NEW TRENDS IN MATHEMATICAL SCIENCES
Vol. 1, No. 1, 2013, p.74-84
ISSN 2147-5520 - www.ntmsci.com

Path Partition in Directed Graph-Modeling and Optimization

Prof. Dr. Issam Abdel Kader[1]

[1]*Faculty of Arts and Sciences,Islamic University of Lebanon*

Abstract: The concept of graph theory is therefore perfectly suitable to structure a problem in its initial analysis phases since a graph is the most general mathematical object. At the structural level, the nodes represent the objects, the variables… and the arc forms the binary relation of influence among them. Many real problems can be modeled as path partition in directed graph that played particular role in the operation of arranging a set of nodes especially in case of directed acyclic graph (DAG). We encounter such graph in schedule problems, the analysis of language structure, the probability theory, the game theory, compilers…. Moreover managerial problem can be modeled as acyclic graphs, also the potential problem has a suitable solution if and only if the graph G is acyclic.

The arc – disjoint paths in a graph has an important application in several areas and needs exact algorithms to find it. In this paper we analyze the bounds of path number in directed graph and we give certain properties characterizing directed acyclic graph that permit to give a structural representation of such graph. The algorithm used determines the topological ordering in time $O(n + |U|)$. We introduce two efficient algorithms that allow the construction of a minimal path-partition, one for the directed acyclic graph with time complexity $O(n^2 + n|U|)$ and the second for the strangely connected tournament having unique Hamiltonian circuit and having time complexity $O(n^2)$.

Keywords: Acyclic graph, Path Partition, tournament, Hamiltonian circuit, Adjacency list, Adjacency matrix, canonical ordering, Spanning tree.

1. Introduction

In management and economic, combinational problems necessitate a complicated formulation since their solution are not easily figured out, need complicated method and are sometime very difficult to set. The graph theory constitutes for instance, without any doubts, one of the most important and most efficient theories to model such kind of problem.

In fact we can use graph as tools to structure relationships among objects, variables etc… where the information can be represented in compact form. The concept of graph theory is therefore perfectly suitable to structure a problem in its initial analysis phase since a graph is the most general mathematical object. At the structural level (relational level) the nodes represent the objects, the variables etc…. and the arcs form the binary relation of influence among them.

Many real problems can be modeled as path-partition in directed graph that played particular role in the operation of arranging a set of points especially in case of directed acyclic graph (DAG). There exists several areas in which DAG arise as models e.g. project management, assignment problem network etc….we encounter such graphs in schedule problems, the analysis of language structure (Computation theory) the probability theory, the game theory etc… moreover managerial problem can be modeled as acyclic graphs. In the other hand the potential problem has a solution of certain type if and only if the graph G is acyclic see [17].

1.1. Concept of Graph

[1] Tel.:+00961 3 288906.
E-mail address: issamabdelkader00@gmail.com

The terminologies and notations are those of ([6], [8], [9], [12]). A directed graph is a pair $G = (X, U)$ where X is a finite set and U is a binary relation on X. The set X is called the vertex set and its elements are called vertices. The set U is called the arcs set and its element is called directed edges or arcs. A path is a sequence of vertices $(x_1, x_2, \dots, x_t)$ such that: $(x_i, x_{i+1}) \in U$ for $i = 1, 2, \dots., t-1$. The length of path is the number of arcs in the path. A path $C = (x_0, x_1, \dots, x_n)$ forms a circuit if $x_0 = x_n$ and the path contains at least an arc. A directed graph with no circuit is called acyclic (DAG).

Let $G = (X, U)$ be a directed graph, for $x \in X$, we denote by $G - x$ the sub- graph obtained from G by deleting the vertex x and the adjacency arcs to it. The out-degree of vertex x denoted $d_G^+(x)$ is the number of arcs leaving it and in-degree of vertex x denoted $d_G^-(x)$, is the number of arcs entering it.

From now on we denote:

$$I_G^+(x) = \{y \in X: (x, y) \in U\}$$
$$I_G^-(x) = \{y \in X: (y, x) \in U\}$$

$$X_G^+ = \{x \in X: d_G^+(x) > d_G^-(x)\}$$
$$X_G^0 = \{x \in X: d_G^+(x) = d_G^-(x)\}$$

A path partition of directed graph $G = (X, U)$ is a set T of arc-disjoint paths such that every arc in U is include in exactly one path of T. path my start and end anywhere, and they may be of any length including 0.

A minimum path partition of G is a path partition of G that use a fewest possible number of paths. The path number of directed graph G, denoted $P(G)$.

Definition 1.2

An asymmetric graph is a directed graph such that (x, y) is an arc implies (y, x) is not an arc. A tournament of order n denoted $T_n = (X, U)$ is a complete asymmetric graph on n vertices see [5, 7, 21, 24]

Definition 1.3

Let $G = (X, U)$ be a directed graph of order n. If $P(G) = e(G) = \sum_{x \in X_G^+} \left(d_G^+(x) - d_G^-(x)\right)$. We say G has the property Q.

2. Results on the path number in directed graph.

The arc–disjoint paths in a graph has an important application in several areas and needs exact algorithms to find it. Alspach and Pullman [4] have conjectured that for any simple graph G of order n, $P(G) \leq [n^2/4]$, O Brian [22], proved this conjecture. From O.Ore [23], we have:

$$P(G) \geq e(G) = \sum_{x \in X_G^+} \left(d_G^+(x) - d_G^-(x)\right)$$

Thus for a directed graph G, we deduce that:

$$e(G) \leq P(G) \leq \left[\frac{n^2}{4}\right]$$

For a further detailed study of P(G), we refer also to ([1], [11]).

2.1. Path-Partition in Tournaments

Theorem 1.

Let $T_n = (X, U)$ be a tournament of order n then $P[T_n] \geq \left[\frac{(n+1)}{2}\right]$.

The number of arc in tournament T_n, is given by the following result:

$$\sum_{x \in X} \left(d_G^+(x) - d_G^-(x)\right) = 2|U| = n(n-1)$$

Then $|U| = \frac{n(n-1)}{2}$

The maximum number of arcs in any path partition is $(n - 1)$. Thus the minimum number of paths needed to cover every arc in T_n is $\frac{n}{2}$, since $P[T_n]$ is an integer, we must have:

$$P[T_n] \geq \left\lceil \frac{(n+1)}{2} \right\rceil$$

From the preceding result and for any tournament T_n, we deduce then:

$$\left\lceil \frac{(n+1)}{2} \right\rceil \leq P[T_n] \leq \left\lceil \frac{n^2}{4} \right\rceil$$

Thereafter, we study the tournament T_n having a unique Hamiltonian circuit. A characterization of T_n have been given by Douglas [13].

Theorem 2.

For $n \geq 5$, a tournament $T_n = (X, U)$ admits a unique Hamiltonian circuit C if and only if the following conditions hold:

(i) There exist a partition of vertices in $X = A \cup B \cup \{x, y\}$

Where $A = \{a_1, \ldots, a_p\}, B = \{b_1, \ldots, b_q\}$ and $A = \{x, a_1, \ldots, a_p, y, b_1, \ldots, b_q, x\}$

(ii) $d_{T_n}^+(x) = 1 = d_{T_n}^-(y)$

(iii) $(b_q, a_1) \in U$ and $(a_p, b_1) \in U$

(iv) For $j > i + 1, (a_j, a_i) \in U$ and $(b_i, b_j) \in U$

(v) If $i < j$ and $u \leq v$ and if $(a_i, b_u) \in U$ then: $(a_j, b_v) \in U$

If $B = \emptyset$ we have a tournament having a unique Hamiltonian circuit and has exactly $m - n + 1$ elementary circuits. This tournament will be denoted $A_n = (X, U)$. The path $\lambda = (x, a_1, \ldots, a_p, y)$ is a spanning tree and the vertices $(x, a_1, \ldots, a_p, y)$ constitute the canonical ordering of A_n, and in the following this sequence will be denoted $(x_1, x_2, \ldots, x_n)$.

The tournament A_n is characterized by:

$$(x_i, x_j) \in U \text{ if and only if } j = i + 1 \text{ or } i > j + 1$$

A curious fact concerning the number $W(n)$ of tournament T_n having a unique Hamiltonian circuit is equal to $(2n - 6)^{\text{th}}$ Fibonacci number. Garray [15] shows that for $n \geq 6$ we have:

$$W(n) = 3W(n - 1) - W(n - 2).$$

Gutin G. [16] provides a characterization allowing to find the number of non isomorphic tournament for $n \geq 5$

Paths and circuits are fundamental sub-structure in tournaments see : [5], [10], [19], [25], [26].

There is a $O(n)$ algorithm for finding Hamiltonian circuit in a tournaments ([20]).

Theorem 3. *(I. Abdel Kader [1]).*

Let $T_n = (X, U)$ be a tournament having a unique Hamiltonian circuit, then $P(T_n) \le [n^2/4] - 2$.

For a further detailed study of tournament having unique Hamiltonian circuit and their number, we refer to ([15], [16]).

Property 1 $P(A_n) = \left[\frac{n^2}{4}\right] - 2$

It is clear that $X_{A_n}^+ = \{x_i : i = [n/2] + 1, \ldots\ldots, n\}$

$$e(A_n) = \sum_{x \in X_{A_n}^+} \left(d_{A_n}^+(x) - d_{A_n}^-(x)\right)$$

but $d_{A_n}^+(x_i) - d_{A_n}^-(x_i) = 2(i - 1) - (n - 1)$ $(i = [n/2] + 1, \ldots, n - 1)$

$$d_{A_n}^+(x_n) - d_{A_n}^-(x_n) = n - 3$$

Then $e(A_n) = \sum_{i=1}^{n-1} \big(2(i - 1) - (n - 1)\big) + n - 3 = \left[\frac{n^2}{4}\right] - 2$

As $P(A_n) \ge e(A_n)$ and $P(A_n) \le [n^2/4] - 2$

We have then the equality.

From the above result, the upper bound of $P(T_n)$ is the best possible.

It is important to note that there exist tournaments having a unique Hamiltonian circuit which are not isomorphic to A_n and which satisfy the equality $P(T_n) = e(T_n) = \left[\frac{n^2}{4}\right] - 2$. An example of this is the tournament T_n for which $|B| = 1$.

Algorithm 1. Path Partition of Tournament $\mathbf{A_n = (X, U)}$

```
Initialize the number of vertices n_0 of the tournament A_{n_0}
Determine the canonical ordering (x_1, x_2, ..., x_{n_0}) of this tournament
Initialize T = {(x_3 x_4, x_1, x_2), (x_4, x_2, x_3, x_1)}
n = 5
while n ≤ n_0
P = Ø
For j = [n/2] + 1 to n – 2
μ = (x_n, x_j, ..., t) for λ = ( x_j, ..., t) ∈ T
P = P ∪ {μ}
T = T – {λ}
End for
P = P {(x_{n-1}, x_n, x_1)}
T = T ∪ P
For k = 2 to [n/2]
T = T ∪ {(x_n, x_k)}
End for
```

17 $n = n + 1$

18 end while

Theorem 4 .

The path partition in tournament A_n can be computed in $O(n^2)$ time.

Proof.

We not by C_i the cost of statement i $(1 \leq i \leq 18)$. The algorithm 1 is based on 2 consecutive For Loops. The running time of the first For Loop is less than or equal to $\frac{n}{2}(C_8 + C_9 + C_{10})$, whereas for the second For Loop the running time is $\frac{n}{2}C_{15}$. The While Loop implicates that each statement is executed n times. The running time T_n of algorithm 1 is the sum of the running time of each statement executed. Then the worst running time can be expressed as:

$$T_n \leq n\left(C_1 + C_2 + \cdots + \frac{n}{2}(C_8 + C_9 + C_{10}) + C_{13} + \frac{n}{2}C_{15} + \cdots\right)$$

Thus $T(n) = O(n^2)$

Application 1

Consider the tournament A_5 having the spanning tree $\lambda = (x_1, x_2, x_3, x_4, x_5)$

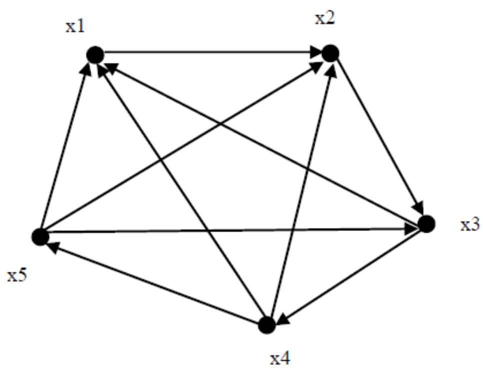

Figure 1. Spanning tree

Let $T_1 = \{(x_3x_4, x_1, x_2), (x_4, x_2, x_3, x_1)\}$ the set of arc-disjoint paths partitioning the arc of tournament A_4 generated by $X - \{x_5\}$. From T_1 we have the set of paths $T = \{(x_4, x_2, x_3, x_1), (x_4, x_5, x_1), (x_5, x_3x_4, x_1, x_2), (x_5, x_2)\}$ that partition the arcs of A_5.

2.2. Path Partition in Directed Acyclic Graph

In this section we prove that any directed acyclic graph G satisfies $P(G) = e(G)$. This result confirms that the lower bound of $P(G)$ is the best possible. The result obtained in this section will be interesting since there exist several important application areas in which directed acyclic graph arises as model: project management, assignment problem, network, etc… see Abdel Kader [2].

In the following we give a short and neat method to take advantage of directed acyclic graphs

Lemma 1.

Let $G = (X, U)$ be a graph of order n, if X included in $\Gamma_G^+(X)$, then G has at least a circuit.

Proof.

Let x_0 be a vertex of graph G, by hypothesis we have $x_0 \in \cup_{y \in X} \Gamma_G^+(y)$, then there exists at least an element $x_1 \in X$ with $x_0 \in \Gamma_G^+(x_1)\big((x_1, x_0) \in U\big)$ etc… at step $i \geq 1, x_{i-1} \in \Gamma_G^+(X)$, there exists a vertex $x_i \in X$ with $(x_i, x_{i-1}) \in U$, if x_i is a vertex already encountered we have a circuit if not the process continues and as X is finite, we will have a circuit.

Theorem 5.

Let $G = (X, U)$ be an acyclic graph, the following statements are held:

i. There exists at least a vertex x such that $d_G^-(x) = 0$

ii. The vertices of G can be arrayed in such a way the index of each vertex is less than the index of its successors.

Proof.

i. Assume $d_G^-(x) \geq 1$, for all $x \in X$, then there exists at least a vertex $x_1 \in X$ such that $x_{i-1} \in \Gamma_G^+(X)$ (for all $x \in X$), so X is included in $\Gamma_G^+(X)$, and from lemma 1, G has a circuit, contradiction then our assumption is false and there exists at least an $x \in X$, with $d_G^-(x) = 0$

ii. Let $A = \{x_{0i} \in X: d_G^-(x_{0i}) = 0\}$ and $G_1 = G_{X-A}$ the sub-graph of G induced by $X - A$. The graph G_1 is acyclic then there exists at least a vertex $x_2 \in X - A$ such that: $(x_{0k}, x_2) \in U$ for certain $x_{0k} \in A$, and so on, at step i we have the sub-graph $G_i = G_{i-1} - x_{i-1}$ of graph G. G_i is acyclic then there exists at least a vertex x_i such that$d_G^-(x_i) = 0$ and $(x_{i-1}, x_i) \in U$. Thus the vertices of G can be arrayed in such a way that the index of each vertex is less than the index of its successors.
It is obvious that the condition (ii) is equivalent to the fact that G is directed acyclic graph.

From the previous result we deduce that the vertices of the directed acyclic graph G can be indexed as: $x_1, x_2, \dots, x_n$ such that: $d_G^+(x_1) \geq d_G^+(x_2) \dots \geq d_G^+(x_n)$ where the arcs in G run from left to right. In this way we have a topological ordering of graph G.

Theorem 6. The topological ordering can be computed in $O(n + |U|)$ time.

To prove this result, enumerate the arcs of G one by one, this allows the computation of the in-degree $\big(d_G^-(x)\big)$ for all node i in linear time. Consider the array L that contains all the sources of graph G. Now execute the following algorithm, using an auxiliary list L' that is initially empty:

```
Procedure:   Typological Ordering (G)
             repeat
             for each vertex i ∈ L do
                     for each arc (i, j) ∈ U do
                             begin
                             d_G^-(j) = d_G^-(j) − 1
                             If d_G^-(j) = 0 then add j∈ L'
                             end
                     print L
                     L = L'
                     L' = ∅
                     until L = ∅
```

It is obvious that the computation takes only $O(n + |U|)$ total time since every node and every arc appears precisely one in the process.

Theorem 7.

Let $G = (X, U)$ be a directed graph of order n and $v \in X_G^+$. If G satisfies the following conditions:

i. $d_G^-(v) = 0$
ii. $P(G - v) = e(G - v)$ (that is $G - v$ has the property Q),

Then G has the property Q.

Proof.

$X_{G-v}^+ = (X_G^+ - \{v\}) \cup \left(X_G^+ \cap \Gamma_G^+(v)\right)$, and $P(G - v) = e(G - v)$. If $x \in X_{G-v}^+ \cap \Gamma_G^+(v)$, then $d_{G-v}^+(x) = d_G^+(x)$ and $d_{G-v}^-(x) = d_G^-(x) - 1$. Moreover, for $x \in X_{G-v}^+ - \left(X_{G-v}^+ \cap \Gamma_G^+(v)\right)$, we have: $d_{G-v}^+(x) = d_G^+(x)$ and $d_{G-v}^-(x) = d_G^-(x)$.

But in $G - v$, through each vertex $x \in X_{G-v}^+ \cap \Gamma_G^+(v)$ there pass $\left(d_{G-v}(x) - d_{G-v}^-(x)\right) + 1$ elementary paths of origin x which belong to a path partition R of the arcs of the digraph, the cardinal of R being $P\left(G - v\right)$. Among those paths of origin x, consider the path $\lambda = (x, \ldots)$. Since $(v, x) \in U$, the path λ allows the construction in G of the path $\mu = (v, x, \ldots)$ of origin v. Thus the number of paths of origin x in G becomes$\left(d_G^+(x) - d_G^-(x)\right)$. Moreover, for each $x \in \Gamma_G^+(v) - \left(X_{G-v}^+ \cap \Gamma_G^+(v)\right)$, we construct the path (v, x) of origin v in G. Let R' be the set of elementary paths obtained from R by cancelling those paths λ which have been used to define the path μ of origin v in G. Let T be the following set of elementary paths:

$$T = R' \cup \{\mu = (v, x, \ldots) \,|\, x \in X_{G-v}^+ \cap \Gamma_G^+(v)\} \cup \left\{(v, x) | x \in \Gamma_G^+(v) - \left(X_{G-v}^+ \cap \Gamma_G^+(v)\right)\right\}$$

It is obvious that the set T partitions the arcs of G, and we have $|T| \leq e(G) \leq P(G)$.

Therefore $P(G) = e(G)$.

Remark. If we replace the condition (i) of the Theorem by condition (i'), $d_G^+(x) = 0$, we get a similar result. Moreover the preceding Theorem allows us the construct from a digraph of order $\left(n - 1\right)$ satisfying Q, another digraph of order n still satisfying Q.

From this theorem, we deduce the following results:

Property 2

Let $G = (X, U)$ be a directed acyclic graph, then:

$P(G) = e(G)$

Proof.

By recurrence on the number n of vertices, for $n = 3, 4$ the property is true. Assume that it is true for any acyclic graph with $n - 1$ vertices and prove it for any acyclic graph having n vertices. In G there exists a vertex v such that $d_G^-(x) = 0$. The sub graph G_1 of G induced by $X - v$ is an acyclic graph with $n - 1$ vertices, then from our assumption $P\left(G - v\right) = e\left(G - v\right)$. Thus from theorem 7 we have $P(G) = e(G)$.

Remark: If TT_n is the transitive tournament then the vertices of TT_n can be arrayed as:

$X = \{x_1, x_2, \ldots, x_n\}$ where: $d_{TT_n}^+(x_i) = n - i$ for $i = 1, 2, \ldots n$.

Property 3

If TT_n, is the transitive tournament of order n, then $P(TT_n) = [n^2/4]$.

TT_n is an acyclic tournament then from Property 2 we have:

$P(TT_n) = e(TT_n) = (n - 1) + (n - 3) + \cdots + 1 = [n^2/4]$.

This result prove that the upper bound of $P(G)$ is the best possible.

The following algorithm allows finding a path partition in directed acyclic graph $G = (X, U)$.

Algorithm 2. Path Partition $(\mathbf{G})$

1 Call Topological ordering (G)

2 Determine the set S of sinks of graph G

3 $i = n - |S|$

4 Initialize $T = \{x_n, x_{n-1}, x_{n-|S|+1}\}$

5 While $i \geq 1$

6 $P = \emptyset$

7 While $x \in \Gamma_G^+(x_i)$

8 If $\lambda = (x, \dots, t) \in T$ then

9 $\mu = (x_i, x, \dots, t)$

10 $T = (T - \{\lambda\}) \cup \{\mu\}$

11 Else $P = P \cup \{(x_i, x)\}$

12 End if

13 $T = T \cup P$

14 $i = i - 1$

15 end while

Theorem 8.

The path partition in directed acyclic graph can be computed in $O(n^2)$ time.

Proof.

We not by C_i the cost of the statement i for $1 \leq i \leq 15$. The statement 1 can be executed in $T(1) = C_1(n + |U|)$ time but the maximal value of $|U|$ is around $n^2(|U| \approx n^2)$. Then $T(1) \leq C_1 n^2$. The algorithm 2 is based on 2 nested While Loops. In the worst case the internal While Loops has a running time of $n(C_8 + C_9 + \cdots + C_{13})$. Since the running time of algorithm 2 is the sum of the running time of each statement executed; from the external While Loop of algorithm 2, the worst running time $T(n)$ can be expressed as:

$$T(n) \leq C_1 n^2 + \cdots + C_7 + n\,(n(C_8 + C_9 + \cdots + C_{13})\,) + \cdots$$

$$T(n) \leq C_1 n^2 + K n^2 \leq C n^2$$

Thus $T(n) = O(n^2)$

Application 2

Consider the following acyclic graph $G = (X, U)$

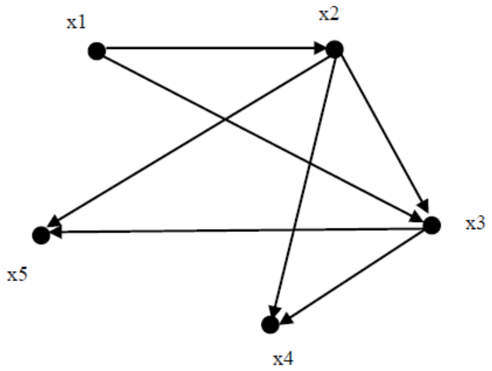

Figure 2. Acyclic Graph $G = (X, U)$

The topological ordering of graph G is $\{x_1, x_2, x_3, x_4, x_5\}$

Let $T_1 = \{(x_2, x_3, x_4), (x_3, x_5), (x_2, x_4), (x_2, x_5)\}$ be a set of arc-disjoint paths partitioning $G_1 = G - x_1$. From T_1 we have:

$$T = \{(x_1, x_2, x_3, x_4), (x_1, x_3, x_5)(x_2, x_4), (x_2, x_5)\}.$$

2.3. Computer Representation

The particular implementation for the graph G, can have a profound effect on the complexity of algorithm. In the following we give the most useful representation, for more details we refer to: [3], [14], [18].

2.3.1. Vertex Query Representation

The first representation use the Adjacency matrix $A(G) = A(n, n)$ of graph G is define as follows:

$$A(i, j) = 1 \text{ if and only if } (i, j) \in U \text{ and } 0 \text{ otherwise.}$$

The adjacency matrix requires $O(n^2)$ storage locations while retaining $O(1)$ time access to its elements.

We note that the form of adjacency matrix $A(G)$, depends on the order in which the vertices of G can be arrayed. Then we have the following result:

Theorem 9.

Two graphs G and G' are isomorphic if and only if $A(G) = A(G')$.

Proof.

If G and G' are isomorphic then $(x_i, x_j) \in U$ if and only if $\left(f(x_i), f(x_j)\right) = (y_i, y_j) \in U'$ if and only if $A(i, j) = 1$ and $A'(i, j) = 1$. Then $A(G) = A(G')$

If $A(G) = A(G')$, it is obvious that G and G' are isomorphic.

We deduce then that the order in which the adjacency matrix is written does not have any influence on the result of computation.

2.3.2. Adjacency list representations

The adjacency list of graph $G = (X, U)$ consists of an array Adj of $|X|$ lists, one of each vertex $x \in X$. Adj (x) contains all the vertices $y \in X$ such that: $(x, y) \in U$. We note that in Adj (x) the vertices are stored in any arbitrary order and are usually a more compact representation that the adjacency matrix. The sum of the length of the entire adjacency list is $|U|$. The adjacency list representation requires $O(n + |U|)$ storage locations.

The simplicity of a adjacency matrix may make it preferable when graphs are reasonable small.

If G has a particular representation, it may well be exploited to give a suitable representation in computer storage ([18]). For example if the graph $G = (X, U)$ is acyclic, from the above theorem the nodes of G can be arrayed in such a way that all arc run strictly from left to right; we obtain then the topological ordering of graph G.

Conclusion

There exists theoretical and practical reasons for studying special classes of directed graph, and it can be very interesting and worthwhile to explore a graph – algorithm problem for special types of graphs. This leads to study the very important class, we mean the networks, related to important problems in several field such as: Science, Economic, Management, Wireless network etc….. Many real problems can be modeled as path partition problems in directed graph especially in the case of network. The problem of constructing of arc-disjoint paths is a hard problem and has been studied by several authors.

In this paper, we give some properties concerning the number $P(G)$ of arc-disjoint paths partitioning the arcs of a given graph G, also certain properties characterizing the directed acyclic graph, that allow to give a structural representation of such class of graph. The result obtained facilitates the implementation of the given algorithms, so the problem of finding minimum arc-disjoint paths is of obvious interest in many network problems.

For future, research, we plan to study the most important areas related to the problem of arc-disjoint paths such as: schedule problems, network etc… especially the wireless sensor network.

In sensor network, to manage how a sensor node uses its power, we need a power management plan that allows the sensor nodes to work together in a power efficiency way, to transfer data in wireless network. In other words the goal is to extend the life time of the network by reducing the every use in the routing phase while maintaining a similar level of resilience to node failures. To achieve this objective, we need a routing protocol for energy efficiency in real-time communication over sensor network, avoiding then each sensor to work on its own.

The result obtained from these algorithms can be used to provide a reliable transmission of the entire data sent from the source to the sink over the available disjoint paths, which will be split into sub-packets corresponding to the number of available paths to ensure efficient energy consumption. Actually we work in the implementation of a new method to resolve such kind of problems.

We propose the following conjecture**:**

For any strongly connected graph G of order n, $\mathbf{P(G) \neq \left[\frac{n^2}{4}\right]}$.

References

[1] Abdel Kader I., Decomposition of directed graph, International conference on applied analysis and algebra (ICAAA), 29 June – 2 July, Istanbul 2011, Editor Irfan Siap, Adam Kilicman, Mustafa Bayram.

[2] Abdel Kader I, Directed Acyclic graph and e-Management, flight program assignment, the third international arab conference on e-Technology, proceeding of IACe-T, Zarqa Univeristy, Jordan, 25-27 April 2012, Editors: Fawarah H. And Al-Qerem A.

[3] Aho A.V., Hopcroft J.E. and Ullman J.D., the design and analysis of computer algorithms, Addison-wesley, Reading M A, 1974.

[4] Alspach Brian R., Pullman Norman J., "Path decomposition of digraphs", Bull. Austral. Math. Soc., 10(1974), 421-427.

[5] Bang-Jensen J., Gutin G., "Generalization of Tournaments, a survey, J.Graph Theory 28, 171-202, 1998.

[6] Bang-Teusen J., Gutin G., “Digraphs: Theory, Algorithms and Application”, 2^{nd} edition Springer, 2008.

[7] Beineke L.W. Wilson R.J., "A survey of recent results on the tournaments", 31-48, in Recent Advances in Graph Theory, Proc. Prague, 1974, Academia.

[8] Berge C., "Graphes et Hypergraphes", Dunod Paris 1970.

[9] Bondy J. A., Murty V.S.R., Graph Theory, Springer, 2008.

[10] Camion P., "Quelques proprites de chemins et dans circuits Hamiltonienes dus la Theorie des graphes", Cahiers de Centre d'Etude de recherche opeiatiannelles 2, 10-15, 1960.

[11] Chaty G., Chein M., "path-number of k-graphs and symmetric digraphs", proceeding of seventh southeastern conference on combinatorics, Graph Theory and Computing, 203-216, (Louisana state University, Baton-Roage, 1976, Congressus Numerantium, 17, utilities Mathematics Winnipeg, 1976.

[12] Distel R., Graph Theory, Springer, Third Edition, 2008.

[13] Douglas R.J., “Tournament that admit exactly one Hamiltonian circuit”, Proc. London Math. Soc. N^0 4, 716-730, 1970.

[14] Galperin H., and wigderson A., succinct representation of graphs, Inform and control 56(1983) pp183-198.

[15] Garey M.R. M., "On enumerating tournaments that admit exactly one Hamiltonian circuit", J. Comb. Theory B, 13, 266-269, 1972.

[16] Gutin, G. Rafieg, Yev, A. On n-partite tournament with unique n-cycle. To appear in Graph and Combination.

[17] Harary F., Moser L., « The theory of round robin tournaments” Amer. Math. Month 73, 231-246, 1966.

[18] Itai A. and Rodeh M., Representation of Graphs, Acta Inform. 17 (1982) 215-219.

[19] Las-Vergnas M., "Sur le nombre de circuits dans un tournoi fortement connexe ", Cahiers du centre d'Etude de Recherche opérationnelle Belge, Vol. 17, 261-265.

[20] Manoussakis Y., "A linear-Time Algorithm for Finding Hamiltonian cycle in Tournaments", Discrete Appl. Math. 36, 199-201, 1992.

[21] Moon J. W., "Topics on Tournaments", Holts Reinehart and Winston, 1968.

[22] O' Brien R.C., "An upper bound on the path number of a digraph", J. combinatorial theory seri B22 (1977), 168-174.

[23] Ore Oystein, "Theory of graphs", American Mathematical Society colloquium publications, 38, American Mathematical society, providence, Rhode Island, 1962.

[24] Reid K. B., Tournaments, Handbook of Graph Theory, C.R. C Press L.L.C., 2004.

[25] Reid K.B., "Tournaments scores, Kings, Generalizations and Special Topics", Congresses number 115, 171-211, 1996.

[26] Reid K.B., Beinke L.W., "Tournaments in select Topics in graph Theory, 169-204, Academia Press, London, 1979.