Optimal level placement of the transitive oriented graph and bipartite oriented graph by height

Stepan, Markosyan
Doct.-proff. of
Yervan State University
Yerevan, Armenia

Armen, Khachaturyan
Yerevan State University
Yerevan, Armenia
e-mail: armenarmenian@mail.ru
ph. 077302501

ABSTRACT

In this work, we will discuss level numeration (placement, arrangement) by height optimal algorithms for bipartite oriented graphs and transitive oriented graphs. There are described three definitions of the oriented graph, and for those three definitions it is solved the level numeration (placement, arrangement) problem for transitive oriented graph. The algorithm of the solution for bipartite oriented graph is of linear complexity, whereas for transitive oriented, it is of quadratic complexity.

Keywords

Transitive oriented graph, level placement.

1. INTRODUCTION

1.1. Necessary information

Definition

Let us consider $G = (V, E)$ graph. $F: V \to \{1, 2, ..., |V|\}$ one-by-one transformation is called the numeration (placement, arrangement) of graph G.

F(v) is called position or number of vertex $v \in V$.

Let us consider $G = (V, E)$ graph and its F numeration. Let us define the length of edge of the graph

$$length_F(e) = |F(v) - F(u)|, e = (u, v) \in E$$

In other words, to numerate graph G means to put its vertices equally far from each other on the line.

If $F(u) < F(p) < F(v)$ or $F(v) < F(p) < F(u)$, we will say that the edge e is passing through vertex $p \in V$.

For the given vertex $v \in V$, $h_F(v)$ will be the number of edges passing through it. Let us define the following definitions for F numeration of graph G:

length $L(F, G) = \sum_{e \in E} length_F(e)$,

width $W(F, G) = \max_{e \in E}\{length_F(e)\}$,

height $H(F, G) = \max_{v \in V}\{h_F(v)\}$.

And

length $L(G) = \min_F L(F, G)$,

width $W(G) = \min_F W(F, G)$,

height $H(G) = \min_F H(F, G)$ for graph G.

The aforementioned definitions are applied for oriented graphs as well.

The numeration for oriented graph G is called permissible, if $F(u) - F(v)$ for arbitrary $(u, v) \in E$ arc. The enumeration of an oriented graph is permissible, if its vertices are placed on the line, such that all arcs are stretched in the same direction, from left to right. Note that for a permissible numeration

$$length_F(e) = F(v) - F(u), e = (u, v) \in E$$

It is clear, that if an oriented graph contains a contour, a permissible numeration will not exist.

1.2. The placement problem for transitive oriented graphs

Let us consider $G = (V, E)$ oriented graph.

For every $v \in V$ vertex, the set $P(v) = \{u \in V; (u, v) \in E\}$ is called the preimage set of vertex v. Let us consider G contourless oriented graph. G oriented graph is called transitive if $\forall x, y, z \in V$, $(x, y) \in E, (y, z) \in E \Rightarrow (x, z) \in E$.

Let us consider the following division of vertices of oriented graph.

$$X_1 = \{v \in V; P(v) = \varnothing\}$$

$$X_2 = \{v \in V; v \notin X_1, P(v) \subseteq X_1\}$$

…..

$$X_n = \left\{v \in V; v \notin \bigcup_{i=1}^{n-1} X_i, P(V) \subseteq \bigcup_{i=1}^{n-1} X_i\right\}, n >= 2$$

The set X_k is the k-th level of oriented graph, $1 \le k \le n$.

It is clear that if an oriented graph contains a contour, such division will be impossible.

Let's discuss some specifications of oriented graph levels

- The set X_k is independent set.
- The number of oriented graph levels is greater than the longest chain by exactly 1.
- It exists only one level for each vertex, that the vertex belongs to.
- For each $v \in X_k$ vertex there is a $u \in X_{k-1}$ such that $(u, v) \in E$, $2 \le k \le n$.

- For each $v \in X_k$ vertex there is a $u \in X_1$ such that there is a path from u to v with the length $k-1, 2 \le k \le n$.

The F permissible numeration is called level numeration, if for $1 \le i < n$, the vertices of X_i come before those of X_{i+1}.

Let us assign $a(x) = |\{y \in V \mid (y,x) \in E\}|$, $b(x) = |\{y \in V \mid (x,y) \in E\}|$.

1.3. Minimal height numeration problem for oriented graphs

$G = (V, E)$ oriented graph is given. Find a F_0 permissible numeration for G, such that its height equals to the oriented graph's height $H(F_0, G) = H(G)$.

This problem is NP-complete. Minimal length and width numeration problems are also NP-complete[4-6]. The solution algorithms for some special cases of these problems are of polynomial complexity[3,7].

Let us discuss the special cases of minimal height numeration problem, the numeration for a transitive oriented graph and bipartite oriented graph by height.

2. DETAILS: OPTIMAL LEVEL ORDERINGS BY THE HEIGHT

2.1. Problem 1

For the given $G = (X, Y, E)$ bipartite oriented graph find the minimum level numeration (placement) with the height, where the height of the vertex $p \in X \bigcup Y$ for the given F numeration is $h_F(p) = |\{(u,v) \in E; F(u) < F(p) < F(v)\}|$.

It means that the height of the vertex is equal to the number of arcs, passing through it.

- Place the vertex x of the set X with the maximum b(x) at the end of the positions of set X vertices. Place the other vertices of X arbitrarily.
- Then place the vertex of the set Y with the maximum a(y), after which put the other vertices of set Y arbitrarily.

Let's enumerate the algorithm's complexity.

To find the vertex of set X with the maximum b(x) and the vertex of set Y with the maximum a(y) it will take |X|+|Y| operations. Thus the complexity of algorithm is linear.

Taking into account, that the orientation of the graph is not essential in proving algorithm 1, the latter can also be applied to solve the following problem.

Arrange the bipartite graph on the line so that the vertices of X are in the first place and then only the vertices of set T and the height of the graph placement be minimal.

Problem 1 is a special case of problem 2 and it can be solved by means of algorithm 2. But the algorithm suggested in problem 1 is simpler and more effective as there is no need to arrange all the vertices of X and Y levels, but it is sufficient to find only the vertices having the maximum b(x) of X and maximum a(y) of Y.

Now let's study the following problems.

2.2. Problem 2

For the given $G = (V, E)$ transitive oriented graph find the minimum level numeration (placement) with the height, where the height of the vertex $p \in V$ for the given F numeration is $h_F(p) = |\{(u,v) \in E; F(u) < F(p) < F(v)\}|$.

It means that the height of the vertex is equal to the number of arcs, passing through it.

Let $X_1,\ldots,X_i,\ldots X_n$ are the levels of the oriented graph.

The height of the oriented graph for arbitrary level arrangement will be equal to the maximum of the level heights. As the height of the vertices of arbitrary $X_i (1 \le i \le n)$ level is not changed after having changed the positions of the other level vertices and the contrary, we can arrange each level of oriented graph G separately to obtain optimal arrangement.

We can arrange the X_1 and X_n levels by using the algorithm 1 by the following way.

- Place the vertex of the set X_1 with the maximum b(x) at the end of the positions of set X_1 vertices. Place the other vertices of the set X_1 arbitrarily.
- Place the vertex of the set X_n with the maximum a(y) at the first of the positions of set X_n vertices. After which place the other vertices of the set X_n arbitrarily.

For the optimal ordering of the level $X_i (2 \le i \le n)$ let's suggest the following algorithm

- Place the vertices satisfying to the a(x)>b(x) condition by the a(x) descending way.
- Then place the vertices satisfying to the a(x) $\le$ b(x) condition by the b(x) ascending way.

2.3. Problem 3

For the given $G = (V, E)$ transitive oriented graph find the minimum level numeration (placement) with the height, where the height of the vertex $p \in V$ for the given F numeration is $h_F(p) = |\{(u,v) \in E; F(u) \le F(p) \le F(v)\}|$.

It means that the height of the vertex is equal to the number of arcs, passing through it or adjacent to it.

Let $X_1,\ldots,X_i,\ldots X_n$ are the levels of the oriented graph.

As in problem 2, here also we can arrange each level of oriented graph G separately to obtain optimal arrangement.

We can arrange the X_1 and X_n levels by using the algorithm 1 by the following way.

- Place the vertex of the set X_1 with the maximum b(x) at the end of the positions of set X_1 vertices. Place the other vertices of the set X_1 arbitrarily.
- Place the vertex of the set X_n with the maximum a(y) at the first of the positions of set X_n vertices. After which place the other vertices of the set X_n arbitrarily.

For the optimal ordering of the level $X_i (2 \le i \le n)$ let's suggest the following algorithm

- Place at first the vertices satisfying to the a(x)>b(x) condition by the b(x) ascending way.
- Then place the vertices satisfying to the a(x) $\le$ b(x) condition by the a(x) descending way.

2.4. Problem 4

For the given $G = (V, E)$ transitive oriented graph find the minimum level numeration (placement) with the height, where the height of the vertex $p \in V$ for the given F numeration is $h_F(p) = |\{(u, v) \in E; F(u) \le F(p) < F(v)\}|$. It means that the height of the vertex is equal to the number of arcs, passing through it or outgoing from it.

Let $X_1, \ldots, X_i, \ldots X_n$ are the levels of the oriented graph.

As in problem 2, here also we can arrange each level of oriented graph G separately to obtain optimal arrangement.

We can arrange the X_1 and X_n levels by using the algorithm 1 by the following way.

- Place the vertex of the set X_1 with the maximum b(x) at the end of the positions of set X_1 vertices. Place the other vertices of the set X_1 arbitrarily.
- Place the vertex of the set X_n with the maximum a(y) at the first of the positions of set X_n vertices. After which place the other vertices of the set X_n arbitrarily.

For the optimal ordering of the level $X_i(2 \le i \le n)$ let's suggest the following algorithm

- Place the vertices satisfying to the a(x)>b(x) condition by the b(x)-a(x) ascending way.
- Then place the vertices satisfying to the a(x) $\le$ b(x) condition arbitrarily.

Let's enumerate the complexity of algorithm 2, algorithm 3 and algorithm 4. For the division of the oriented graph to levels first $O(|V|^2)$ operations are necessary, then O(|V|log|V|) operations for the arrangement of the level's vertices according to the features mentioned in algorithms. Thus the complexities of algorithms are $O(|V|^2)$.

REFERENCES

[1]. В.А. Евстигнеев "Применение теории графов в программировании".

[2]. А.В. Петросян, С.Е. Маркосян, Ю.Г. Шукурян "Математические вопросы автоматизации и проектирования ЭВМ".Ереван,1977.

[3]. Геолецян Г.Г. "Плоское размещение вершин дерева на линии с минимизацией ширины." ДАН Апм ССР, вып. 56Но4, стр.202-207, 1973

[4]. Garey, M.R., R.L. Graham, D.S. Johnson, and D.E. Knuth "Complexity results for bandwidth minimization", SIAM J. Appl. Math. 34,477-495, 1978.

[5]. Garey, M.R., D.S. Johnson, and L. Stockmeyer "Some simplified NP-complete graph problems", Theor. Comput. Sci.1, 237-267, 1976.

[6]. Papadimitriou, Ch. H., "The NP-copleteness of the bandwidth minimization problem" Computing 16,pp.263-270, 1976.

[7]. Adolphson, D., and T. C. Hu["Optimal linear ordering" SIAM J. Appl. Math. Vol.25,No.3, pp. 403-423, Nov.1973.