2 do
```

Движение от корня, кодировка единицами

```
> vd1:=0:
> while vd1<>1 do
>   v1:=min(op(neighbors(v1,G) minus S));
>   vd1:=vdegree(v1,G):
>   i:=i+1;
>   KOD[i]:=1;
>   S:=S union {v1};
> od:
```

Возвращение назад, кодировка нулями

```
> V0:={v1}:
> while nops(v1)=1 do
>  v10:=v1:
>  v1:=neighbors(v1,G) minus V0;
>  if nops(v1)=1 and nops(v1 intersect S)<>0 then
>    V0:=V0 union {v1[1]}:
>    i:=i+1: KOD[i]:=0:
>    delete(edges({op(v10),op(v1)},G),G):
>    else v1:=v10: break;
>  fi;
> od;
> v1:=op(v1);# Начальная вершина для движения от корня
> od:# Конец цикла while i<n2
> Z2:=convert(KOD,list);
> Z10:=add(Z2[i]*2^(n2-i),i=1..n2);# Перевод в десятичное
число
> S11:=S11 union {Z10};         # Множество чисел-деревьев
> od:od:od:od:od:od:S11;nops(S11);
```

$$\{172, 178, 180, 202, 210, 216, 228, 232\}$$

В программе применяются функции пакета `networks.` Перебор кодировок производится в *n* вложенных циклах для деревьев G_n. С учетом изменения положения корня для дерева порядка *n* требуется перебор *n*!*n* вариантов. При *n*=6 время счета на компьютере с хорошим двухядерным процессором достигает несколько минут. Программа приведена для *n*=5. Рассчитывается второе дерево на рис. 8. Команды `addedge(Path(..),G)`, формирующие два других варианта, закомментированы знаком #. Текст программы доступен на сайте автора http://vuz.exponenta.ru.

Литература

1. Касьянов В.Н., Евстигнеев В.А. Графы в программировании: обработка, визуализация и применение. — СПб.: БХВ-Петербург, 2003. – 1104 с.
2. Кирсанов М.Н. Графы в Maple.— М.: ФИЗМАТЛИТ,2007-168 с.
3. Харари Ф. Теория графов. — М.:Едиториал УРСС, 2003. -296 с.