PROCEEDINGS OF THE YEREVAN STATE UNIVERSITY

Physical and Mathematical Sciences 2009, № 2, p. 43–49

Informatics

OPTIMAL LEVEL PLACEMENT OF THE TRANSITIVE ORIENTED AND BIPARTITE ORIENTED GRAPHS BY HEIGHT

S.E. MARKOSYAN, A.H. KHACHATURYAN*

Chair of Discrete Mathematics and Theoretical Informatics, YSU

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

***Keywords*: transitive oriented graph, level placement.**

1. Introduction.

1.1. Necessary Information.

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

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

Let's consider $G=(V,E)$ graph and its F placement(numeration). Let's 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 place 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's define the following definitions for F placement of graph G: length $L(F,G)=\sum_{e\in E} length_F\left(e\right)$, width $W(F,G)=\max_{e\in E}\{length_F(e)\}$, height $H(F,G)=\max_{v\in V}\{h_F(v)\}$. And for graph G: length $L(G)=\min_F L(F,G)$, width $W(G)=\min_F W(F,G)$, height $H(G)=\min_F H(F,G)$.

The aforementioned definitions are applied for oriented graphs as well. The placement for oriented graph G is called permissible, if $F(u)-F(v)$ for arbitrary $(u,v)\in E$ arc. The placement of an oriented graph is permissible, if its vertices are

* E-mail: armenarmenian@mail.ru

placed on the line such that all arcs are stretched in the same direction from left to right. Note that for a permissible placement

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

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

1.2. The Placement Problem for Transitive Oriented Graphs. Let's 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's consider an oriented graph G without a contour. The oriented graph G is called transitive, if $\forall x,y,z \in V$, $(x,y) \in E,\ (y,z) \in E \ \Rightarrow (x,z) \in E$.

Let's consider the following division of vertices of oriented graph:

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

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

. . . .

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

The set X_k is the *k*-th level of oriented graph, $1 \leq k \leq n$. 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.
- There is only one level for each vertex, to which the vertex belongs.
- For each $v \in X_k$ vertex there is a $u \in X_{k-1}$ such that $(u,v) \in E$, $2 \leq k \leq n$.
- 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 $k-1,\ 2 \leq k \leq n$.

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

Let's 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 Placement Problem for Oriented Graphs. $G = (V,E)$ oriented graph is given. Find a F_0 permissible placement 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 placement problems are also NP-complete [1–3]. Some special cases are solved by polynomial complexity algorithms [4–6]. Let's discuss the special cases of minimal height placement problem, the placement for a transitive oriented graph and bipartite oriented graph by height.

2. Optimal level orderings by the height.

2.1. Problem 1. For the given $G = (X,Y,E)$ bipartite oriented graph find the minimum level placement (numeration) by the height, where the height of the vertex $p \in X \cup Y$ for the given F placement 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.

Algorithm 1 proof. The height of bipartite graph is equal to the maximum of the height of vertices of set X and set Y. Since the height of vertices of set Y is not changed by replacing the positions of vertices of set X and vice versa, so to obtain optimal placement for oriented graph G. Let's arrange the vertices of set X and vertices of set Y. Let's consider any placement of vertices of set X on the line. Let's assume by b_i the quantity of the arcs outgoing from the vertex, placed in the *i*-th position $(i = 1,...,m; m =| X |)$. Denote by $h_F(i)$ the height of the vertex placed in the *i*-th position. For the positions of vertices of set X $h_F(j) = \sum_{i=1}^{j-1} b_i$, $j = 2,...,n$, so for the arbitrary placement the maximum height will have the vertex in the last position. Since the height of the vertices of set X is less than the height of the vertex, placed on the last position, therefore, to find the optimal placement we must minimize the $h_F(m)$ by all placements: $h_F(m) \underset{F}{\rightarrow} \min$.

And since $h_F(m) = \sum_{i=1}^{m-1} b_i = \sum_{i=1}^{m} b_i - b_m$, $\sum_{i=1}^{m} b_i = const$, for the arbitrary placement, we'll have that for optimal placement b_m will be equal to $\max_{x\in X} b(x)$. Thus placing in the last position the vertex with the maximum $b(x)$, and in the other positions placing other vertices of set X arbitrarily, we'll have the optimal placement of vertices of set X.

Doing the same reasoning for the vertices of set Y, we'll have that for the optimal placement of those vertices, we must place at first the vertex $y \in Y$ with the maximum $a(y)$ $(a(y) = \max_{v\in Y} a(v))$, and then place the other vertices 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 Y, 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 consider the following problems.

2.2. Problem 2. For the given $G=(V,E)$ transitive oriented graph find the minimum level placement by the height, where the height of the vertex $p\in V$ for the given F placement 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 $X_1,\ldots,X_i,\ldots,X_n$ be the levels of the oriented graph. The height of the oriented graph for arbitrary level placement 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 placement.

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 placement 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.

Algorithm 2 proof. Let's do the following assignments for X_i layer:

$$S=\{x\in X_i \mid a(x)>b(x)\}, T=\{x\in X_i \mid a(x)\le b(x)\}.$$

Let's observe any optimal placement and show that the vertices from S come before the vertices from T. Let's consider the opposite. The vertices from T', a subset of T, is placed near the end and the vertices from S', a subset of S, are right before them (T' can be empty as well), and the vertex $x\in T$ is before them. Let's move vertex x after vertices of S'. The height of vertices of S' will be reduced by $b(x)-a(x)$ (since $x\in T\Rightarrow b(x)-a(x)\ge 0$). And instead of edges $\sum_{y\in S'} a(y)$ from the set S', edges $\sum_{y\in S'} b(y)$ will be passing through vertex x, and since $a(y)>b(y)\ \forall y\in S \ \Rightarrow \sum_{y\in S'} b(y)<\sum_{y\in S'} a(y)$, therefore, height of x will be reduced. So the height of the new placement will not be more than the previous one. Therefore, vertices of S will be placed first, and then those of T. Let's assume that the vertices of T'', a subset of T, are placed at the end, in addition, $b(t')\le b(t''),\ \forall t''\in T'',\ t'\in T',\ T'=T\setminus T''$. T'' could be empty as well. Let's put the vertices of T in order.

Let's choose the vertex x with maximal $b(x)$ from T' and move it to the end of vertices from T' (before vertices from T''). The height of vertices of

T' placed after previous position of x will be reduced by $b(x)-a(x)$, since now there will be $a(x)$ arcs, passing through those vertices instead of $b(x)$ ($a(x)\le b(x)$). And the height of that moved vertex x (which equals $\sum\limits_{x_i\in T'} b(x_i)-b(x)$ + $\sum\limits_{t\in T''} a(t)+\sum\limits_{s\in S} b(s)$) does not exceed the height of vertex z, belonging to the last position of vertices of set T' in the previous placement (which equals $\sum\limits_{x_i\in T'} b(x_i)-b(z)+\sum\limits_{t\in T''} a(t)+\sum\limits_{s\in S} b(s)$), because $b(x)\ge b(z)$.

Thus, all the vertices of set T' with the maximum $b(x)$ must be placed at the end of the positions of set T' vertices (before the vertices of set T''). Let's consider these vertices. Since for those vertices $a(x)<b(x)$ and $b(x)$ is the same for all, it can be easily proved that the most arcs would pass through the vertex, placed on the last position, the height of which does not depend on the ordering of the abovementioned vertices. Therefore, the arrangement of the vertices with the maximum $b(x)$, replaced at the end of the positions of vertices of set T', is not essential. Thus, it was proved that the vertices of set T must be placed in the $b(x)$ ascending order.

By the same reasoning we can prove that the vertices of set S must be placed in the $a(x)$ descending order.

2.3. Problem 3. For the given $G=(V,E)$ transitive oriented graph find the minimum level placement by the height, where the height of the vertex $p\in V$ for the given F placement 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 $X_1,\dots,X_i,\dots,X_n$ be 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 placement 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.

Algorithm 3 proof. Let's do the following assignments for X_i layer:

$$S=\{x\in X_i \mid a(x)>b(x)\}, T=\{x\in X_i \mid a(x)\le b(x)\}.$$

As in problem 2, here it can also be denoted that the vertices from S come before the vertices from T. Therefore, let's consider any placement satisfying that condition. Let's assume that the vertices of T'', a subset of T, are placed at the end, in addition, $a(t'') \le a(t'),\ \forall t'' \in T'',\ t' \in T',\ T' = T \setminus T''$. T'' could be empty as well. Let's put the vertices of T in order.

Let's choose the vertex x with minimal $a(x)$ from T', and move it to the end of vertices from T' (before the vertices from T''). Since $a(x) \le b(x)$, the height of vertices of T' placed after previous position of x not be increased. As the height of vertex z, which belongs to the last position of the vertices of T', is equal to $\sum_{y\in T''} a(y) + \sum_{y\in T'\cup S} b(y) + a(z)$, $\sum_{y\in T''} a(y) + \sum_{y\in T'\cup S} b(y) = const$, and for vertex x $a(z)$ was the minimum among the vertices of set T', hence, the height of x will not be greater than the height of the last vertex of the previous placement of T'.

As in problem 2, here it can also be denoted that the reciprocal arrangement of vertices with equal $a(x)$ is not essential. By the same reasoning we can arrange the vertices of S in the $b(x)$ ascending order.

Since the abovementioned is applied for arbitrary placement, the placement in the algorithm is optimal.

2.4. Problem 4. For the given $G = (V, E)$ the transitive oriented graph find the minimum level placement by the height, where the height of the vertex $p \in V$ for the given F placement 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 the optimal placement.

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 placement 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.

Algorithm 4 proof. Let's do the following assignments for X_i layer:

$$S = \{x \in X_i \mid a(x) > b(x)\},\ \ T = \{x \in X_i \mid a(x) \le b(x)\}.$$

As in problem 2, here it can also be denoted that the vertices from S come before the vertices from T. Therefore, let's consider any placement satisfying that condition. Let's assume that the vertices of S', a subset of S, are placed before those of S'', in addition, $b(s') - a(s') \le b(s'') - a(s'')$, $\forall s'' \in S''$, $s' \in S'$, $S'' = S \setminus S'$. S' could be empty as well. Let's put the vertices of S'' in order.

Let's choose the vertex x with minimal $b(x) - a(x)$ from S'', and move it immediately to the right of vertices of S' (before the vertices from S''). Since $x \in S \Rightarrow a(x) > b(x)$, the height of the remained vertices will not increase. Before moving vertex x the previous height of vertex y immediately to the right of set S' is equal to $\sum\limits_{s' \in S'} b(s') + \sum\limits_{p \in S'' \cup T} a(p) + b(y) - a(y)$, $\sum\limits_{s' \in S'} b(s') + \sum\limits_{p \in S'' \cup T} a(p) = const$. After moving x the height of x is $\sum\limits_{s' \in S'} b(s') + \sum\limits_{p \in S'' \cup T} a(p) + b(x) - a(x)$. Since $b(x) - a(x) \le b(y) - a(y)$, after moving x its height will not be greater than the previous height of vertex y. As in problem 2, here it can also be denoted that the reciprocal arrangement of vertices with equal $b(x) - a(x)$ is not essential. Thus, the vertices of S must be placed in $b(x) - a(x)$ ascending order.

Since for the vertices of set T $a(x) \le b(x)$ the rightmost vertex will have the maximum height, that is equal to $\sum\limits_{x_i \in T} b(x_i) = const$ and does not depend on the arrangement of vertices of T. Thus, the vertices of set T can be placed arbitrarily.

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

Thus, the complexities of algorithms are $O(|V|^2)$.

Received 01.04.2009

REFERENCES

1. **Garey M.R., Graham R.L., Johnson D.S. and Knuth D.E.** SIAM J. Appl. Math., 1978, v. 34, p. 477–495.
2. **Garey M.R., Johnson D.S. and Stockmeyer L.** Theor. Comput. Sci., 1976, v. 1, p. 237–267.
3. **Papadimitriou Ch.H.** Computing, 1976, v. 16, p. 263–270.
4. **Petrosyan A.V, Markosyan S.E., Shukuryan Yu.G.** Mathematical Problems of Automation and Projection of Calculating-Machine. Yer., 1977 (in Russian).
5. **Geoletsyan G.G.** DAN Arm. SSR, 1973, issue 56, № 4, p. 202–207 (in Russian).
6. **Adolphson D. and Hu T.C.** SIAM J. Appl. Math., 1973, v. 25, № 3, p. 403–423.

Proc. of the Yerevan State Univ. Phys. and Mathem. Sci., 2009, № 2, p. 43–49.

Տրանզիտիվ կողմնորոշված և երկկողմ կողմնորոշված գրաֆների օպտիմալ շերտային համարակալումը ըստ բարձրության

Աշխատանքում շարադրված են երկկողմ կողմնորոշված և տրանզիտիվ կողմնորոշված գրաֆների գագաթների ըստ բարձրության շերտային տեղադրման (համարակալման, կարգավորման) օպտիմալ ալգորիթմներ: Նկարագրված են կողմնորոշված գրաֆի բարձրության երեք սահմանումներ և երեքի համար էլ լուծված է տրանզիտիվ կողմնորոշված գրաֆի շերտային տեղադրման խնդիրը:

Երկկողմ կողմնորոշված գրաֆի շերտային տեղադրման խնդիրը լուծվել է գծային բարդության ալգորիթմով, իսկ տրանզիտիվ կողմնորոշված գրաֆի շերտային տեղադրման խնդիրները՝ քառակուսային բարդության ալգորիթմներով:

Оптимальное слойное упорядочение транзитивно ориентированного и двудольного ориентированного графов по высоте

В работе изложены оптимальные алгоритмы слойного размещения (нумерации, упорядочивания) по высоте транзитивно ориентированного и двудольного ориентированного графов. Описаны три определения высоты орграфа, и для них решена задача слойного размещения транзитивно ориентированного графа.

Задача слойного размещения двудольного орграфа решена линейным алгоритмом, а задачи слойного размещения транзитивного орграфа – квадратичными алгоритмами.