

DAGmaps and ε -Visibility Representations of DAGs

Vassilis Tsiaras and Ioannis G. Tollis

Institute of Computer Science, Foundation for Research and Technology-Hellas,
Department of Computer Science, University of Crete,
Vassilika Vouton, P.O. Box 1385, Heraklion, GR-71110 Greece
{tsiaras,tollis}@ics.forth.gr

Abstract. DAGmaps are space filling visualizations of DAGs that generalize treemaps. Deciding whether or not a DAG admits a DAGmap is NP-complete. Recently we defined a special case called one-dimensional DAGmap where the admissibility is decided in linear time. However there is no complete characterization of the class of DAGs that admit a one-dimensional DAGmap. In this paper we prove that a DAG admits a one-dimensional DAGmap if and only if it admits a directed ε -visibility representation. Then we give a characterization of the DAGs that admit directed ε -visibility representations. Finally we show that a DAGmap defines a directed three-dimensional ε -visibility representation of a DAG.

Keywords: DAGmap, Treemap, DAG, Visibility.

1 Introduction

Among the many alternative ways to visualize a tree, space filling visualizations, such as treemaps, have become very popular due to their efficiency, their scalability, and their easiness of navigation and user interaction [1]. Space filling techniques make optimal use of the available space and have the capacity to show thousands of items legibly. On the other hand, the node-link representations do not make optimal use of the available space since most of the pixels are used for background. Recently, we investigated space filling visualizations for hierarchies that are modeled by Directed Acyclic Graphs (DAG). We assumed that the available space is a rectangle and we defined the constraints for a visualization that extends the treemap techniques [3,10,1,8] to DAGs and where the vertices and edges of a DAG are drawn as rectangles [13]. In [13] we use the term “DAGmap” to describe space filling visualization according to the constraints and we show that there are DAGs that admit and DAGs that do not admit DAGmap drawings. Moreover deciding whether or not a DAG admits a DAGmap drawing is NP-complete. In the special cases, of Two Terminal Series Parallel digraphs [14] and of layered planar DAGs the admissibility question can be answered in linear time with respect to input size [13].

A visibility representation of a graph G maps vertices of G to sets in Euclidean space and the edges are expressed as visibility relations between these sets. In a (two-dimensional) visibility representation of a graph G , the vertices are drawn as horizontal segments in the plane and the edges are represented by pairs of vertically visible segments [11,5,15,7,4]. Recently, interest has developed in investigating visibility representations in three-dimensions where vertices are represented by disjoint axis-aligned closed rectangles lying in planes parallel to the xy -plane and edges correspond to z -parallel visibility among these rectangles [2]. If graph G is directed then for every edge (u, v) the rectangle of v is below the rectangle of u . Note that in order to be consistent with the downward representation of DAGs we draw the visibility representation downwards whereas in the literature it is drawn upwards [5].

In a DAGmap as well as in a visibility representation of a DAG the vertices are represented by closed rectangles and the edges are closed sets which have non-empty intersection with the source and destination vertex rectangles. In this paper we show that a DAGmap (resp. treemap) determines a directed three-dimensional visibility representation of a DAG (resp. tree). Additionally we show that there is a one-to-one correspondence between a one-dimensional DAGmap and a directed ε -visibility representation of a DAG. Using this correspondence we show that the class of DAGs that admit a one-dimensional DAGmap is the class of downward planar digraphs that admit an embedding such that all source and sink vertices appear on the boundary of the external face. Additionally we propose an admissibility and drawing algorithm that runs in $O(n)$ time.

2 Preliminaries

Let $G = (V, E)$ be a directed acyclic graph (DAG) with $n = |V|$ vertices and $m = |E|$ edges. A path of length k from a vertex u to a vertex w is a sequence $v_0, v_1, v_2, \dots, v_k$ of vertices such that $u = v_0$, $w = v_k$, and $(v_{i-1}, v_i) \in E$ for $i = 1, 2, \dots, k$. There is always a zero-length path from u to u . If there is a path p from u to w , we say that w is reachable from u via p .

A topological numbering of G is an assignment of numbers to the vertices of G , such that for every edge (u, v) of G , the number assigned to v is greater than the one assigned to u (i.e., $number(v) > number(u)$). If the edges of G have nonnegative weights assigned to them, then the number assigned to v is greater than or equal to the number assigned to u plus the weight of (u, v) (i.e., $number(v) \geq number(u) + weight(u, v)$). The numbering is optimal if the range of numbers assigned to vertices is minimized.

If $e = (u, v) \in E$ is a directed edge, we say that e is incident from u (or outgoing from u) and incident to v (or incoming to v); vertex u is the origin of e and vertex v is the destination of e . The origin of e is denoted by $orig(e)$ and the destination of e by $dest(e)$. For every vertex $u \in V$, $\Gamma^+(u) = \{e \in E \mid orig(e) = u\}$ and $\Gamma^-(u) = \{e \in E \mid dest(e) = u\}$ are the sets of edges incident from and to vertex u , respectively.

A *drawing* \mathcal{G} of a graph (digraph) G is a function which maps each vertex v to a distinct point $\mathcal{G}(v)$ and each edge (u, v) to a simple open Jordan curve $\mathcal{G}(u, v)$, with endpoints $\mathcal{G}(u)$ and $\mathcal{G}(v)$. A drawing is *planar* if no two distinct edges intersect. A graph is planar if it admits a planar drawing. A (planar) *embedding* \widehat{G} of G is an equivalence class of planar drawings and is described by the circular order of the neighbors of each vertex. An *embedded graph* is a graph with a specified embedding.

An st-graph is an acyclic digraph with a single source s and a single sink t . A planar st-graph is an st-graph that is planar and embedded with vertices s and t on the boundary of the external face.

Let \mathcal{S} be a set of horizontal non-overlapping segments in the plane. Two segments σ, σ' of \mathcal{S} are said to be visible if they can be joined by a vertical segment not intersecting any other segment of \mathcal{S} . Furthermore, σ and σ' are called ε -visible if they can be joined by a vertical band of nonzero width that does not intersect any other segment of \mathcal{S} .

Definition 1. A directed (weak) w -visibility representation for a DAG G consists of mapping each vertex v of G into a horizontal segment $\sigma(v)$ (called vertex-segment), and each edge $(u, v) \in E$ into a vertical segment $\sigma(u, v)$ (called edge-segment), so that, the vertex-segments do not overlap, and for each edge $(u, v) \in E$ the corresponding edge-segment $\sigma(u, v)$ has its top endpoint on $\sigma(u)$, its bottom endpoint on $\sigma(v)$, and it does not cross any other vertex-segment $\sigma(a), a \neq u, v$.

Definition 2. A directed ε -visibility representation for a DAG G is a directed w -visibility representation with the additional property that two vertex-segments are directed ε -visible if and only if the vertex that corresponds to the bottom vertex-segment is adjacent to the vertex that corresponds to the top vertex-segment.

Now consider an arrangement of closed, non-overlapping rectangles in \mathbb{R}^3 such that the planes determined by the rectangles are perpendicular to the z -axis, and the sides of the rectangles are parallel to the x - or y -axes. Two rectangles R_i and R_j are ε -visible if and only if between the two rectangles there is a closed cylinder C of positive height and radius such that the ends of C are contained in R_i and R_j , the axis of C is parallel to the z -axis, and the intersection of C with any other rectangle in the arrangement is empty [2].

Definition 3. [9] A directed three-dimensional ε -visibility representation for a DAG G consists of mapping each vertex v of G into a rectangle R_v (called vertex-rectangle), and each edge $(u, v) \in E$ into a vertical closed cylinder C of positive length and radius (called edge-cylinder), so that, the vertex-rectangles do not overlap, and for each edge $(u, v) \in E$ the corresponding edge-cylinder C has its top base on R_u , its bottom base on R_v , and does not intersect any other vertex-rectangle $R_w, w \neq u, v$. Additionally, two vertex-rectangles are ε -visible if and only if the vertex that corresponds to the bottom vertex-rectangle is adjacent to the vertex that corresponds to the top vertex-rectangle.

In the following R denotes the initial rectangle, R_u denotes the drawing region of a vertex $u \in V$ and R_e denotes the drawing region of an edge $e \in E$.

Definition 4 (DAGmap drawing [12,13]). A DAGmap drawing of a DAG $G = (V, E)$ is a space filling visualization of G that satisfies the following drawing constraints:

- B1. Every vertex is drawn as a rectangle (R_u is a rectangle for every $u \in V$).
- B2. The union of the rectangles of the sources of G form a partition of the initial drawing rectangle ($R = \cup_{s \in S} R_s$ and $\forall s_1, s_2 \in S$ with $s_1 \neq s_2$ $\text{area}(R_{s_1} \cap R_{s_2}) = 0$, where $S \subset V$ is the set of sources of G).
- B3. Every edge is drawn as a rectangle that has positive area ($\forall e = (u, v) \in E$, R_e is a rectangle and $\text{area}(R_e) > 0$).
- B4. The rectangle of every non-source vertex $u \in V$ is equal to the union of the rectangles of edges incident to u ($R_u = \cup_{e \in \Gamma^-(u)} R_e$).
- B5. The rectangles of edges incident from a non-sink vertex $u \in V$ form a partition of the rectangle of u ($R_u = \cup_{e \in \Gamma^+(u)} R_e$ and $\forall e_1, e_2 \in \Gamma^+(u)$ with $e_1 \neq e_2$ $\text{area}(R_{e_1} \cap R_{e_2}) = 0$).

Theorem 1. [12] In a DAGmap drawing of DAG $G = (V, E)$, if for some pair of edges $e_1, e_2 \in E$ with $e_1 \neq e_2$, it holds that that $\text{orig}(e_1)$ is not reachable from $\text{dest}(e_2)$ and $\text{orig}(e_2)$ is not reachable from $\text{dest}(e_1)$, then the rectangles R_{e_1} and R_{e_2} do not overlap (i.e., $\text{area}(R_{e_1} \cap R_{e_2}) = 0$).

Proposition 1. [13] In a DAGmap drawing of a DAG G the following hold: For every pair of vertices $u, v \in V$ if there is no path from u to v and from v to u then their rectangles R_u, R_v do not overlap ($\text{area}(R_u \cap R_v) = 0$).

3 One-Dimensional DAGmaps and Directed ε -Visibility Representations

One-dimensional DAGmaps were introduced in [13]. They are constructed by partitioning the space only along the vertical direction. We will show that one-dimensional DAGmaps are related to directed ε -visibility representations of DAGs.

Definition 5. A DAGmap is called one-dimensional if the initial drawing rectangle is sliced in one dimension either the vertical or the horizontal. See Figure 1 for an example.

Since the height of all the rectangles is constant and equal to the height of the initial drawing rectangle, the combinatorial properties of the problem are unaffected if instead of the vertex and edge rectangles R_q we consider their projections on the horizontal axis. These projections are intervals I_q .

From the vertex rectangles R_q (resp. intervals I_q) of a one-dimensional DAGmap we can construct a directed three-dimensional (resp. two-dimensional) ε -visibility representation by assigning to rectangles (resp. intervals) a z -coordinate. The construction is described in Theorem 2. A directed three-dimensional ε -visibility representation of the DAG of Fig. 1(a) is shown in Fig. 2(a). The corresponding directed (two-dimensional) ε -visibility representation of this DAG is shown in Fig. 2(b). The segments of Fig. 2(b) are the projections of the rectangles of Fig. 2(a) onto the xz -plane.

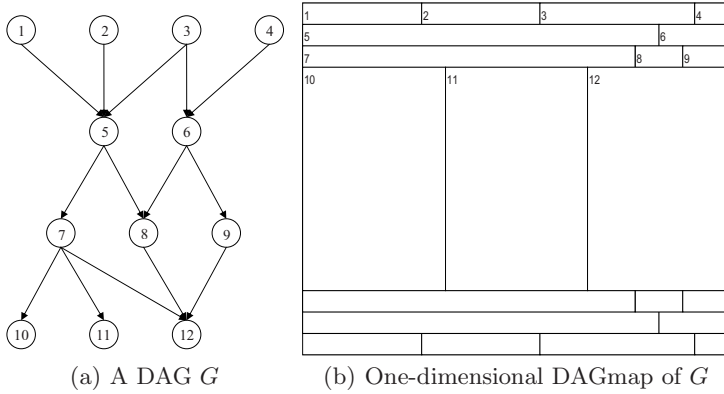


Fig. 1. An example of a one-dimensional DAGmap drawing of a DAG G . The hierarchy structure is visualized via nesting [8] along the vertical direction.

Theorem 2. *If a DAG $G = (V, E)$ admits a one-dimensional DAGmap then it admits a directed ε -visibility representation.*

Sketch of Proof. Suppose that G admits a one-dimensional DAGmap. From the one-dimensional DAGmap we will construct a directed ε -visibility representation as follows: We compute an optimal topological numbering Y of G , such that only integer numbers are used and the sources are assigned the number 0. We also compute the longest path length, h , in the DAG. Each interval I_u , $u \in V$ of the one-dimensional DAGmap is shifted along the vertical direction and is drawn on the horizontal line with equation $y = y(u) = h - Y(u) + \epsilon \cdot j(u)$, where ϵ is a small positive number (e.g. $0 < \epsilon < \frac{1}{1000 \cdot |V|}$) and $j = j(u) \in \{0, 1, \dots, |V| - 1\}$ is a unique vertex index. The shifted intervals I_u , $u \in V$ become the vertex-segments $\sigma(u)$ of the ε -visibility representation. The vertex-segments do not overlap since they are drawn on different horizontal lines. Next we add the edge-segments. For each $e = (u, v) \in E$, an edge-segment $\sigma(u, v) = \{(\mu_{uv}, y) \mid y(u) \geq y \geq y(v)\}$ is created, where μ_{uv} is the horizontal coordinate of the middle of the interval I_e . This construction is always possible since $length(I_e) > 0$.

Note that the above construction can also be done using st-numbering instead of an optimal topological numbering. The advantage in this case is that the term $\epsilon \cdot j(u)$ is not needed since the vertex rectangles have distinct y coordinates and therefore are disjoint. The disadvantages are: a) Figures 2, 3 and 4 require more space and b) some proofs become slightly longer since in order to apply Proposition 1 we need further checks to show that there is no path between two vertices in the case when two vertices have the same optimal topological numbering.

Due to space limitation we briefly outline the rest of the proof. First we show that the conditions of w -visibility are satisfied and then we prove that if

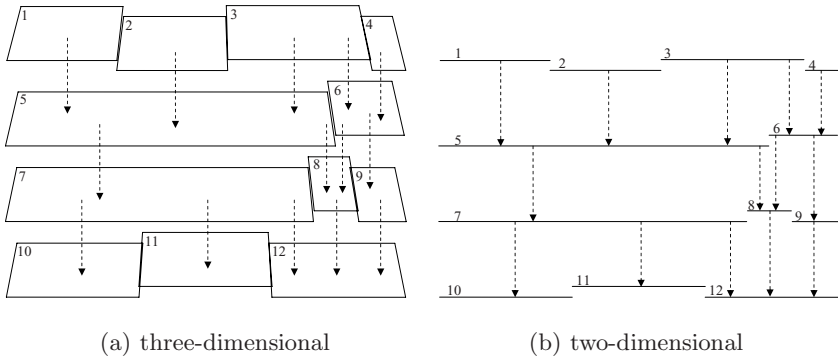


Fig. 2. Directed ε -visibility representation of the DAG of Fig. 1(a)

two vertex segments $\sigma(u)$ and $\sigma(v)$, such that $Y(u) < Y(v)$, are ε -visible then $(u, v) \in E$. Therefore the conditions of ε -visibility are satisfied. \square

4 Characterization of Directed ε -Visibility Representations

Theorem 2 reveals an interesting relationship between one-dimensional DAGmaps and ε -visibility representations. Namely a one-dimensional DAGmap defines an ε -visibility representation of a DAG. The converse of this theorem is even more interesting because a) the problem of visibility representation of a DAG has been thoroughly studied and b) it allows us to characterize the class of DAGs that admit a one-dimensional DAGmap. Before we give the converse of Theorem 2 we will characterize the class of DAGs that admit a directed ε -visibility representation. This minor result seems to be lacking from the literature. We should mention here that a complete characterization of the class of (undirected) graphs that admit an ε -visibility representation was given by Tamassia and Tollis [11]. A complete characterization of the class of digraphs that admit a (weak) w -visibility representation was given by Di Battista and Tamassia [5].

Theorem 3. [5] *A digraph G admits a directed w -visibility representation if and only if G is a subgraph of a planar st -graph.*

For directed ε -visibility representation the following lemma holds.

Lemma 1. [12] *If a DAG G admits a directed ε -visibility representation, then there exists a planar embedding \widehat{G} of G such that all source and sink vertices appear on the boundary of the external face.*

Let C_w and C_ε be the classes of DAGs that admit a directed w -visibility and a directed ε -visibility representation respectively. From the definition of directed

ε -visibility representation we have that $C_\varepsilon \subset C_w$. From Lemma 1 and Theorem 3 it follows that C_ε is properly included in C_w .

Lemma 2. [11] *For every vertex v of a planar st-graph G , the incoming (outgoing) edges appear consecutively around v .*

Now we will present an algorithm, which is based on the Algorithm Tessellation [4], that computes a directed ε -visibility representation of a planar st-graph. To describe the algorithm we need to introduce some definitions. Let G be a planar st-graph and F be its set of faces (recall that G is embedded). We conventionally assume that F contains two representatives for the external face: the *left external face* s^* , which is incident with the edges on the left boundary of G , and the *right external face* t^* , which is incident with the edges on the right boundary of G . For each element o of $V \cup E$ we define *orig*(o), *dest*(o), *left*(o), and *right*(o) as follow:

- 1) If $o = v \in V$, we define $orig(v) = dest(v) = v$. Also, with reference to Lemma 2 we denote by *left*(v) (resp. *right*(v)) the face that separate the incoming from the outgoing edges of a vertex $v \neq s, t$ in the clockwise direction (resp. counter-clockwise direction). For $v = s$ or $v = t$, we conventionally define $left(v) = s^*$ and $right(v) = t^*$.
- 2) If $o = e \in E$, we denote by *left*(e) (resp. *right*(e)) the face on the left (resp. right) side of e . Also, *orig*(e) (resp. *dest*(e)) denotes the origin (resp. destination) vertex of e .

We define a digraph G^* , associated with planar st-graph G , as follows: The vertex set of G^* is the set of faces of G . For every edge $e \neq (s, t)$ of G , G^* has an edge $e^* = (f, g)$ where $f = left(e)$ and $g = right(e)$.

Theorem 4. *Let G be a planar st-graph with n vertices. Algorithm 1 constructs a directed ε -visibility representation in $O(n)$ time.*

Sketch of Proof. For any pair of vertices $u, v \in V$, the vertex segments $\sigma(u)$ and $\sigma(v)$ do not overlap since they have distinct y coordinates.

For each edge $e = (u, v) \in E$, the corresponding maximal visibility band $b(e)$ has its top side on $\sigma(u)$ since $y_T(e) = y(u)$ and $x_L(u) \leq x_L(e) < x_R(e) \leq x_R(u)$, and its bottom side on $\sigma(v)$ since $y_B(e) = y(v)$ and $x_L(v) \leq x_L(e) < x_R(e) \leq x_R(v)$. Then we can choose a vertical band $b'(e) \subset b(e)$ of non-zero width that has its top side on $\sigma(u)$ its bottom side on $\sigma(v)$ and does not intersect with any other vertex-segment $\sigma(w)$.

Finally since the topological numbering X is optimal the vertex-segment of a non-sink vertex u is covered by the bottom sides of the maximal visibility bands of edges incident from u . Similarly the vertex-segment of a non-source vertex v is covered by the top sides of the maximal visibility bands of edges incident to v . Therefore vertex-segment $\sigma(v)$ is ε -visible from vertex-segment $\sigma(u)$ only if G has an edge (u, v) .

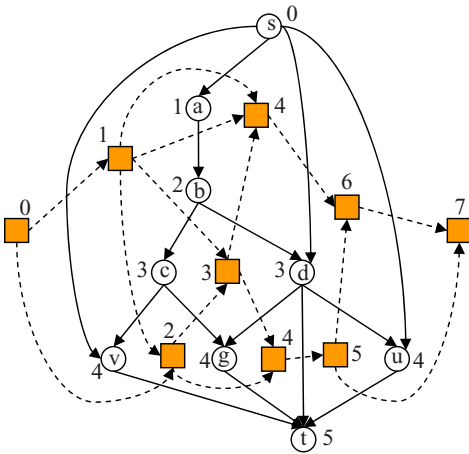
The $O(n)$ time bound follows easily since each step of the algorithm can be accomplished in $O(n)$ time. □

Algorithm 1. Directed ϵ -visibility representation

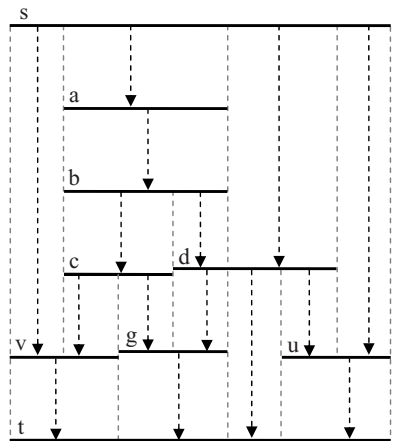
Input: a planar st-graph $G = (V, E)$

Output: a) a directed ϵ -visibility representation \mathcal{G} of G
 b) visibility bands of maximal width that their internal points do not intersect with any vertex-segment of \mathcal{G} and that each one having its top and bottom sides on two vertex-segments $s(u)$ and $s(v)$ respectively if and only if $(u, v) \in E$.

1. Construct the planar st-graph G^* .
 2. Compute an optimal topological numbering Y of G such that only integer numbers are used.
 3. Compute an optimal topological numbering X of G^* .
 4. Let ϵ be a very small positive number e.g. $0 < \epsilon < \frac{1}{1000 \cdot |V|}$
 5. $j = 0$;
 6. For each vertex $u \in V$, let the coordinates of segment $\sigma(u)$ be:
 $x_L(u) = X(left(u)); x_R(u) = X(right(u));$
 $y(u) = Y(t) - Y(u) + \epsilon \cdot j$; //perturb slightly by adding $\epsilon \cdot j$
 $j = j + 1$;
 7. For each edge $e \in E$, let the coordinates of the corresponding maximal visibility band $b(e)$ be:
 $x_L(e) = X(left(e)); x_R(e) = X(right(e));$
 $y_T(e) = y(orig(e)); y_B(e) = y(dest(e)).$
-



(a) Planar st-graphs G and G^*



(b) ϵ -visibility representation of G and maximal visibility bands

Fig. 3. Example of a run of Algorithm 1. Planar st-graphs G and G^* are labeled by topological numbering Y and X , respectively.

Let $G = (V, E)$ be a DAG and $G' = (V', E')$ be the DAG that is formed by augmenting DAG G with two vertices s' and t' and edges from s' to all sources of G and edges from all sinks of G to t' (i.e. $V' = V \cup \{s', t'\}$ and $E' = E \cup \{(s', s) \mid s \text{ is a source of } G\} \cup \{(t, t') \mid t \text{ is a sink of } G\}$). Note that in order to test for st-planarity and find an appropriate planar embedding of G' we add edge (s', t') . This edge constrains the embedding such that vertices s' and t' appear on the boundary of the same face, say the external face. The following theorem holds.

Theorem 5. [12] *DAG G admits a directed ε -visibility representation if and only if DAG G' is a planar st-graph.*

Corollary 1. *A DAG G admits a directed ε -visibility representation if and only if there exists a planar embedding \widehat{G} of G such that all source and sink vertices appear on the boundary of the external face.*

5 Characterization of One-Dimensional DAGmaps

We are now in a position to prove the converse of Theorem 2

Theorem 6. *If a DAG $G = (V, E)$ admits a directed ε -visibility representation then it admits a one-dimensional DAGmap.*

Sketch of Proof. We compute a directed ε -visibility representation \mathcal{G}' of G' using Algorithm 1 and from \mathcal{G}' a directed ε -visibility representation \mathcal{G} of G by deleting segments $\sigma(s')$ and $\sigma(t')$. From the arrangement of the vertex-segments of \mathcal{G} we will construct a one-dimensional DAGmap drawing of G . To each vertex-segment $\sigma(u), u \in V$ we correspond an interval I_u by taking its projection on the horizontal axis. For each edge $e = (u, v) \in E$ there is exactly one vertical band b of maximal width that has its bottom side on $\sigma(u)$ its top side on $\sigma(v)$ and does not ε -intersect any other segment. The coordinates of b are calculated by Algorithm 1. The edge rectangle I_e is equal to the projection of b on the horizontal axis.

We show that intervals $\{I_u \mid u \in V\}$ and $\{I_e \mid e \in E\}$ satisfy the DAGmap drawing constraints. Drawing constraints B1, B3 are clearly satisfied. Constraint B2 is satisfied due to the optimality of the topological numbering X of G^* . Constraints B4 and B5 are satisfied when the ε -visibility representation of G is produced by Algorithm 1 due to the optimality of the topological numbering X of G^* . □

Combining Theorems 2 and 6 we have the following theorem:

Theorem 7. *A DAG $G = (V, E)$ admits a one-dimensional DAGmap if and only if it admits a directed ε -visibility representation.*

Corollary 2. *The class of DAGs that admit a one-dimensional DAGmap are the planar st-graphs that admit a planar embedding such that all source and sink vertices appear on the boundary of the external face.*

Algorithm 2. One-dimensional DAGmap drawing

Input: DAG $G = (V, E)$ and drawing rectangle $R = (x_{left}, y_{bottom}, x_{right}, y_{top})$ **Output:** one-dimensional DAGmap drawing of G if G admits such a drawing or error message otherwise.

1. From DAG $G = (V, E)$ we construct an st-digraph $G' = (V', E')$, where $V' = V \cup \{s', t'\}$ and $E' = E \cup \{(s', u) \mid u \text{ is a source of } G\} \cup \{(u, t') \mid u \text{ is a sink of } G\} \cup (s', t')$.
 2. If G' is not st-planar return "DAG G does not admit a one-dimensional DAGmap."
 3. Else find a planar embedding of G' such that s' and t' appear on the boundary of the external face.
 4. Remove edge (s', t') from G'
 5. Call Algorithm 1 with input G' to compute the horizontal coordinates of vertex-segments and maximal visibility bands.
 6. Use these coordinates to fill the coordinates of vertex and edge rectangles of G .
-

Algorithm 2 recognizes whether or not a DAG admits a one-dimensional DAGmap and in the first case it constructs a one-dimensional DAGmap drawing. All steps of this algorithm can be computed in $O(n)$ time. Therefore we have the following theorem.

Theorem 8. *Algorithm 2 computes a one-dimensional DAGmap of a DAG or returns an error message in time $O(n)$ time.*

6 DAGmaps and Three Dimensional ε -Visibility Representations

A treemap determines a three-dimensional ε -visibility representation of a tree T if vertex rectangles are placed in three dimensional space such that their x and y coordinates are unaltered and their z coordinates are equal to the tree height minus the distance of the corresponding vertices from the root (plus tiny perturbations in order to keep the rectangle disjoint) (see Fig. 4). The above discussion is extended to DAGmaps as described by Theorem 9.

Theorem 9. *When a DAG G admits a DAGmap then it admits a directed three-dimensional ε -visibility representation which can be constructed by shifting the vertex rectangles along the vertical direction in such a way that their z -coordinates are determined by an optimal topological numbering of G (plus a tiny perturbation).*

The proof of Theorem 9 is similar to the proof of Theorem 2 and is omitted in this version. It is interesting to investigate if the converse of the above theorem holds. If yes, then DAGmap admissibility would be equivalent to three-dimensional visibility representation and results derived for the former problem would be valid for the second and vice versa. However the converse of Theorem 9 does not hold as the counter-example in Fig. 5 shows.

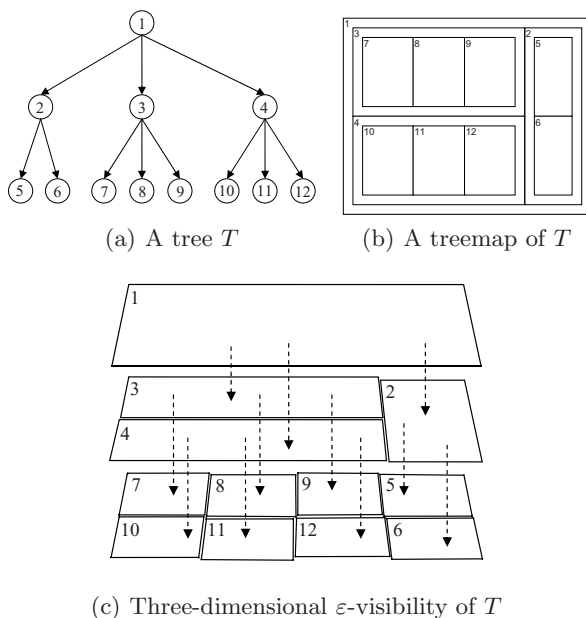


Fig. 4. A tree T , a treemap of T and a three-dimensional ε -visibility representation of T that is constructed from the treemap of T

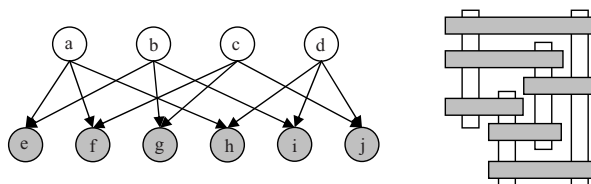


Fig. 5. The DAG of this figure does not admit a DAGmap [13]. However it admits a three-dimensional directed ε -visibility representation.

7 Discussion

In this paper we show that there is a one-to-one correspondence between a one-dimensional DAGmap and an ε -visibility representation of a DAG. Based on this correspondence we give a characterization of the class of DAGs that admit a one-dimensional DAGmap. They are those that admit a planar embedding such that all source and sink vertices appear on the boundary of the external face. Additionally we propose linear time, $O(n)$, testing and drawing algorithms for one-dimensional DAGmaps. Our next steps are a) to implement Algorithm 2 using SPQR trees [6] to find a planar embedding if one exists and b) to investigate under what restrictions the converse of Theorem 9 holds.

References

1. Bederson, B.B., Shneiderman, B., Wattenberg, M.: Ordered and quantum treemaps: Making effective use of 2D space to display hierarchies. *ACM Transactions on Graphics* 21(4), 833–854 (2002)
2. Bose, P., Everett, H., Fekete, S.P., Houle, M.E., Lubiw, A., Meijer, H., Romanik, K., Rote, G., Shermer, T.C., Whitesides, S., Zelle, C.: A visibility representation for graphs in three dimensions. *Journal of Graph Algorithms and Applications* 2, 1–16 (1998)
3. Bruls, M., Huizing, K., van Wijk, J.: Squarified treemaps. In: de Leeuw, W., van Liere, R. (eds.) *Data Visualization 2000, Proceedings of the Second Joint Eurographics and IEEE TCVG Symposium on Visualization*, pp. 33–42 (2000)
4. Di Battista, G., Eades, P., Tamassia, R., Tollis, I.G.: *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall, Upper Saddle River (1998)
5. Di Battista, G., Tamassia, R.: Algorithms for plane representations of acyclic digraphs. *Theoretical Computer Science* 61(2-3), 175–198 (1988)
6. Di Battista, G., Tamassia, R.: On-line maintenance of triconnected components with SPQR-trees. *Algorithmica* 15(4), 302–318 (1996)
7. Kant, G., Liotta, G., Tamassia, R., Tollis, I.G.: A visibility representation for graphs in three dimensions. *Area requirement of visibility representations of trees* 62, 81–88 (1997)
8. Lü, H., Fogarty, J.: Cascaded treemaps: examining the visibility and stability of structure in treemaps. In: *ACM Proceedings of graphics interface 2008*, pp. 259–266 (2008)
9. Romanik, K.: Directed VR-representable graphs have unbounded dimension. In: Tamassia, R., Tollis, I.G. (eds.) *GD 1994. LNCS*, vol. 894, pp. 177–181. Springer, Heidelberg (1995)
10. Shneiderman, B.: Tree visualization with tree-maps: 2-d space-filling approach. *ACM Transactions on Graphics* 11(1), 92–99 (1992)
11. Tamassia, R., Tollis, I.G.: A unified approach to visibility representations of planar graphs. *Discrete and Computational Geometry* 1(1), 321–341 (1986)
12. Tsiaras, V.: *Algorithms for the Analysis and Visualization of Biomedical Networks*. Ph.D. thesis, Computer Science Department, University of Crete (2009) (submitted)
13. Tsiaras, V., Triantafilou, S., Tollis, I.G.: DAGmaps: Space Filling Visualization of Directed Acyclic Graphs. *Journal of Graph Algorithms and Applications* 13(3), 319–347 (2009)
14. Valdes, J., Tarjan, R.E., Lawler, E.L.: The recognition of series parallel digraphs. *SIAM Journal on Computing* 11(2), 298–313 (1982)
15. Wismath, S.K.: Characterizing bar line-of-sight graphs. In: *SCG 1985: Proceedings of the first annual symposium on Computational geometry*, pp. 147–152. ACM, New York (1985)