Univ. u Novom Sadu
Zb. Rad. Prirod.–Mat. Fak.
Ser. Mat. 24, 1 (1994), 261-267

Review of Research
Faculty of Science
Mathematics Series

AN ALGORITHM FOR GENERATION AND ENUMERATION OF HAMILTONIAN CYCLES IN $P_m \times P_n$

Olga Bodroža
Institute of Mathematics, University of Novi Sad
Trg Dositeja Obradovića 4, 21000 Novi Sad, Yugoslavia

Abstract

An algorithm which generates and enumerates Hamiltonian cycles in $P_m \times P_n$ (rectangular lattice graph) is offered in the paper. It was implemented in PASCAL.

AMS Mathematics Subject Classification (1991): 05C45
Key words and phrases: cartesian product, Hamiltonian graphs.

1. Introduction

We denote by $P_m \times P_n$ a graph whose vertices form a rectangular set of lattice points in the plane and the set of edges consists of pairs of vertices that are adjacent along horizontal or verticial lines [1] (m vertices in each column and n vertices in each row). In this graph, there are $(m-1)\times(n-1)$ squares (cycles of order 4) which we call the *windows* of the graph . We can associate with the graph $P_m \times P_n$ its *window lattice graph* $W_{m,n}$ as it was done in [1]) whose vertices are the *windows* of $P_m \times P_n$, two vertices being adjacent in $W_{m,n}$ if and only if the two *windows* of $P_m \times P_n$ which correspond to those vertices have a common edge. Now , for each Hamiltonian cycle , we associate with the graph $W_{m,n}$ a binary matrix $A = [a_{i,j}]_{m-1,n-1}$ defining its elements in the following way:

$$a_{i,j} = \begin{cases} 1 & \text{if} \quad w_{i,j} \quad \text{belongs to the interior of the Hamiltonian cycle} \\ 0 & \text{otherwise} \end{cases}$$

The graph $P_9 \times P_{12}$ and one of its Hamiltonian cycles with associated matrix A are shown in Fig.1 and Fig.2 below.

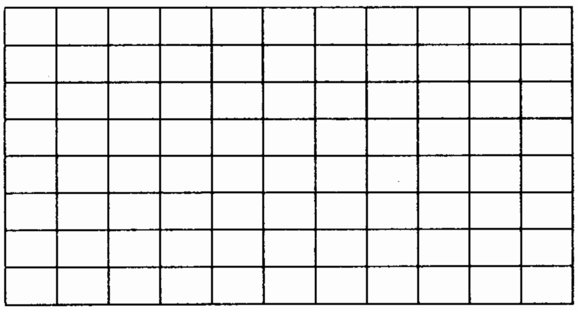

Fig.1

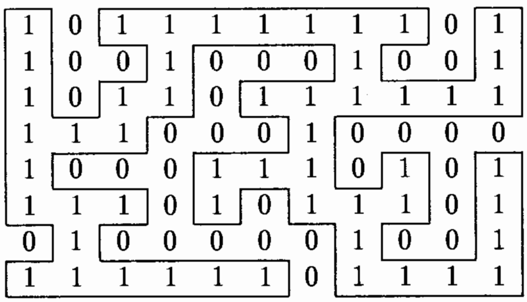

Fig.2

This matrix satisfies the following necessary conditions which are easy to verify (these conditions correspond to conditions (BC) , (IC) , (CC) and (EC) from [3]):

- *Adjacency of Column Conditions:*

  (1) $(\forall j)(1 \leq j \leq n-2)$

  $\neg(a_{1,j} = a_{1,j+1} = 0 \ \vee \ a_{m-1,j} = a_{m-1,j+1} = 0)$

  (2) $(\forall i)(1 \leq i \leq m-2)(\forall j)(1 \leq j \leq n-2)$

  $(a_{i,j}, a_{i+1,j}, a_{i,j+1}, a_{i+1,j+1}) \notin \{(0,0,0,0),(1,1,1,1),(1,0,0,0),(0,1,1,0)\}$

- *First and Last Column Conditions :*

  (3) $a_{1,1} = a_{m-1,1} = a_{1,n-1} = a_{m-1,n-1} = 1$

  (4) $(\forall i)(1 \leq i \leq m-2)$

  $\neg(a_{i,1} = a_{i+1,1} = 0 \ \vee \ a_{i,n-1} = a_{i+1,n-1} = 0)$

- *Connection Condition:*

  The subgraph of $W_{m,n}$ induced by the windows belonging to the interior of the Hamiltonian cycle forms a tree and the subgraph of $W_{m,n}$ induced by the windows belonging to the exterior of the Hamiltonian cycle forms a forest.

  We call the components of the forest *exterior trees (ET)* .

The converse is also satisfied : every binary matrix $A = [a_{i,j}]_{m-1,n-1}$ which satisfies the adjacency of two column conditions , the first and last column conditions and the connection condition determines exactly one Hamiltonian cycle for the $P_m \times P_n$.

In the upper assertion we can replace the connection condition with the following :

- *Root Condition :*
  The subgraph of $W_{m,n}$ induced by the windows belonging to the exterior of the Hamiltonian cycle forms a forest such that each component (ET) contains exactly one window $w_{i,j}$ (we call it the *root of the exterior tree*) that satisfies the following condition :

  $(i \in \{1,(m-1)\} \wedge j \notin \{1,(n-1)\}) \vee (j \in \{1, n-1\} \wedge i \notin \{1,(n-1)\})$
  (5)

Our task is to generate all binary matrices which fulfill (1), (2), (3), (4) and the root condition.

In what order ?

2. Generation of Hamiltonian cycles

Since each 0-window (the one to which a 0-element of A corresponds) belongs to exactly one ET our task is to find all possible dispositions of all possible exterior trees , which satisfy all above conditions. Since each ET has exactly one root (and we know where it is , see (5)), we can now adopt the following priority in construction of the exterior trees. First we build the exterior trees, whose roots are at the first row (if such ET exist at all) and among them those with smaller second indices (if ET with the root $w_{1,i}$ is built before ET with the root $w_{1,j}$, then $i < j$ is satisfied); then we build the ones whose roots are at the last column (if they exist) and among them the first are the ones with smaller first indices ; then the one whose roots are at the last row (if they exist) and among them the first are the ones with larger second indices, and finally the one whose roots are at the first column (if they exist) and among them the first are the ones with larger first indices.

The elements of matrix A , actually the vertices of $W_{m,n}$ are realized in the following way :

```
point=^window  ;
window=record
        value:integer;
        level:integer;
        north,south,west,east,boundary:point ;
end;
```

For each element $a_{i,j}$ of matrix A pointers *north* , *south* , *east* and *west* determine the adjacent elements (for $1 < i < m-1$ and $1 < j < n-1$, they are $a_{i-1,j}, a_{i+1,j}, a_{i,j+1}$ and $a_{i,j-1}$ in this order) or point to *nil* if there is no corresponding element.

Pointer *boundary* is determined in the following way :

$$\text{boundary} \uparrow = \begin{cases} a_{i,j+1} & \text{for} \quad i = 1 & \text{and} \quad j < n-2 \\ a_{i+1,j} & \text{for} \quad j = n-1 & \text{and} \quad i < m-2 \\ a_{i,j-1} & \text{for} \quad i = m-1 & \text{and} \quad j > 2 \\ a_{i-1,j} & \text{for} \quad j = 1 & \text{and} \quad i > 2 \\ nil & \text{for} \quad 1 < i < m-1 & \text{and} \quad 1 < j < n-1 \end{cases}$$

At the beginning all elements of the matrix A are *free* i.e. , the values of all windows are *undetermined* i.e. equal 2 except the elements $a_{1,1}$, $a_{1,n-1}$, $a_{m-1,1}$ and $a_{m-1,n-1}$ whose values are equal to 1 (the corresponding windows belong to the interior of each Hamiltonian cycle).

We are permanently trying to make new exterior trees (join 0 to some free elements) and after that "hem" them with 1-windows (join 1 to some free elements if it hasn't done that yet) which are adjacent elements of the ET or "touch" the tree with one of their corners at least , taking into account that the conditions : (1), (2), (3), (4) and the root condition are satisfied. If we finish this "hemming " successfully we continue by choosing new possible exterior trees and their *1-hems* but if we don't manage to "hem" a tree we have to look for a new possibility for a tree with the fixed root. When we exhaust all the possibilities with the fixed root , we have to "delete" the existing tree (and the root) and all the windows which got a value 1 in the process of 1-heming of the tree (their window's value becomes equal to 2 again) and then choose a new root with the same ordering number (it is the ordering number of the tree whose root it is)

The level of the window $w_{i,j}$ in fixed moment is equal to $max = const$ if the value of the window is undetermined (i.e. equals 2) and if the value of the window equals 0 or 1 then the level of the window $w_{i,j}$ is equal to the ordering number of the tree to which it belongs or to whose 1-hem it belongs, so if the value of $w_{i,j}$ equals 1 then the level of $w_{i,j}$ is equal to the maximum of orders of the tree with which it has at least one common corner. It is necessary that we know whether a window with the value of 1 belongs to the hem of the tree which we delete at the moment or not and whether we should delete the window or not.

We are trying to join elements from the set $\{0, 1\}$ to each window (actually to each element of the matrix A) and find all such binary matrices A using BACK-TRACK (every time we manage to do it we get a new Hamiltonian cycle).

The following procedures and functions were used in the program :

- *function forward(r1,r2:point):point;*
- *function right(r1,r2:point):point;*
- *function left(r1,r2:point):point;*
  We use these functions for walking trough the graph $W_{m,n}$
- *function limited(x:integer;p:point):boolean;*
  It is true if and only if the window $p \uparrow$ touches with its corner three windows with the values x (in this case the value of $p \uparrow$ must not be equal to x because of the condition (2)).
- *function OK(k:integer;t1,t2:point):boolean;*
  It is true if and only if t2 ↑ can be added to the exterior tree with the ordering number k passing the window t1 ↑ which already belongs to the tree.
- *procedure hem(k5:integer);*
  This procedure tries to hem the ET whose ordering number is k5. It results in *success* (boolean, global variable).
- *procedure rid(k6:integer; r:point);*
  If $r \uparrow$ *.level* is equal to k6 then this procedure gives the window r ↑ value *max* to its *level* and 2 for *value* (the window becomes undetermined again). We use the procedure during deleting (ridding) the exterior trees or some of their parts.
- *procedure newtree(k4:integer);*
  It looks for a new allowed ET with the ordering number k4 which can be correctly hemmed till it manages to do it or concludes that (case: last=true) there are not any possibilities for choosing a new ET (or even a new root) with the ordering number k4 and in that case we have to go back to the level $(k4-1)$ and look for a new possibility for ET with the ordering number $(k4-1)$.

How does this procedure look for a possible new ET ?

It is constantly trying to make a new ET by adding a new window to the previous ET using BACK–TRACK with deleting (ridding) some windows in principle : *on right – first* .

Using the program the number of Hamiltonian cycles in the graph $P_m \times P_n$ (we mark it as $H(m,n)$) for some values of m and n is determined for the first time ($H(m,n)$ for $m = 4$ and $n = 5$ were known , see [2] and [3]). The new results are given below. (It is well known that $H(m,n) = 0$ if and only if $m \cdot n$ is oddy).

	m	6	7	8
n				
3		4	0	8
4		37	92	236
5		154	0	1696
6		1072	5320	32675
7		5320	0	301384
8		32675	301384	4638576
9		175294	0	

References

[1] Myers,B.R., Enumeration of tours in Hamiltonian rectangular lattice graphs, Math. Magazine, 54 (1981) , 19-23 .

[2] Tošić,R. , Bodroža, O. , Kwong, Y.H.H., Straight, H.J., On the number of Hamiltonian cycles of $P_4 \times P_n$, Indian J. of Pure and Applied Math., 21(5)(1990), 403-409.

[3] Kwong Y.H.H., Enumeration of Hamiltonian cycles in $P_4 \times P_n$ and $P_5 \times P_n$, Ars Combinatoria, 33(1992), 87-96.

REZIME

ALGORITAM ZA GENERISANJE I PREBROJAVANJE HAMILTONOVIH KONTURA U GRAFU $P_m \times P_n$

U radu je predložen jedan algoritam za generisanje i prebrojavanje Hamiltonovih kontura u grafu $P_m \times P_n$ koji je implementiran u PASCAL-u.

Received by the editors October 3, 1991