An Algorithm of Constructing a Bipartite Graph from a Bipartite Graphical Sequence Set

高橋 昌也 （愛知技術短期大学電子工学科）
Masaya Takahashi (Department of Electronic Engineering, Aichi College of Technology)

グラフ G＝（V, E） が2組(多重)グラフであるとは、次の（１）～（３）が成り立つような頂点集合 V が存在することである。

（１） $V=V_1 \cup V_2$, （２） $V_1 \cap V_2 = \Phi$,

（３） 任意の辺 $e=(u,v)\in E$ について、$u\in V_1$ ならば $v\in V_2$ が成り立つ。

本稿では、２つの次数列 $s_1 : d_{11}, d_{12}, \cdots, d_{1p}$, $s_2 : d_{21}, d_{22}, \cdots, d_{2q}$ と２つの頂点集合 $V_1=\{v_{11}, v_{12}, \cdots, v_{1p}\}$, $V_2=\{v_{21}, v_{22}, \cdots, v_{2q}\}$ が与えられた時、次の（１）～（３）を行うアルゴリズムを提案する。

（１） $S=(s_1, s_2)$ が２組(多重)グラフ的次数列集合であるか否かを判定する。

（２） S がそうならば、$1\leqq j\leqq p$ $(1\leqq j\leqq q)$ なる j について、v_{1j} (v_{2j}) に入っている辺の数が d_{1j} (d_{2j}) であるような２組(多重)グラフを構成する。

（３） 上記の（１）（２）を時間複雑度

$$\begin{cases} O(|V|+|E|) & \cdots\cdots \text{２組多重グラフの場合} \\ O(pq+|V|) & \cdots\cdots \text{２組グラフの場合} \end{cases}$$

で行う。但し、$|V|=p+q$ 、$|E|=\Sigma^{p}_{j=1}d_{1j}$ である。

For a graph G＝（V, E）, G is called a bipartite (multi) graph if there is a set of vertices V such that the following (1) through (3) are satisfied:

(1) $V=V_1 \cup V_2$, (2) $V_1 \cap V_2 = \Phi$, and

(3) For any edge $e=(u,v)\in E$, if $u\in V_1$ then $v\in V_2$ are satisfied.

In this paper, when two degree sequences $s_1 : d_{11}, d_{12}, \cdots, d_{1p}$, $s_2 : d_{21}, d_{22}, \cdots, d_{2q}$, and two sets of vertices $V_1=\{v_{11}, v_{12}, \cdots, v_{1p}\}$, $V_2=\{v_{21}, v_{22}, \cdots, v_{2q}\}$, are given, propose an algorithm which performs the following (1) through (3):

(1) Decide that whether a given non-negative integer sequence set $S=(s_1, s_2)$ is a degree sequence set of a bipartite (multi) graph.

(2) If $S=(s_1, s_2)$ is a degree sequence set of a bipartite (multi) graph then construct a bipartite (multi) graph $G=(V_1\cup V_2, E)$ such that $V_1=\{v_{11}, v_{12}, \cdots, v_{1p}\}$, such that $V_2=\{v_{21}, v_{22}, \cdots, v_{2q}\}$ and such that, for every j, $1\leqq j\leqq p$ ($1\leqq j\leqq q$, respectively), the degree of v_{1j} is d_{1j} (v_{2j} is d_{2j}).

(3) The time complexity of above (1) through (2) is

$$\begin{cases} O(|V|+|E|) & \text{for bipartite multigraph construction,} \\ O(pq+|V|) & \text{for bipartite graph construction,} \end{cases}$$

where $|V|=p+q$ and $|E|=\Sigma^{p}_{j=1}d_{1j}$.

In the following sections, $S=(s_1, s_2)$ is called a bipartite (multi) graphical sequence set if S is a degree sequence set of a bipartite (multi) graph.

1. Introduction

The subject of this paper is the problem of finding an algorithm of constructing a bipartite (multi) graph from a bipartite (multi) graphical sequence set : " For two given non-negative integer sequences s_1 : d_{11}, d_{12}, ····, d_{1p} ($p \geqq 1$ and $\Sigma^p{}_{j=1} d_{1j} \geqq 1$), and s_2 : d_{21}, d_{22}, ····, d_{2q} ($q \geqq 1$ and $\Sigma^q{}_{j=1} d_{2j} \geqq 1$), decide that whether $S=(s_1, s_2)$ is a bipartite (multi) graphical sequence set. If S is so then construct a bipartite (multi) graph $G=(V_1 \cup V_2, E)$ from it ", where $V_1=\{v_{11}, v_{12}, \cdots, v_{1p}\}$, $V_2=\{v_{21}, v_{22}, \cdots, v_{2q}\}$ and, for every j, $1 \leqq j \leqq p$ ($1 \leqq j \leqq q$, respectively), the degree of v_{1j} is d_{1j} (v_{2j} is d_{2j}).

In this paper, show that the bipartite multigraph construction problem (BMC-problem, for short) can be solved in linear time, and that the bipartite graph construction problem (BC-problem, for short) can be solved in polynomial time.

The problem of finding an algorithm of constructing a (multi) graph from a (multi) graphical sequence, is solved in [1][2][3][5]. In them, a polynomial time algorithm has been given by Havel and Hakimi.

In this paper, an $O(|V|+|E|)$ algorithm of solving the BMC-problem and an $O(pq+|V|)$ algorithm of solving the BC-problem are given, where $|V|=p+q$ and $|E|=\Sigma^p{}_{j=1} d_{1j}$. In the following sections, the following (1) through (2) will be discussed :

(1) Show a condition C such that a non-negative integer sequence set $S=(s_1, s_2)$ is a bipartite (multi) graphical sequence set if and only if C holds, where s_1 : d_{11}, d_{12}, ····, d_{1p}, and s_2 : d_{21}, d_{22}, ····, d_{2q}.

(2) Propose an algorithm satisfying the following (i) through (iii) (i.e., algorithm of solving the BMC-problem and the BC-problem) :

(i) Decide that whether a given non-negative integer sequence set $S=(s_1, s_2)$ is a bipartite (multi) graphical sequence set.

(ii) If $S=(s_1, s_2)$ is a bipartite (multi) graphical sequence set then construct a bipartite (multi) graph $G=(V_1 \cup V_2, E)$ such that $V_1=\{v_{11}, v_{12}, \cdots, v_{1p}\}$, such that $V_2=\{v_{21}, v_{22}, \cdots, v_{2q}\}$ and such that, for every j, $1 \leqq j \leqq p$ ($1 \leqq j \leqq q$, respectively), the degree of v_{1j} is d_{1j} (v_{2j} is d_{2j}).

(iii) The time complexity of above (i) through (ii) is

$$\begin{cases} O(|V|+|E|) & \text{for BMC-problem,} \\ O(pq+|V|) & \text{for BC-problem.} \end{cases}$$

2. Preliminaries

A graph $G=(V, E)$ consists of a finite set of vertices V and finite set of edges E such that each element of E is an unordered pair of distinct elements of V : $E=\{(u,v) \mid u,v \in V\}$.

For a graph $G=(V, E)$, G is called a bipartite (multi) graph if the following (1) through (3) are satisfied :

(1) $V=V_1 \cup V_2$, (2) $V_1 \cap V_2=\Phi$, and

(3) For any edge $e=(u,v)$, if $u \in V_1$ then $v \in V_2$ are satisfied.

For an edge $e=(u,v)$, u (v, respectively) is adjacent to v (u), u (v) is incident to e, and e is incident to v (u).

If u=v then the edge e is called a self-loop. For two edges $e_1=(u,v)$ and $e_2=(u',v')$, e_1 and e_2 are called multiple edges if and only if $e_1 \neq e_2$, $u=u'$ and $v=v'$ hold.

For a graph G, G is called a multigraph if G contains some multiple edges and no self-loop. For a graph G, G is called a simple graph (graph, for short) if G contains no multiple edge and no self-loop.

For a vertex $v \in V$, a number of edges being incident to v, is called a degree of v and it is denoted by deg(v).

A non-negative integer sequence set $S=(s_1, s_2)$ is a bipartite (multi) graphical sequence set if the following (1) through (2) are satisfied, where s_1 : $d_{11}, d_{12}, \cdots, d_{1p}$, and s_2 : $d_{21}, d_{22}, \cdots, d_{2q}$:

(1) all vertices of V_1 can be labeled $v_{11}, v_{12}, \cdots, v_{1p}$ such that the degree of v_{1j} is d_{1j} for every j, $1 \leqq j \leqq p$, and

(2) all vertices of V_2 can be labeled $v_{21}, v_{22}, \cdots, v_{2q}$ such that the degree of v_{2j} is d_{2j} for every j, $1 \leqq j \leqq q$.

3. Bipartite Multigraph

3.1 Data Structure

Suppose that $V_1=\{v_{11}, v_{12}, \cdots, v_{1p}\}$ and that $V_2=\{v_{21}, v_{22}, \cdots, v_{2q}\}$. Use an array ADJLIST containing two listheads. Each listhead represents a set of vertices V_j. The nodes in the linked lists have the form [VTX, DEG, LINK] , where VTX is a vertex number, DEG is a current degree of a vertex and LINK is a pointer field.

For example, suppose that s_j : d_{j1}, d_{j2}, d_{j3}, and that $V_j=\{v_{j1}, v_{j2}, v_{j3}\}$, for every j, $1 \leqq j \leqq 2$. Then the data structure is the following.

ADJLIST

V_1 [→] → [v_{11}, d_{11}, →] → [v_{12}, d_{12}, →] → [v_{13}, d_{13}, Λ]

V_2 [→] → [v_{21}, d_{21}, →] → [v_{22}, d_{22}, →] → [v_{23}, d_{23}, Λ]

In the following of this paper, a listhead of V_j $(1 \leqq j \leqq 2)$ is denoted by POINT(V_j), VTX of a vertex v_{jh} $(1 \leqq h \leqq p$ or $1 \leqq h \leqq q)$ is denoted by VTX(v_{jh}), DEG of a vertex v_{jh} is denoted by DEG(v_{jh}) and LINK of a vertex v_{jh} is denoted by LINK(v_{jh}).

3.2 Algorithm

In this section, discuss the algorithm for BMC-problem. The algorithm is the following.

Algorithm BMGC.

Begin

1. (1) $x \leftarrow \Sigma^p{}_{j=1} d_{1j}$; $y \leftarrow \Sigma^q{}_{j=1} d_{2j}$;

   (2) If $\{x \neq y\}$ then begin

   A sequence set $S=(s_1, s_2)$ is not a bipartite multigraphical sequence set ;

   halt end ;

2. (1) For every j, $1 \leqq j \leqq 2$, POINT(V_j) ← Λ ;

   (2) For j=1, p do begin

   If $\{d_{1j} > 0\}$ then begin

   LINK(v_{1j}) ← POINT(V_1) ; DEG(v_{1j}) ← d_{1j} ; VTX(v_{1j}) ← v_{1j} ;

   POINT(V_1) ← VTX(v_{1j}) end end ;

   (3) For j=1, q do begin

   If $\{d_{2j} > 0\}$ then begin

   LINK(v_{2j}) ← POINT(V_2) ; DEG(v_{2j}) ← d_{2j} ; VTX(v_{2j}) ← v_{2j} ;

POINT(V_2) ← VTX(v_{2j}) end end ;

(4) V_1 ← {v_{11}, v_{12}, ⋯, v_{1p}} ; V_2 ← {v_{21}, v_{22}, ⋯, v_{2q}} ; G ← $V_1 \cup V_2$;

3. (1) u ← POINT(V_1) ;

(2) while {u≠Λ} do begin

while {DEG(u)>0} do begin

PTR ← listhead of V_2 ; v ← POINT(V_2) ;

while {v≠Λ} do begin

G ← G+e, where e=(VTX(u),VTX(v)) ; DEG(u) ← DEG(u)−1 ;

DEG(v) ← DEG(v)−1 ;

If {DEG(v)=0} then begin

If {PTR=listhead of V_2} then POINT(V_2) ← LINK(v)

else LINK(PTR) ← LINK(v) end

else PTR ← v ;

v ← LINK(v) ;

If {DEG(u)=0} then go to EXIT end end ;

EXIT : u ← LINK(u) end ;

4. A bipartite multigraph G= ($V_1 \cup V_2$, E) with deg(v_{1j})=d_{1j} (with deg(v_{2j})=d_{2j}, respectively) for every j, 1≦j≦p (1≦j≦q) , is constructed ; halt

End. (Algorithm BMGC terminates.)

3.3 Time complexity

Clearly, Step 3-(2) are performed at most p times and edge addition is performed at $\Sigma^p_{j=1} d_{1j}$ times. The time complexity of Step 2-(2) is O (|V|) , where |V| =p+q.

Thus, the time complexity of Algorithm BMGC is O (|V|+|E|) , where |E| = $\Sigma^p_{j=1} d_{1j}$.

4. Bipartite Graph

4.1 Necessary and Sufficient Condition

In this section, discuss the condition C such that S= (s_1, s_2) is a bipartite graphical sequence set if and only if C holds, where s_1 : d_{11}, d_{12}, ⋯, d_{1p}, $d_{11} \leqq d_{12} \leqq \cdots \leqq d_{1p}$, $1 \leqq d_{1p} \leqq q$, s_2 : d_{21}, d_{22}, ⋯, d_{2q}, $d_{21} \leqq d_{22} \leqq \cdots \leqq d_{2q}$, $1 \leqq d_{2q} \leqq p$, are two given non-negative integer sequences.

Such the condition C is obtained by the following theorem.

Theorem 1. For two non-negative integer sequences s_1 : d_{11}, d_{12}, ⋯, d_{1p}, and s_2 : d_{21}, d_{22}, ⋯, d_{2q}, S= (s_1, s_2) is a bipartite graphical sequence set if and only if the following (1) through (2) are satisfied :

(1) $\Sigma^p_{j=1} d_{1j} = \Sigma^q_{j=1} d_{2j}$, and

(2) A sequence set S'= (s'_1, s'_2) is a bipartite graphical sequence set, where s'_1 : d_{11}, d_{12}, ⋯, $d_{1,p-1}$, and s'_2 : $d'_{21}=d_{21}$, $d'_{22}=d_{22}$, ⋯, $d'_{2t}=d_{2t}$, $d'_{2,t+1}=d_{2,t+1}-1$, ⋯, $d'_{2q}=d_{2q}-1$ ($t=q-d_{1p}$) .

Proof. Suppose that S'= (s'_1, s'_2) is a bipartite graphical sequence set. There is a bipartite graph G_1= ($V'_1 \cup V'_2$, E_1) having two sets of vertices V'_1= {v'_{11}, v'_{12}, ⋯,

$v'_{1,p-1}\}$ such that $\deg(v'_{1j})=d_{1j}$ holds for each j, $1\leqq j\leqq p-1$, and $V'_2=\{v'_{21}, v'_{22}, \cdots, v'_{2q}\}$ such that $\deg(v'_{2j})=d'_{2j}$ holds for each j, $1\leqq j\leqq q$.

Let $G=(V_1 \cup V_2, E)$ be a new bipartite graph having $V_1=V'_1 \cup \{v_{1p}\}$, $V_2=V'_2$ and $E=E_1 \cup A$, where $A=\{e_j=(v_{1p},v'_{2j}) \mid t+1\leqq j\leqq q\}$. For every vertex $v'_{1j}\in G_1$, $1\leqq j\leqq p-1$, assume that the label of v'_{1j} is replaced to $v_{1j}\in G$. Then, for G, $\deg(v_{1j})=d_{1j}$ is satisfied for every j, $1\leqq j\leqq p$, and $\deg(v'_{2j})=d_{2j}$ is satisfied for every j, $1\leqq j\leqq q$.

Since $S'=(s'_1, s'_2)$ is a bipartite graphical sequence set, $\Sigma^{p-1}_{j=1}d_{1j}=\Sigma^{q}_{j=1}d'_{2j}$ holds for G_1. Thus $\Sigma^{p}_{j=1}d_{1j}=\Sigma^{q}_{j=1}d_{2j}$ is satisfied for G.

Hence $S=(s_1, s_2)$ is a bipartite graphical sequence set.

Inversely, suppose that $S=(s_1, s_2)$ is a bipartite graphical sequence set. (i.e., suppose that there is a bipartite graph G having two sets of vertices $V_1=\{v_{11}, v_{12}, \cdots, v_{1p}\}$ such that $\deg(v_{1j})=d_{1j}$ holds for every j, $1\leqq j\leqq p$, and $V_2=\{v_{21}, v_{22}, \cdots, v_{2q}\}$ such that $\deg(v_{2j})=d_{2j}$ holds for every j, $1\leqq j\leqq q$.)

Assume that G contains a vertex $u\in V_1$ such that the following conditions (1) through (2) hold:

(1) $\deg(u)=d_{1p}$, and

(2) there is an edge (u,u_j) for every vertex $u_j\in V_2$ with $\deg(u_j)=d_{2j}$, $t+1\leqq j\leqq q$.

Then a bipartite graph $G-u$ has a sequence set $S'=(s'_1, s'_2)$, and, therefore, $S'=(s'_1, s'_2)$ is a bipartite graphical sequence set. Let the label of u be v_{1p}.

Assume that G does not contain a vertex $u\in V_1$ such that above the conditions (1) through (2) hold.

Then the following conditions (3) through (4) hold:

(3) For some j, $t+1\leqq j\leqq q$, G has a vertex $u_j\in V_2$ such that $\deg(u_j)=d_{2j}$ holds and such that there is not an edge $e_1=(v_{1p},u_j)$, and

(4) For some j, $1\leqq j\leqq t$, G has a vertex $u'_j\in V_2$ such that $\deg(u'_j)=d_{2j}$ holds and such that there is an edge $e_2=(v_{1p},u'_j)$.

Since $\deg(u'_j)\geqq 1$ and $\deg(u_j)\geqq\deg(u'_j)$, there is a vertex $v_{1k}\in V_1$ $(1\leqq k\leqq p-1)$ such that there is an edge $e_3=(v_{1k},u_j)$ and such that there is not an edge $e_4=(v_{1k},u'_j)$. Set $G'=G+\{e_1, e_4\}-\{e_2, e_3\}$. Then G' has same sequence set of G which is $S=(s_1, s_2)$.

By repeating above operation, a bipartite graph containing v_{1p} satisfying above the conditions (1) through (2), can be obtained.

By above discussion, $S'=(s'_1, s'_2)$ is a bipartite graphical sequence set.

Q. E. D.

4.2 Algorithm

By Theorem 1, an algorithm of solving the BC-problem, can be obtained directly. In this section, discuss such an algorithm.

Suppose that s_1 : $d_{11}, d_{12}, \cdots, d_{1p}$, $d_{11}\leqq d_{12}\leqq\cdots\leqq d_{1p}$, $1\leqq d_{1p}\leqq q$, that s_2 : $d_{21}, d_{22}, \cdots, d_{2q}$, $d_{21}\leqq d_{22}\leqq\cdots\leqq d_{2q}$, $1\leqq d_{2q}\leqq p$, and that $1\leqq p\leqq q$.

4.2.1 Data Structure

Suppose that $V_1=\{v_{11}, v_{12}, \cdots, v_{1p}\}$ and that $V_2=\{v_{21}, v_{22}, \cdots, v_{2q}\}$.

Use an array $ADJLIST_1$ containing two listheads, for V_j $(1\leqq j\leqq 2)$. The data structure of V_j and its vertices are the following (1) through (2):

(1) There are two listheads. j-th listhead represents a set of vertices V_j for every j, $1\leqq$

$j \leqq 2$.

(2) The nodes in the linked lists have the form [VTX, DEG, LINK], where VTX is a vertex number, DEG is a current degree of a vertex and LINK is a pointer field.

Use an array $ADJLIST_2$ containing p listheads, for V_2. The data structure of V_2 and its vertices are the following (1) through (2) :

(1) There are p listheads. Each of them represents a degree of vertices of V_2. For every k, $1 \leqq k \leqq p$, k-th element of the array indicates a node which represents a vertex v with $deg(v)=k$.

(2) The nodes in the linked lists have the form [VTX, DEG, LINK], where VTX is a vertex number, DEG is a current degree of a vertex and LINK is a pointer field.

For example, suppose that s_1 : 2, 4, 4, 4, 4, that s_2 : 2, 2, 4, 5, 5, that $V_1 = \{v_{11}, v_{12}, v_{13}, v_{14}, v_{15}\}$ and that $V_2 = \{v_{21}, v_{22}, v_{23}, v_{24}, v_{25}\}$. Then the data structure is the following.

$ADJLIST_1$

V_1 [→] → [v_{11}, 2, →] → [v_{12}, 4, →] → [v_{13}, 4, →] → [v_{14}, 4, →] → [v_{15}, 4, Λ]

V_2 [→] → [v_{21}, 2, →] → [v_{22}, 2, →] → [v_{23}, 4, →] → [v_{24}, 5, →] → [v_{25}, 5, Λ]

$ADJLIST_2$

1 [Λ]　　2 [→] → [v_{21}, 2, →] → [v_{22}, 2, Λ]　　3 [Λ]

4 [→] → [v_{23}, 4, Λ]　　5 [→] → [v_{24}, 5, →] → [v_{25}, 5, Λ]

In the following of this paper, for every j, $1 \leqq j \leqq 2$, a listhead of V_j of $ADJLIST_1$ is denoted by <u>POINT(V_j)</u> and a k-th listhead of V_2 of $ADJLIST_2$ is denoted by <u>POINT(k)</u>. For every j, $1 \leqq j \leqq 2$, VTX of a vertex v_{jh} ($1 \leqq h \leqq p$ or $1 \leqq h \leqq q$) is denoted by <u>VTX(v_{jh})</u>, DEG of a vertex v_{jh} is denoted by <u>DEG(v_{jh})</u> and LINK of a vertex v_{jh} is denoted by <u>LINK(v_{jh})</u>.

4.2.2 <u>Algorithm</u>

In this section, discuss the algorithm of solving the BC-problem. The algorithm is the following.

<u>Algorithm BGC.</u>

<u>Begin</u>

1. <u>If</u> $\{d_{1p} > q\}$ <u>or</u> $\{d_{2q} > p\}$ <u>then</u> <u>go to</u> Step 6 ; $x \leftarrow \Sigma^{p}_{j=1} d_{1j}$; $y \leftarrow \Sigma^{q}_{j=1} d_{2j}$;
   <u>If</u> $\{x \neq y\}$ <u>then</u> <u>go to</u> Step 6 ;

2. (1) For every j, $1 \leqq j \leqq 2$, POINT(V_j) ← Λ ; For every j, $1 \leqq j \leqq p$, POINT(j) ← Λ ;

   (2) <u>For</u> j=1, p <u>do</u> <u>begin</u>
   <u>If</u> $\{d_{1j} > 0\}$ <u>then</u> <u>begin</u>
   LINK(v_{1j}) ← POINT(V_1) ; DEG(v_{1j}) ← d_{1j} ; VTX(v_{1j}) ← v_{1j} ;
   POINT(V_1) ← VTX(v_{1j})　<u>end</u>　<u>end</u> ;

   (3) <u>For</u> j=1, q <u>do</u> <u>begin</u>
   <u>If</u> $\{d_{2j} > 0\}$ <u>then</u> <u>begin</u>
   LINK(v_{2j}) ← POINT(V_2) ; DEG(v_{2j}) ← d_{2j} ; VTX(v_{2j}) ← v_{2j} ;
   POINT(V_2) ← VTX(v_{2j})　<u>end</u>　<u>end</u> ;

   (4) $V_1 \leftarrow \{v_{11}, v_{12}, \cdots, v_{1p}\}$; $V_2 \leftarrow \{v_{21}, v_{22}, \cdots, v_{2q}\}$; $G \leftarrow V_1 \cup V_2$;

3. u ← POINT(V_1) ;

4. <u>while</u> {u≠Λ} <u>do</u> <u>begin</u>

   (1) v ← POINT(V_2) ;
   <u>while</u> {v≠Λ} <u>do</u> <u>begin</u>
   h ← DEG(v) ; POINT(V_2) ← LINK(v) ; LINK(v) ← POINT(h) ; POINT(h) ← VTX(v) ;
   v ← POINT(V_2)　<u>end</u> ;

(2) <u>For</u> j=p, 1, -1 <u>do</u> <u>begin</u>

(i) v ← POINT(j) ;

(i i) <u>while</u> {v≠Λ} <u>do</u> <u>begin</u>

G ← G+e, where e=(VTX(u),VTX(v)) ;

DEG(u) ← DEG(u)−1 ; DEG(v) ← DEG(v)−1 ; POINT(j) ← LINK(v) ;

<u>If</u> {DEG(v)>0} <u>then</u> <u>begin</u>

LINK(v) ← POINT(V_2) ; POINT(V_2) ← VTX(v) <u>end</u> ;

v ← POINT(j) ; <u>If</u> {DEG(u)=0} <u>then</u> <u>go to</u> step (4) <u>end</u> <u>end</u> ;

(3) <u>If</u> {DEG(u)>0} <u>then</u> <u>go to</u> Step 6 ;

(4) u ← LINK(u) <u>end</u> ;

5. A bipartite graph G= ($V_1 \cup V_2$, E) with deg(v_{1j})=d_{1j} (with deg(v_{2j})=d_{2j}, respectively) for every j, 1≦j≦p (1≦j≦q) , is constructed ; halt ;

6. A sequence set S= (s_1, s_2) is not a bipartite graphical sequence set ; halt

<u>End.</u> (Algorithm BGC terminates.)

4.3 Example

Set $d_{11}=2$, $d_{12}=3$, $d_{13}=3$, $d_{14}=5$, $d_{21}=1$, $d_{22}=1$, $d_{23}=3$, $d_{24}=4$, $d_{25}=4$ (p=4 and q=5) . Then $\Sigma^4{}_{j=1} d_{1j} = \Sigma^5{}_{j=1} d_{2j} = 13$ is obtained.

(1) By Step 2, obtain the following.

V_1 [→] → [v_{14}, 5, →] → [v_{13}, 3, →] → [v_{12}, 3, →] → [v_{11}, 2, Λ]

V_2 [→] → [v_{25}, 4, →] → [v_{24}, 4, →] → [v_{23}, 3, →] → [v_{22}, 1, →] → [v_{21}, 1, Λ]

(2) By Step 4-(1), obtain that u=v_{14} and the following.

1 [→] → [v_{21}, 1, →] → [v_{22}, 1, Λ] 2 [Λ]

3 [→] → [v_{23}, 3, Λ] 4 [→] → [v_{24}, 4, →] → [v_{25}, 4, Λ]

(3) By Step 4-(2), obtain that DEG(u)=0, and obtain five edges (v_{14},v_{25}), (v_{14},v_{24}), (v_{14},v_{23}), (v_{14},v_{22}), (v_{14},v_{21}), and the following :

V_2 [→] → [v_{23}, 2, →] → [v_{25}, 3, →] → [v_{24}, 3, Λ] .

(4) By Step 4-(1), obtain that u=v_{13} and the following.

1 [Λ] 2 [→] → [v_{23}, 2, Λ]

3 [→] → [v_{24}, 3, →] → [v_{25}, 3, Λ] 4 [Λ]

(5) By Step 4-(2), obtain that DEG(u)=0, and obtain three edges (v_{13},v_{25}), (v_{13},v_{24}), (v_{13},v_{23}), and the following : V_2 [→] → [v_{23}, 1, →] → [v_{25}, 2, →] → [v_{24}, 2, Λ] .

(6) By Step 4-(1), obtain that u=v_{12} and the following.

1 [→] → [v_{23}, 1, Λ] 2 [→] → [v_{24}, 2, →] → [v_{25}, 2, Λ]

3 [Λ] 4 [Λ]

(7) By Step 4-(2), obtain that DEG(u)=0, and obtain three edges (v_{12},v_{25}), (v_{12},v_{24}), (v_{12},v_{23}), and the following : V_2 [→] → [v_{25}, 1, →] → [v_{24}, 1, Λ] .

(8) By Step 4-(1), obtain that u=v_{11} and the following.

1 [→] → [v_{24}, 1, →] → [v_{25}, 1, Λ] 2 [Λ] 3 [Λ] 4 [Λ]

(9) By Step 4-(2), obtain that DEG(u)=0, and obtain two edges (v_{11},v_{25}), (v_{11},v_{24}), and the following : V_2 [Λ] .

(10) By Step 4-(1), obtain that u=Λ. Hence a final bipartite graph being shown in Fig.1., can be obtained.

A final graph G satisfies deg(v_{11})=1, deg(v_{12})=1, deg(v_{13})=3, deg(v_{14})=4, deg(v_{15})=4, deg(v_{21})=2, deg(v_{22})=3, deg(v_{23})=3, deg(v_{24})=5.

4.4 Time complexity

In this section, discuss the time complexity of Algorithm BGC.

When Step 4 is performed at once, Step 4-(1) and Step 4-(2)(i i) are performed at most q times, respectively. Step 4 is performed at most p times and $p \leqq q$. Hence the time complexity of Step 4 is O (pq). Clearly, the time complexity of Step 2 is $O(|V|)$, where $|V| = p+q$. Edge addition is performed at $|E|$ times, where $|E| = \Sigma^{p}_{j=1} d_{1j} = \Sigma^{q}_{j=1} d_{2j}$. Then, since G is a bipartite graph, $|E| \leqq pq$ is satisfied.

Thus, the time complexity of Algorithm BGC is $O(pq+|V|)$.

5. Conclusion

In this paper, a bipartite (multi) graph construction algorithm which performs the following (1) through (2), can be obtained :

(1) For a given sequence set $S = (s_1, s_2)$, $s_1 : d_{11}, d_{12}, \cdots, d_{1p}$, and $s_2 : d_{21}, d_{22}, \cdots, d_{2q}$, decide that whether S is a bipartite (multi) graphical sequence set, and

(2) If S is so then construct a bipartite (multi) graph from S. Then the results are

$$\begin{cases} O(|V|+|E|) & \text{for BMC-problem,} \\ O(pq+|V|) & \text{for BC-problem,} \end{cases}$$

where $|V| = p+q$ and $|E| = \Sigma^{p}_{j=1} d_{1j}$.

For any integer $k \geqq 3$, I want to find an algorithm of constructing a k-partite (multi) graph from a k-partite (multi) graphical sequence set $S = (s_1, s_2, \cdots, s_k)$ for further investigation.

References

[1] P.Erdos and T.Gallai, Graphs with prescribed degrees of vertices (Hungarian), Mat, Lapok 11(1960), 264-274

[2] S.L.Hakimi, On the realizability of a set of integers as degrees of the vertices of a graph, J.SIAM, Appl, Math, 10(1962), 496-506

[3] V.Havel, A remark on the existence of finite graphs (Czech), Casopis Pest, Mat, 80(1955), 477-480

[4] H.Frank and W.Chou, Connectivity considerations in the design of survivable networks, IEEE Trans. Circuit Theory, CT-17.(1970), 486-490

[5] M.Behzad, G.Chartrand and L.Lesnik-Foster, "Graphs and Digraphs," Prindle, Weber and Schmidt, (1979)

Fig.1