TAIWANESE JOURNAL OF MATHEMATICS
Vol. 10, No. 2, pp. 327-338, February 2006
This paper is available online at http://www.math.nthu.edu.tw/tjm/

SOLVING THE HAMILTONIAN CYCLE PROBLEM USING SYMBOLIC DETERMINANTS

V. Ejov, J. A. Filar, S. K. Lucas and J. L. Nelson

Abstract. In this note we show how the Hamiltonian Cycle problem can be reduced to solving a system of polynomial equations related to the adjacency matrix of a graph. This system of equations can be solved using the method of Gröbner bases, but we also show how a symbolic determinant related to the adjacency matrix can be used to directly decide whether a graph has a Hamiltonian cycle.

1. Introduction

The Hamiltonian Cycle Problem (HCP) is a well known NP-complete problem (see for example Cormen *et al.* [1] or Johnson and Papadimitriou [5]). Given a graph $G = (V, E)$, can a cycle be found that visits every vertex $v \in V$ exactly once. Such a cycle is known as a Hamiltonian Cycle (HC), and a graph G with an HC is called Hamiltonian.

While not as well developed as the Travelling Salesman Problem, there are a number of algorithms specifically developed for the HCP, many of which are well summarised in Vandegriend [9]. The algorithms naturally fall into two classes: backtrack and heuristic. Backtrack algorithms systematically go through every possible path from a given starting vertex, eliminating those that reach dead ends and cannot form an HC. They are guaranteed to find an HC if one exists, or show the graph is not Hamiltonian, but are of exponential complexity as the graph size increases. Heuristic algorithms are approximate ones which will usually find an HC (if there is one) in polynomial time, but are not guaranteed to find one that exists. The heuristic algorithms described in Vandegriend [9] are typically some form of genetic algorithm, while in the optimisation community, the HCP is converted to an

Received November 14, 2004; Accepted April 7, 2005.
2000 *Mathematics Subject Classification*: 05C45, 13P10, 68W30, 68R10.
Key words and phrases: Hamiltonian cycle, Gröbner bases, Symbolic algebra.

integer programming problem and then solved by heuristics inspired either by other integer programming heuristics or by the structure of the graph (see for instance, the survey papers in Lawler *et al.* [7]). An alternative form of heuristic algorithm has recently been developed based on embedded Markov chains (see Ejov *et al.* [3] or Feinberg [4]).

The aim of this paper is to introduce a new style of algorithm for the HCP which guarantees finding an HC if it exists, and for non-Hamiltonian graphs at least will find its longest cycle. The method is motivated by applications in symbolic linear algebra, and essentially investigates the symbolic determinant of a graph's adjacency matrix. While this method is not computationally competitive with backtrack algorithms, it has advantages in terms of ease of finding long cycles in non-Hamiltonian graphs, and in terms of novelty for practitioners of the HCP.

2. Graph Properties

2.1. Definitions and the Adjacency Matrix

We define a *graph* $G = (V, E)$ as a set of vertices V and a set of edges E that are associated with pairs of vertices. A *directed* graph is one in which a direction is associated with each edge, and so each edge is described as an ordered pair (i, j) where $i, j \in V$. An *undirected* graph is one where each edge can be traversed in either direction, and is equivalent to a directed graph where $(i, j) \in V \to (j, i) \in V$ for $i, j \in V$. We will also only be considering *simple* graphs, in the sense that $(i, i) \notin E$, or there are no loops.

One simple way to represent a graph is by its *adjacency* matrix. Given a graph with n vertices, define

$$A = (a_{ij}), \quad \text{where} \quad a_{ij} = \begin{cases} 1 & \text{if } (i, j) \in E, \\ 0 & \text{otherwise.} \end{cases}$$

The adjacency matrix of an undirected graph will be symmetric, and a simple graph will have zeros on the main diagonal.

One of the more obvious approaches to a graph is to look at the eigenvalues of its adjacency matrix. Unfortunately, this approach will not determine if a graph is Hamiltonian. For example, consider the two graphs in Figure 1 with adjacency matrices

$$A_1 = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{pmatrix} \quad \text{and} \quad A_2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

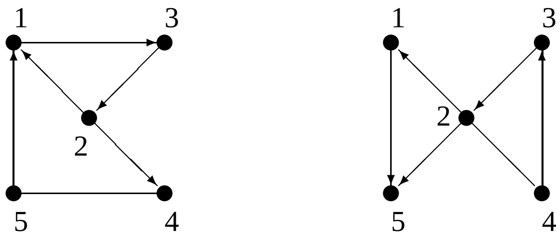


Figure. 1. A pair of graphs with the same characteristic polynomial.

An arrow represents a directed edge, while a line can be traversed in either direction. The first graph is Hamiltonian with an HC $1-3-2-4-5-1$, while the second is not Hamiltonian, but $\det(\lambda I - A_1) = \det(\lambda I - A_2) = \lambda^5 - \lambda^3 - \lambda^2$.

2.2. A Property of Hamiltonian Cycles

We begin with a theorem of Fröbenius, which is described in a more general form in Varga [10].

Theorem 1. *Given a connected graph, it is called* cyclic of index k *if its adjacency matrix has* $k > 1$ *eigenvalues* λ_j *that satisfy* $|\lambda_j| = \max_i |\lambda_i|$ *for* $j = 0, 1, \ldots, k-1$. *Additionally, these* k *eigenvalues are the roots of*

$$\lambda^k - \left(\max_i |\lambda_i|\right)^k = 0.$$

While we could use this theorem here, the proof presented in Varga [10] is overly complex. What follows is a specialisation of Theorem 1 tailored to our particular problem. First, we note that a Hamiltonian cycle is a subgraph of a given graph with all the original n vertices and only n selected edges. Since a Hamiltonian cycle will enter and leave each vertex exactly once, each row and column of the adjacency matrix for a Hamiltonian cycle will contain exactly one entry of '1', and all the rest zeros. This is in fact the definition of a permutation matrix, and thus:

Theorem 2. *An* $n \times n$ *permutation matrix is the adjacency matrix of some Hamiltonian cyclic graph on* n *vertices if and only if its characteristic polynomial is* $\lambda^n - 1 = 0$.

In what follows, we only need to consider the $n \times n$ permutation matrix

$$C_n = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & \cdots & 0 & 0 \end{pmatrix},$$

since any other permutation matrix A_n corresponding to a Hamiltonian cycle can be transformed to C_n by a combination of row and column swaps. This means that A_n can be written in terms of C_n as $A_n = PC_nP^{-1}$ for some other permutation matrix P, so A_n and C_n are similar, and have the same characteristic polynomial.

To find the characteristic polynomial of C_n, we first prove a lemma concerning submatrices of $\lambda I - C_n$.

Lemma 3. *All $n \times n$ matrices of the form*

$$S_n^{(1)} = \begin{pmatrix} 0 & -1 & 0 & \cdots & 0 \\ 0 & \lambda & -1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda & -1 \\ -1 & 0 & \cdots & 0 & \lambda \end{pmatrix},$$

have determinant equal to -1, and all $n \times n$ matrices of the form

$$S_n^{(2)} = \begin{pmatrix} \lambda & -1 & 0 & \cdots & 0 \\ 0 & \lambda & -1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda & -1 \\ 0 & 0 & \cdots & 0 & \lambda \end{pmatrix},$$

have determinant equal to λ^n.

Proof. We use an inductive argument to prove the first part of this lemma. For the base case,

$$\det(S_2^{(1)}) = \begin{vmatrix} 0 & -1 \\ -1 & \lambda \end{vmatrix} = -1.$$

If we assume $\det(S_n^{(1)}) = -1$, we want to prove $\det(S_{n+1}^{(1)}) = -1$. We can expand along the first row of the determinant of $S_{n+1}^{(1)}$, whose only nonzero element is $s_{1,2}^{(1)} = -1$. We thus have $\det(S_{n+1}^{(1)}) = (-1)^{1+2}(-1)\det(S_n^{(1)}) = -1$ using the assumption. Thus the first part of the lemma follows by induction for all positive integers n. The second part is immediately obvious. ■

Given this lemma, we can now prove Theorem 2. Let us begin by proving the case $n = 2$:

$$\det(\lambda I - C_2) = \begin{vmatrix} \lambda & -1 \\ -1 & \lambda \end{vmatrix} = \lambda^2 - 1.$$

For $n > 3$, let us expand the determinant of $\lambda I - C_n$ across the first row:

$$\begin{vmatrix} \lambda & -1 & 0 & \cdots & 0 \\ 0 & \lambda & -1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda & -1 \\ -1 & 0 & \cdots & 0 & \lambda \end{vmatrix} \begin{array}{l} = (-1)^{1+1}\lambda S^{(2)}_{n-1} + (-1)^{1+2}(-1)S^{(1)}_{n-1} \\ = \lambda\lambda^{n-1} + (-1) = \lambda^n - 1, \end{array}$$

which completes the proof. ■

3. Reduction to a System of Polynomial Equations

We are now in a position to use Theorem 2 to establish whether a given graph is Hamiltonian. To do so, we first need to define the *modified* adjacency matrix of a graph. Instead of placing a one at row i column j when there a (directed) edge from vertex i to vertex j, we place the variable x_{ij} in that position. For the graphs in Figure 1, we obtain modified adjacency matrices of the form

$$X_1 = \begin{pmatrix} 0 & 0 & x_{1,3} & 0 & 0 \\ x_{2,1} & 0 & 0 & x_{2,4} & 0 \\ 0 & x_{3,2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x_{4,5} \\ x_{5,1} & 0 & 0 & x_{5,4} & 0 \end{pmatrix},$$

and

$$X_2 = \begin{pmatrix} 0 & 0 & 0 & 0 & x_{1,5} \\ x_{2,1} & 0 & 0 & x_{2,4} & x_{2,5} \\ 0 & x_{3,2} & 0 & 0 & 0 \\ 0 & x_{4,2} & x_{4,3} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Our aim is to choose values of the x_{ij}'s to form a Hamiltonian cycle within the graph, if at all possible.

3.1. Characterisation as a System of Polynomial Equations

The variables x_{ij} in the modified adjacency matrix X have certain restrictions when forming a (hopefully Hamiltonian) subgraph. Since the entries in an adjacency matrix can only take the values zero or one, a natural constraint is $x_{ij}(x_{ij} - 1) = 0$ for all x_{ij} variables in the modified adjacency matrix. In addition, a Hamiltonian cycle must be a permutation matrix, with exactly one 1 in every row and column. This ensures only one arc joins a vertex, and only one leaves. This can be written as $\sum_i x_{ij} - 1 = 0$ for all j, and $\sum_j x_{ij} - 1 = 0$ for all i.

Finally, we require the result of Theorem 2, that the adjacency matrix has characteristic polynomial $\lambda^n - 1$. This can be written as $\det(\lambda I - X) - \lambda^n + 1 = 0$, where we should read this equation as equating each of the $n+1$ coefficients of the polynomial to zero.

In summary, given the modified adjacency matrix of a graph X, a Hamiltonian cycle is equivalent to the solution of the system of polynomial equations

$$\text{(1)} \qquad \begin{cases} x_{ij}(1-x_{ij}) \;=\; 0 \text{ for all } (i,j) \in E, \\ \sum_j x_{ij} - 1 \;=\; 0, \text{ for all } i \\ \sum_i x_{ij} - 1 \;=\; 0 \text{ for all } j, \\ \det(\lambda I - X) - \lambda^n + 1 \;=\; 0. \end{cases}$$

If this system of polynomial equations has no solution, then the graph does not have a Hamiltonian cycle.

3.2. Solution Using Gröbner Bases

A powerful technique for solving systems of polynomial equations uses the technique of Gröbner bases, as applied using Buchberger's algorithm. The technique transforms a system of polynomial equations into a "canonical" form which is much easier to solve. We do not intend to provide a tutorial on Gröbner bases, but refer the reader to the literature (for example Cox *et al.* [2] or Kreuzer and Robbiano [6]) for more details. One important detail is related to Hilbert's Nullstellensatz, which states that a system of polynomial equations has no solution if its Gröbner basis is $\{1\}$. Thus the Gröbner basis method provides an easy check on whether a graph is Hamiltonian or not. Finally, we should note that in general the computational complexity of solving a system of polynomial equations via Gröbner bases is at least exponential [2, 6].

3.3. Examples

Let us begin by considering the two examples from Figure 1, using the symbolic manipulation package Maple. For the first case, after finding the determinant of $\lambda I - X_1$ and equating coefficients, we have the input

```
with(grobner):ff:=[x13*(1-x13),x21*(1-x21),x24*(1-x24),
 x32*(1-x32),x45*(1-x45),x51*(1-x51),x54*(1-x54),x21+x51-1,
 x32-1,x13-1,x24+x54-1,x45-1,x13-1,x21+x24-1,x32-1,x45-1,
 x51+x54-1,x45*x54,x21*x32*x13,x21*x32*x13*x45*x54-x51*x32*
  x13*x24*x45+1];
gbasis(ff,[x13,x21,x24,x32,x45,x51,x54])
```

which returns

```
gbasis= [x13 - 1, x21, x24 - 1, x32 - 1, x45 - 1, x51 - 1, x54]
```

and implies that $x_{13} = 1$, $x_{21} = 0$, $x_{24} = 1$, $x_{32} = 1$, $x_{45} = 1$, $x_{51} = 1$ and $x_{54} = 0$, so is the Hamiltonian cycle 1-3-2-4-5-1.

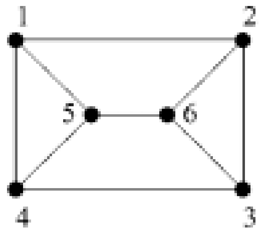
In the second case, with input

```
with(grobner):ff:=[x15*(1-x15),x21*(1-x21),x24*(1-x24),
 x25*(1-x25),x32*(1-x32),x42*(1-x42),x43*(1-x43),x21-1,
 x32+x42-1,x43-1,x24-1,x15+x25-1,x15-1,x21+x24+x25-1,x32-1,
 x42+x43-1,x24*x42,x32*x43*x24,0];
gbasis(ff,[x14,x21,x24,x25,x32,x52,x53])
```

which perhaps unsurprisingly returns the output `[1]` indicating no Hamiltonian cycle, since the characteristic polynomial for X_2 has no constant term, immediately leading to an equation $0 = 1$.

The last example we wish to consider here is for the (undirected) cubic graph with six nodes whose graphical and modified adjacency matrix representations are in Figure 2. The associated system of equations as input to Maple is quite large. One part is the characteristic polynomial which is equated to $\lambda^n - 1$. It is:

$$
\begin{aligned}
&\lambda^6-(x_{23}x_{32}+x_{36}x_{63}+x_{14}x_{41}+x_{12}x_{21}+x_{15}x_{51}+x_{56}x_{65}+x_{54}x_{45}\\
&\quad+x_{26}x_{62}+x_{34}x_{43})\lambda^4-(x_{26}x_{63}x_{32}+x_{14}x_{51}x_{45}+x_{36}x_{23}x_{62}+x_{54}x_{41}x_{15})\\
&\lambda^3+(-x_{14}x_{43}x_{32}x_{21}+x_{54}x_{23}x_{32}x_{45}+x_{56}x_{23}x_{32}x_{65}+x_{12}x_{56}x_{21}x_{65}\\
&\quad+x_{54}x_{62}x_{45}x_{26}+x_{54}x_{12}x_{21}x_{45}+x_{34}x_{43}x_{56}x_{65}-x_{12}x_{51}x_{26}x_{65}\\
&\quad+x_{12}x_{21}x_{36}x_{63}+x_{14}x_{56}x_{41}x_{65}-x_{56}x_{21}x_{15}x_{62}+x_{34}x_{43}x_{26}x_{62}\\
&\quad+x_{14}x_{41}x_{36}x_{63}-x_{34}x_{63}x_{45}x_{56}+x_{51}x_{15}x_{36}x_{63}+x_{51}x_{26}x_{15}x_{62}\\
&\quad+x_{54}x_{45}x_{36}x_{63}+x_{34}x_{12}x_{43}x_{21}-x_{54}x_{36}x_{43}x_{65}-x_{34}x_{12}x_{23}x_{41}\\
&\quad+x_{14}x_{62}x_{41}x_{26}+x_{15}x_{51}x_{23}x_{32}+x_{34}x_{51}x_{43}x_{15}+x_{14}x_{23}x_{32}x_{41})\\
&\lambda^2+(-x_{54}x_{12}x_{41}x_{26}x_{65}+x_{54}x_{62}x_{45}x_{36}x_{23}-x_{34}x_{12}x_{63}x_{41}x_{26}\\
&\quad+x_{14}x_{63}x_{32}x_{41}x_{26}+x_{54}x_{41}x_{15}x_{23}x_{32}-x_{54}x_{43}x_{32}x_{26}x_{65}\\
&\quad+x_{14}x_{51}x_{62}x_{45}x_{26}-x_{14}x_{51}x_{36}x_{43}x_{65}+x_{51}x_{36}x_{23}x_{15}x_{62}\\
&\quad+x_{14}x_{62}x_{41}x_{36}x_{23}-x_{56}x_{21}x_{15}x_{63}x_{32}-x_{14}x_{62}x_{45}x_{56}x_{21}\\
&\quad+x_{54}x_{41}x_{26}x_{15}x_{62}+x_{54}x_{41}x_{15}x_{36}x_{63}+x_{14}x_{51}x_{45}x_{36}x_{63}\\
&\quad-x_{54}x_{21}x_{15}x_{43}x_{32}-x_{34}x_{12}x_{51}x_{23}x_{45}-x_{12}x_{51}x_{36}x_{23}x_{65}\\
&\quad+x_{14}x_{51}x_{23}x_{32}x_{45}+x_{51}x_{26}x_{15}x_{63}x_{32}-x_{14}x_{62}x_{36}x_{43}x_{21}\\
&\quad-x_{34}x_{63}x_{56}x_{41}x_{15}+x_{54}x_{63}x_{32}x_{45}x_{26}-x_{34}x_{23}x_{45}x_{56}x_{62})\\
&\lambda+(+x_{54}x_{41}x_{26}x_{15}x_{63}x_{32}+x_{54}x_{41}x_{36}x_{23}x_{15}x_{62}-x_{54}x_{12}x_{41}x_{36}x_{23}x_{65}\\
&\quad-x_{14}x_{51}x_{43}x_{32}x_{26}x_{65}+x_{14}x_{43}x_{32}x_{56}x_{21}x_{65}+x_{34}x_{43}x_{56}x_{21}x_{15}x_{62}\\
&\quad-x_{34}x_{12}x_{43}x_{56}x_{21}x_{65}-x_{34}x_{51}x_{43}x_{26}x_{15}x_{62}+x_{34}x_{12}x_{51}x_{43}x_{26}x_{65}\\
&\quad+x_{54}x_{12}x_{21}x_{36}x_{43}x_{65}+x_{14}x_{51}x_{62}x_{45}x_{36}x_{23}+x_{34}x_{12}x_{63}x_{45}x_{56}x_{21}\\
&\quad-x_{34}x_{12}x_{51}x_{63}x_{45}x_{26}-x_{14}x_{63}x_{32}x_{45}x_{56}x_{21}+x_{14}x_{51}x_{63}x_{32}x_{45}x_{26}\\
&\quad-x_{54}x_{12}x_{21}x_{45}x_{36}x_{63}-x_{14}x_{23}x_{32}x_{56}x_{41}x_{65}-x_{34}x_{23}x_{56}x_{41}x_{15}x_{62}\\
&\quad+x_{34}x_{12}x_{23}x_{56}x_{41}x_{65}-x_{54}x_{21}x_{36}x_{43}x_{15}x_{62}).
\end{aligned}
\tag{2}
$$



$$X_3 = \begin{pmatrix} 0 & x_{1,2} & 0 & x_{1,4} & x_{1,5} & 0 \\ x_{21} & 0 & x_{2,3} & 0 & 0 & x_{2,6} \\ 0 & x_{3,2} & 0 & x_{3,4} & 0 & x_{3,6} \\ x_{4,1} & 0 & x_{4,3} & 0 & x_{4,5} & 0 \\ x_{5,1} & 0 & 0 & x_{5,4} & 0 & x_{5,6} \\ 0 & x_{6,2} & x_{6,3} & 0 & x_{6,5} & 0 \end{pmatrix}$$

Figure 2. A cubic graph with its modified adjacency matrix.

The solution returned from the Maple Gröbner basis implementation is

```
[x43ˆ2 - x43, 2 x54 x43 - x43 - x54 + x65, x54ˆ2 - x54,
  x56 x43, x54 x56, x56ˆ2 - x56, x43 x63, x54 x63,
  2 x56 x63 - x43 - x54 - 2 x56 - 2 x63 - x65 + 2, x63ˆ2 -
  x63, 2 x43 x65 - x43 + x54 - x65, 2 x65 x54 + x43 - x54 -
  x65, x65 x56, x63 x65, x65ˆ2 - x65, x32 - x43 - x56 - x63 -
  x65 + 1, x23 + x43 + x63 - 1, x14 - x43 - x56 - x63 - x65 +
  1, x12 + x43 + x56 - 1, x15 + x63 + x65 - 1, x21 - x43 -
  x54 - x56 - x63 + 1, x26 + x54 + x56 - 1, x43 + x34 + x54 +
  x56 + x63 + x65 - 2, x35 - x54, x41 + x43 + x63 - 1, x45 -
  x63, x51 + x54 + x56 - 1, x62 + x63 + x65 - 1],
```

which can be easily solved using Maple's `solve` command to give the six sets of solutions

$$\begin{cases} x_{14} = x_{21} = x_{32} = x_{45} = x_{56} = x_{63} = 1, \\ x_{15} = x_{23} = x_{34} = x_{41} = x_{56} = x_{62} = 1, \\ x_{12} = x_{26} = x_{34} = x_{45} = x_{51} = x_{63} = 1, \\ x_{14} = x_{26} = x_{32} = x_{43} = x_{51} = x_{65} = 1, \\ x_{12} = x_{23} = x_{36} = x_{41} = x_{54} = x_{65} = 1, \\ x_{15} = x_{21} = x_{36} = x_{43} = x_{54} = x_{62} = 1, \end{cases}$$

where the other variables in each case take the value zero. These solutions naturally lead to the six distinct Hamiltonian cycles 1-4-5-6-3-2-1, 1-5-6-2-3-4-1, 1-2-6-3-4-5-1, 1-4-3-2-6-5-1, 1-2-3-6-5-4-1, and 1-5-4-3-6-2-1, although the last three are reversals of the first three.

4. Solution Using the Symbolic Determinant

While theoretically elegant, the procedure as described above is not likely to be competitive with backtrack algorithms. It is well known that forming a Gröbner basis is of at least exponential time complexity, but more importantly, the number of terms in the initial system of equations grows at least exponentially, and the process of forming the symbolic characteristic polynomial before inputting to Buchberger's algorithm is itself of at least exponential complexity. Combined with the additional computational effort in performing symbolic algebraic manipulation, this algorithm will be relatively slow. However, the results shown previously hint at a more efficient method for establishing the existence of Hamiltonian cycles, or the longest subcycle in non-Hamiltonian graphs.

Consider, for example, the constant term of (2), which is the symbolic expansion of $\det(X_3)$. It is made up of 21 product terms, where each term appears to represent a set of subcycles on all the vertices of the graph in a particular order. For example, the first product term is $x_{54}x_{41}x_{26}x_{15}x_{63}x_{32}$. If all these coefficients equal one, we can follow the path defined by the order of the subscripts, and see that it represents the pair of subcycles of length three 5-4-1-5 and 2-6-3-2. By checking each product term, we obtain the same six distinct Hamiltonian cycles observed previously, as well as a host of different sets of subcycles. In fact, this is not simply a coincidence, and we can state.

Theorem 3. *The product terms making up the symbolic determinant of the modified adjacency matrix represent sets of subcycles on all vertices of a given graph. In addition, a Hamiltonian cycle will be obtained when the path from any vertex visits every other vertex before returning.*

Proof. As described in most undergraduate linear algebra texts, an *elementary product* from a matrix A is a product of n entries from the matrix, exactly one from each row and column. It can thus be written as $a_{1j_1}a_{2j_2}\cdots, a_{nj_n}$ where $(j_1, j_2, \ldots, j_n)$ is a permutation of $(1, 2, \ldots, n)$. A *signed* elementary product is an elementary product multiplied by ± 1 where the sign depends on whether the permutation $(j_1, j_2, \ldots, j_n)$ is even or odd. Finally $\det(A)$ is the sum of all the signed elementary products from A.

In the context of the modified adjacency matrix of a graph, we will only require elementary products of non-zero elements. Since x_{ij} represents an edge from vertex i to vertex j, each elementary product will have exactly one (directed) edge out from each vertex. By following these edges from any given vertex, a cycle will be formed, which may or may not include all the vertices. If it doesn't, then additional subcycles can be formed by starting from a vertex not yet visited, but if it does include all vertices in the cycle, then we have a Hamiltonian cycle, which completes the proof. ∎

A solution to the Hamiltonian cycle problem is thus simply stated as follows: find the symbolic determinant of the modified adjacency matrix of the graph, and identify the elementary products. A simple linear scan of each elementary product will determine whether it represents a Hamiltonian cycle or not. In fact, about one half of the elementary products will not need to be considered, since in fact we require $\det(X) = -1$, so only negative elementary products should be scanned.

This new algorithm has some interesting similarities to a model for the HCP described in Plotnikov [8], where a logical model for the existence of HCs in a graph is derived, related to the *satisfiability* problem. While we have associated symbolic variables to each edge of the graph, Plotnikov associates a Boolean variable with each edge, and constructs a Boolean expression which is true whenever those edges that form an HC have true Boolean variables associated with them, and the other edges are false. The advantage of Plotnikov's method is that Hamiltonian cycles can be immediately obtained from the Boolean expression, whereas we have to scan through all our product terms. Unfortunately, a key step in Plotnikov's construction requires identifying *all* the cycles in the graph that do not contain all the vertices, and no straightforward way of doing so is described. By using the symbolic determinant instead of logical variables, we are able to form essentially the same result as Plotnikov, but with a well posed algorithm.

Despite its novelty, this symbolic determinant algorithm is, as one would expect, of at least exponential complexity. In the case of a dense matrix, the number of elementary products is of order $n!$, and even in the case of cubic graphs, which in some sense are the simplest graphs for establishing Hamiltonicity, the number of elementary products grows exponentially. However, we may "prune" the determinant, cutting down on the number of product terms. If we find the symbolic determinant step by step by expanding along rows, we notice that the terms in smaller determinants are product terms of increasing length. If a term can be identified that already represents a subcycle, it could be set to zero rather than kept in the determinant until it eventually appears in many final product terms that include that subcycle. In fact, this pruned determinant can be identified with more classical backtrack algorithms, where the product terms represent the path currently chosen, and the determinant holds information on the connectivity of the rest of the graph.

5. Conclusion

We have shown how the Hamiltonian Cycle problem is equivalent to both solving a system of polynomial equations (typically using a Gröbner basis method) and more efficiently, finding patterns within a symbolic determinant. In both cases, the modified adjacency matrix representation of a graph is a crucial intermediate step. While novel in the sense of solving the Hamiltonian Cycle problem using symbolic algebra, unfortunately the complexity of these new methods remains exponential, and the additional overheads of symbolic calculations make them uncompetitive compared with more traditional algorithms.

ACKNOWLEDGMENT

Partial funding for this project was supplied by ARC Grant DP0343028.

REFERENCES

1. T. H. Cormen, C. E. Leiserson and R. L. Rivest, *Introduction to algorithms*, MIT Press, Cambridge MA, 1996.
2. D. Cox, J. Little and D. O'Shea, *Ideals, varieties, and algorithms: an introduction to computational algebraic geometry and commutative algebra, 2nd edition*, Springer, New York, 1997.
3. V. Ejov, J.A. Filar and J. Thredgold, Geometric interpretation of Hamiltonian cycles problem via singularly perturbed Markov decision processes, *Optimization*, **52** (2003), 441-468.
4. E. Feinberg, Constrained discounted Markov decision process with Hamiltonian cycles, *Mathematics of Operations Research*, **25** (2000), 130-140.
5. D. S. Johnson and C. H. Papadimitriou, *The travelling salesman problem: a guided tour of combinatorial optimization*, ch. Computational Compexity, 37-85, John Wiley & Sons Ltd., New York, 1985.
6. M. Kreuzer and L. Robbiano, *Computational commutative algebra 1*, Springer, New York, 2000.
7. E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan and D. B. Shmoys (eds.), *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*, Wiley Series in Discrete Mathematics & Optimization, John Wiley and Sons, New York, 1985.
8. A. D. Plotnikov, A logical model of HCP, *Int. J. Math. Math. Sci.*, **26** (2001), 679-684.
9. B. Vandegriend, Finding Hamiltonian cycles: algorithms, graphs and performance, *Master of Science, Department of Computing Science, University of Alberta* 1998.
10. R. S. Varga, *Matrix iterative analysis*, Prentice-Hall Inc., Englewood Cliffs, NJ, 1962.

V. Ejov, J. A. Filar, S. K. Lucas
School of Mathematics and Statistics,
University of South Australia,
Mawson Lakes SA 5062,
Australia
E-mail: vladimir.ejov@unisa.edu.au
E-mail: jerzy.filar@unisa.edu.au
E-mail: stephen.lucas@unisa.edu.au

J. L. Nelson
Department of Mathematics,
Harvey Mudd College,
1250 N. Dartmouth Ave.
Claremont, CA 91711,
U.S.A.
E-mail: jnelson@math.hmc.edu