Journal of Graph Algorithms and Applications
http://jgaa.info/ vol. 16, no. 2, pp. 359–380 (2012)

DAGmaps and ε-Visibility Representations for DAGs: Algorithms and Characterizations

Vassilis Tsiaras[1] *Ioannis G. Tollis*[2,1]

[1]Department of Computer Science, University of Crete, P.O. Box 2208,
Heraklion, Crete, GR-71409 Greece
[2]Institute of Computer Science,
Foundation for Research and Technology-Hellas, Vassilika Vouton, P.O. Box
1385, Heraklion, GR-71110 Greece

Abstract

DAGmaps are space filling visualizations of DAGs that generalize treemaps. Deciding whether or not a DAG admits a DAGmap is NP-complete. Although any layered planar DAG admits a one-dimensional DAGmap there was 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 characterize the class of DAGs that admit directed ε-visibility representations. This class consists of the DAGs that admit a downward planar straight-line drawing such that all source and sink vertices are assigned to the external face. Finally we show that a DAGmap defines a directed three-dimensional ε-visibility representation of a DAG.

Key words: Graph Drawing, Visibility Representations, DAGmaps, Treemaps, Planar st-Graphs

Submitted:	Reviewed:	Revised:	Accepted:
November 2011	February 2012	April 2012	April 2012
	Final:	Published:	
	May 2012	May 2012	

Article type: Regular paper
Communicated by: A. Symvonis

E-mail addresses: tsiaras@csd.uoc.gr (Vassilis Tsiaras) tollis@ics.forth.gr (Ioannis G. Tollis)

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 ease of navigation and user interaction [1]. Space filling techniques make optimal use of the available space and have the capacity of showing 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 [1, 6, 13, 17] to DAGs and where the vertices and edges of a DAG are drawn as rectangles [20]. In [20] we use the term "DAGmap" to describe a 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 [21] and of layered planar DAGs [9], the admissibility question can be answered in linear time with respect to input size [20]. Figure 1 shows two alternative DAGmaps of a DAG G. The second DAGmap, shown in Fig. 1(c), is called one dimensional because all rectangles have the same height. To illustrate the structure of the hierarchy, the rectangles are shrunk suitably in vertical, in horizontal or in both directions. This postprocessing operation is called nesting [13] and is independent from the assignment of rectangles to vertices.

In a visibility representation of a graph G, vertices of G are mapped to sets in Euclidean space and edges are expressed as visibility relations between them. 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 [8, 7, 14, 10, 16, 18, 22]. Recently, interest has developed in investigating visibility representations in three dimensions where vertices are represented by disjoint axis-aligned closed rectangles lying in planes

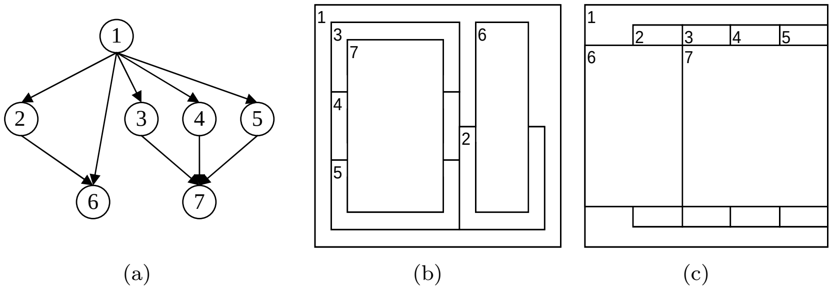


Figure 1: (a) A DAG G and two DAGmaps of G.

parallel to the xy-plane and edges correspond to z-parallel visibility among these rectangles [5]. If graph G is directed, then for every edge (u, v) the rectangle of v is below the rectangle of u.

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 ε-visibility representation of a DAG (resp. tree). Additionally we show that there is a one-to-one correspondence between the class of DAGs that admit one-dimensional DAGmaps and the class of DAGs that admit directed ε-visibility representations. Using this correspondence we show that the class of DAGs that admit a one-dimensional DAGmap is the class of downward (or upward) planar digraphs that admit a drawing such that all source and sink vertices are on the external face f^* and their incident edges on f^* form geometric angles greater than π. Additionally we propose a testing and drawing algorithm that runs in linear 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 $v_0, v_1, v_2, \ldots, v_k$ of vertices such that $u = v_0$, $w = v_k$, and $(v_{i-1}, v_i) \in E$ for $i = 1, 2, \ldots, k$. 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., $number(v) > number(u)$). A 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 $u \in V$, $E^+(u) = \{e \in E \mid orig(e) = u\}$ and $E^-(u) = \{e \in E \mid dest(e) = u\}$ are the sets of edges incident from and to vertex u, respectively.

A *drawing* Γ of a graph (digraph) G is a function which maps each vertex v to a distinct point of the plane and each edge (u, v) to a simple open Jordan curve, with endpoints u and v. A drawing is *planar* if no two edges intersect except, possibly, at common endpoints. A graph is planar if it admits a planar drawing. Two planar drawings of G are *equivalent* if they determine the same circular ordering of the edges around each vertex. An equivalence class of planar drawings is a *(combinatorial) embedding* of G. An *embedded graph* is a graph with a specified embedding. A planar drawing partitions the plane into topologically connected regions that are called *faces*. The unbounded face is the *external face*. Two drawings with the same embedding have the same faces but may still differ in the choice of the external face. A *planar embedding* consists of a combinatorial embedding and an external face.

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.

Lemma 1 *[18] For every vertex v of a planar st-graph G, the incoming (outgoing) edges appear consecutively around v.*

Let G be a planar st-graph and F be its set of faces. We conventionally assume that F contains two representatives for the external face: the *left external face* s^*, which is incident to the edges on the left boundary of G, and the *right external face* t^*, which is incident to 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 follows:

1) If $o = v \in V$, we define $orig(v) = dest(v) = v$. Also, by Lemma 1 there exist a face $left(v)$ and a face $right(v)$ that separate the incoming from the outgoing edges of a vertex $v \neq s, t$ in the clockwise direction and in counter-clockwise direction, respectively. 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 the 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 $g = right(e)$.

Lemma 2 *[7] For any two objects $o_1, o_2 \in V \cup E$ of a planar st-graph G, exactly one of the following holds:*

- *G has a directed path from $dest(o_1)$ to $orig(o_2)$*
- *G has a directed path from $dest(o_2)$ to $orig(o_1)$*
- *G^* has a directed path from $right(o_1)$ to $left(o_2)$*
- *G^* has a directed path from $right(o_2)$ to $left(o_1)$*

Let $\mathcal{S}$ be a set of horizontal non-overlapping closed line 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 width ε 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 $\sigma(q), q \neq u, v$.*

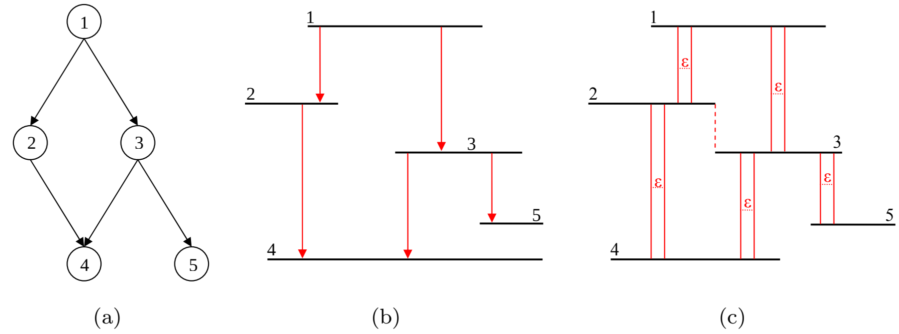


Figure 2: Visibility representations of a DAG G. (b) w-visibility representation of G. (c) ε-visibility representation of G. Segments 2 and 3 block the visibility between segments 1 and 4 without being ε-visible to each other.

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.*

Note that in order to be consistent with the downward representation of DAGs we draw a visibility representation downward whereas in the literature it is drawn upward [8]. Figure 2 shows the main difference between (weak) w- and ε- visibility representations. In w-visibility representation of a DAG G the visibility between two segments does not imply the existence of an edge in G whereas in ε-visibility representation of G two segments are ε-visible if and only if there is an edge between the corresponding vertices.

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 and R' 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 and R', the axis of C is parallel to the z-axis, and the intersection of C with any other rectangle in the arrangement is empty [5].

Definition 3 *[15] 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 $R_q, q \neq u, v$. 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.*

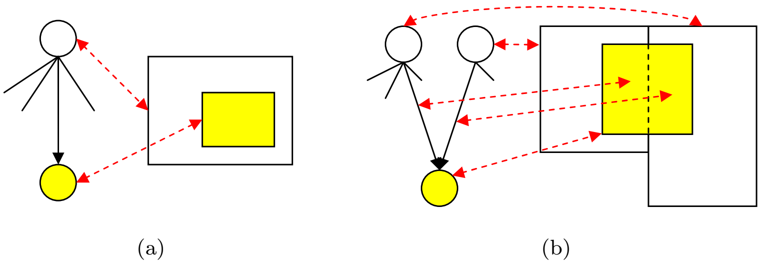


Figure 3: Illustration of the main properties of DAGmaps. (a) In treemaps the rectangle of a child vertex is included into the rectangle of its parent vertex. (b) In DAGmaps the rectangle of a vertex is included into the union of rectangles of its ancestors. Also the rectangle of an edge is contained in the intersection of the rectangles of its source and destination vertices.

In the following 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) *A DAGmap drawing of a DAG $G = (V, E)$ with a single source s is a space filling visualization of G that satisfies the following drawing constraints:*

B1. The vertices and the edges are drawn as rectangles that have positive area.

B2. The rectangle of every non-source vertex $u \in V$ is equal to the union of the rectangles of edges incident to u ($R_u = \cup_{e \in E^-(u)} R_e$).

B3. The rectangles of edges incident from a non-sink vertex $u \in V$ form a partition of the rectangle of u ($R_u = \cup_{e \in E^+(u)} R_e$ and $\forall e_1, e_2 \in E^+(u)$ with $e_1 \neq e_2 \Rightarrow area(R_{e_1} \cap R_{e_2}) = 0$).

Figure 3 shows the main properties of treemaps and of their generalizations which are called DAGmaps. In Definition 4 we require that the rectangles of edges incident from a vertex partition the rectangle associated to the vertex. However the partition property is not required for the edges incident to a vertex. Lemma 3 and Corollary 1 show that constraints B1-B3 imply this partition property.

Lemma 3 *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 $orig(e_1)$ is not reachable from $dest(e_2)$ and $orig(e_2)$ is not reachable from $dest(e_1)$, then the rectangles R_{e_1} and R_{e_2} do not overlap (i.e., $area(R_{e_1} \cap R_{e_2}) = 0$).*

Proof: Assume for a contradiction that there are two edges $e_1, e_2 \in E$, where $e_1 \neq e_2$, such that $orig(e_1)$ is not reachable from $dest(e_2)$, $orig(e_2)$ is not reachable from $dest(e_1)$ and $area(R_{e_1} \cap R_{e_2}) > 0$. The proof is accomplished by the following recursive procedure which ends after a finite number of steps.

If $orig(e_1) = orig(e_2)$ from constraint B3 we arrive at a contradiction. For this reason we assume that $orig(e_1) \neq orig(e_2)$. Then from constraint B3 we have that $R_{e_1} \subset R_{orig(e_1)}$ and $R_{e_2} \subset R_{orig(e_2)}$. Then $R_{e_1} \cap R_{e_2} \subset R_{orig(e_1)} \cap R_{orig(e_2)} \Rightarrow 0 < area(R_{e_1} \cap R_{e_2}) \leq area(R_{orig(e_1)} \cap R_{orig(e_2)})$. Now we have three cases. Case 1: If neither $orig(e_1)$ nor $orig(e_2)$ is the source vertex, then from constraint B2 we have that there is an edge e'_1 incident to vertex $orig(e_1)$ and an edge e'_2 incident to vertex $orig(e_2)$ such that $area(R_{e'_1} \cap R_{e'_2}) > 0$. We repeat the above procedure for edges e'_1 and e'_2. Case 2: $orig(e_1)$ is the source vertex and $orig(e_2)$ is a non-source vertex. Then from constraint B2 there exists an edge e'_2 incident to vertex $orig(e_2)$ such that $area(R_{e_1} \cap R_{e'_2}) > 0$. We repeat the procedure for edges e_1 and e'_2. The third case where $orig(e_1)$ is a non-source vertex and $orig(e_2)$ is the source vertex is similar to case 2 and we repeat the procedure for vertices e'_1 and e_2.

After a finite number of steps the above procedure ends and we get two edges e_k and e_l such that $area(R_{e_k} \cap R_{e_l}) > 0$ and $orig(e_k) = orig(e_l)$ which is a contradiction. □

Corollary 1 *The rectangles of edges incident to a non-source vertex $u \in V$ form a partition of the rectangle of u.*

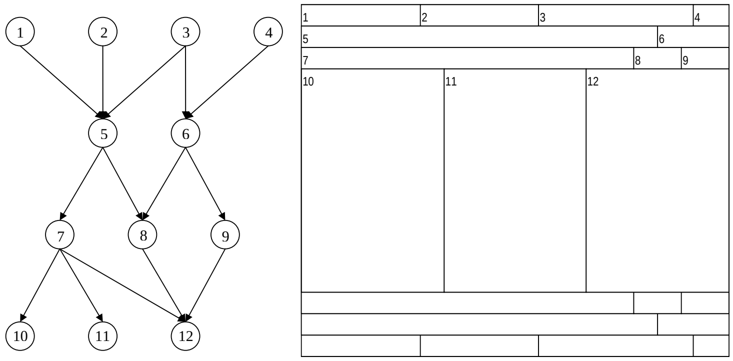
Corollary 2 *In a DAGmap drawing of a DAG G the following holds: For every pair of vertices $u, v \in V$ if there is no path from u to v or from v to u, then their rectangles R_u, R_v do not overlap ($area(R_u \cap R_v) = 0$).*

To extend Definition 4 to a DAG G with more than one source, we require that constraints B1, B2 and B3 hold and additionally that the rectangles of the sources form a partition of the available display rectangle. Then we consider DAG G' that is obtained by introducing an artificial source s' and edges from s' to all sources of G. A DAGmap of G' can be obtained from a DAGmap of G by assigning the available display rectangle to s' and the rectangle R_s to each edge (s', s), where s is a source of G. Conversely, a DAGmap of G can be obtained by a DAGmap of G' by omitting the rectangle of vertex s' and the rectangles of edges incident from s'. Therefore we have the following:

Lemma 4 *DAG G admits a DAGmap if and only if DAG G' admits a DAGmap.*

3 One-Dimensional DAGmaps and Directed ε-Visibility Representations

One-dimensional DAGmaps were introduced in [20]. 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.



(a) A DAG G (b) One-dimensional DAGmap of G

Figure 4: An example of a one-dimensional DAGmap drawing of a DAG G. The hierarchy structure is visualized via nesting along the vertical direction.

Definition 5 *A DAGmap is called one-dimensional if all rectangles have the same height (or equivalently if all rectangles have the same width).*

Figure 4 shows an example of a one-dimensional DAGmap. Since the height of all rectangles is constant, the admissibility and drawing problems 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 (resp. intervals) 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 1 and an example is shown in Fig. 5. In Fig. 5(b) the rectangles of a one-dimensional DAGmap of the DAG of Fig. 5(a) are shown. The directed three-dimensional ε-visibility representation that is constructed from these rectangles is shown in Fig. 5(c). The corresponding directed (two-dimensional) ε-visibility representation is shown in Fig. 5(d). The segments of Fig. 5(d) are the projections of the rectangles of Fig. 5(c) onto the xz-plane.

Theorem 1 *If a DAG $G = (V, E)$ admits a one-dimensional DAGmap then it admits a directed ε-visibility representation.*

Proof: Suppose that G admits a one-dimensional DAGmap. From the one-dimensional DAGmap we 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 length h of the longest path in the DAG. Each interval I_u, with $u \in V$ of the one-dimensional DAGmap is shifted along the vertical direction and is drawn on the horizontal line with equation $y = y(u) = h - Y(u) + \epsilon \cdot j$, where ϵ is a small positive number (e.g. $0 < \epsilon < \frac{1}{1000 \cdot |V|}$) and $j = j(u) \in$

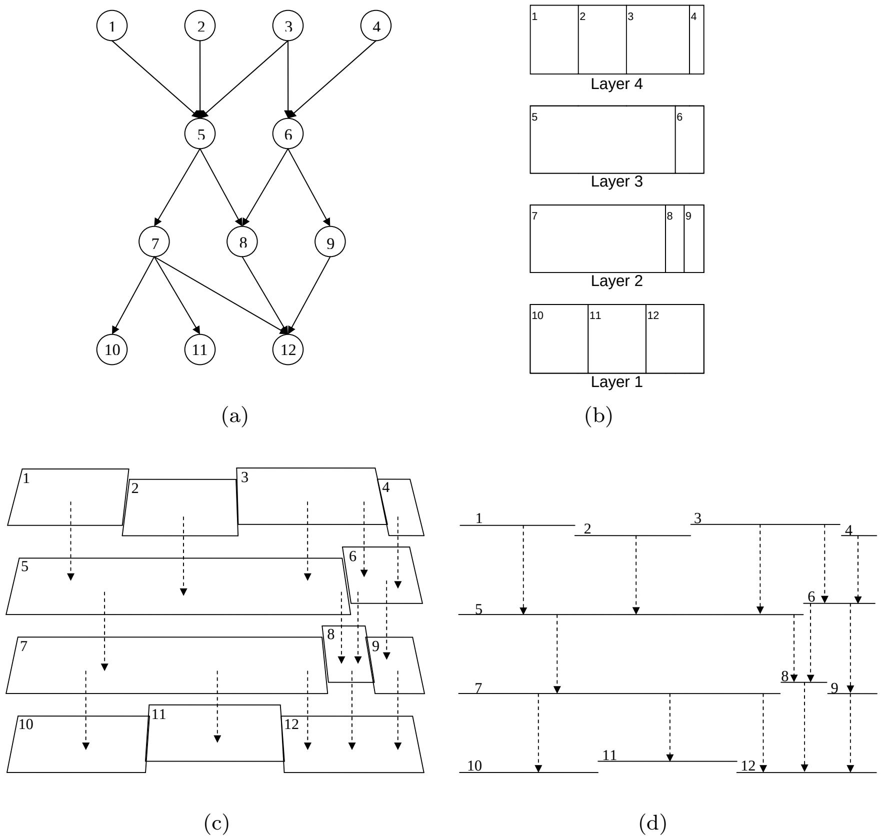


Figure 5: An example of the construction of Theorem 1. (a) A DAG G. (b) The rectangles of a one-dimensional DAGmap of G. (c) The constructed three-dimensional directed ε-visibility representation of G, and (d) the corresponding directed ε-visibility representation of G.

$\{0, 1, \cdots, |V| - 1\}$ is a unique vertex index. The shifted intervals I_u, with $u \in V$, 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 $\sigma(u, v) = \{(\mu_{uv}, y) \mid y(v) \leq y \leq y(u)\}$ is created, where μ_{uv} is the horizontal coordinate of the middle of the interval I_e (recall that I_e is the projection of edge rectangle R_e on the horizontal axis). This construction is always possible since $length(I_e) > 0$.

We will show that edge-segment $\sigma(u, v)$ does not intersect any other vertex-segment apart from vertex-segments $\sigma(u)$ and $\sigma(v)$ at its end points. Suppose for a contradiction that $\sigma(u, v)$ intersects with vertex-segment $\sigma(q)$, where $q \neq u, v$. Then we have $Y(u) \leq Y(q) \leq Y(v)$. If $Y(q) = Y(u)$ then there is no directed path between q and u and from Corollary 2 it follows that I_q does not intersect with $I_u \Rightarrow \sigma(q)$ does not intersect with $\sigma(u, v)$ (a contradiction). Similarly if $Y(q) = Y(v)$ then from Corollary 2 it follows that I_q does not intersect with $I_v \Rightarrow \sigma(q)$ does not intersect with $\sigma(u, v)$ (a contradiction). Therefore the only case that remains is $Y(u) < Y(q) < Y(v)$. The intervals I_e, where $e = (u, v)$, and I_q overlap since edge-segment $\sigma(u, v)$ intersects with vertex-segment $\sigma(q)$. Interval I_q is equal to the union of the intervals of edges incident to q. Among those intervals there is one edge interval I_{e_1} that overlaps with edge interval I_e. However there is no path from $dest(e) = v$ to $orig(e_1)$ or from $dest(e_1) = q$ to $orig(e) = u$ since $Y(orig(e_1)) < Y(dest(e))$ and $Y(orig(e)) < Y(dest(e_1))$. This contradicts Lemma 3. Therefore for each edge $(u, v) \in E$ the segment $\sigma(v)$ is visible from $\sigma(u)$, which is the condition for directed w-visibility. Moreover $\sigma(v)$ is directed ε-visible from $\sigma(u)$, since in the above arguments we can choose ε suitably small and replace the edge-segment $\sigma(u, v)$ with the vertical visibility band $b(u, v) = [\mu_{uv} - \frac{\varepsilon}{2}, \mu_{uv} + \frac{\varepsilon}{2}] \times [y(v), y(u)] = I_\varepsilon \times I_y$.

Now, suppose that two vertex-segments $\sigma(u)$ and $\sigma(v)$, such that $Y(u) < Y(v)$, are ε-visible while $(u, v) \notin E$. From the ε-visibility of $\sigma(u)$ and $\sigma(v)$ we have that vertex-segments $\sigma(u)$ and $\sigma(v)$ can be joined by a vertical band $I_\varepsilon \times I_y$, where $I_y = [y(v), y(u)]$, of width ε that does not intersect any other segment. Let $e_1, e_2, \ldots, e_k$ be the edges incident from u and $e'_1, e'_2, \ldots, e'_l$ be the edges incident to v. From constraint B3 we have: $I_u = I_{e_1} \cup \ldots \cup I_{e_k}$. Similarly, from constraint B2 we have: $I_v = I_{e'_1} \cup \ldots \cup I_{e'_l}$. Interval I_ε is contained in I_u. Therefore there is an interval $I_e \in \{I_{e_1}, \ldots, I_{e_k}\}$ such that $length(I_e \cap I_\varepsilon) \neq 0$. Since $I_e \cap I_\varepsilon \subset I_v$ it holds that there is at least one interval $I_{e'}$ among $\{I_{e'_1}, \ldots, I_{e'_l}\}$ such that $length(I_e \cap I_\varepsilon \cap I_{e'}) \neq 0 \Rightarrow length(I_e \cap I_{e'}) \neq 0$. Then from Lemma 3 we have that there is at least one path from $dest(e)$ to $orig(e')$. Since $(u, v) \notin E$, it follows that any path from u to v has length at least two. Therefore $dest(e) \neq v$. If $dest(e) = q$ then $I_e \subset I_q \Rightarrow length(I_\varepsilon \cap I_q) \neq 0 \Rightarrow$ the vertical band $I_\varepsilon \times I_y$ intersects with $\sigma(q)$ which contradicts the hypothesis that $I_\varepsilon \times I_y$ does not intersect any other segment apart from $\sigma(u)$ and $\sigma(v)$.
We proved that if two vertex segments $\sigma(u)$ and $\sigma(v)$, such that $Y(u) < Y(v)$, are ε-visible then $(u, v) \in E$. Therefore vertex segment $\sigma(v)$ is ε-visible from segment $\sigma(u)$ if and only if $(u, v) \in E$, which is the condition of directed ε-visibility representation. □

4 Directed ε-Visibility Representations and One-Dimensional DAGmaps

Theorem 1 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 representations 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 1 we will characterize the class of DAGs that admit a directed ε-visibility representation. An equivalent characterization of this class of DAGs was given by Kirkpatrick and Wismath [12]. A characterization of the class of (undirected) graphs that admit an ε-visibility representation was given by Tamassia and Tollis [18] and independently by Wismath [22]. A characterization of the class of digraphs that admit a (weak) w-visibility representation was given by Di Battista and Tamassia [8].

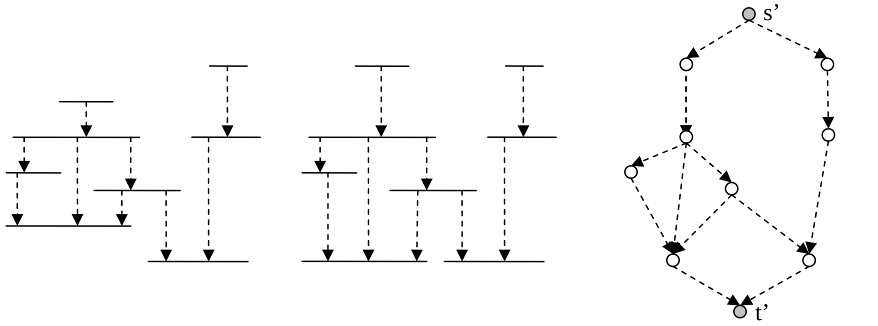
Theorem 2 *[8] A digraph G admits a directed w-visibility representation if and only if G is a subgraph of a planar st-graph.*

4.1 Directed ε-Visibility Representations and Planar st-Graphs

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'. Then the following lemma holds.

Lemma 5 *If a DAG G admits a directed ε-visibility representation, then DAG G' is a planar st-graph.*

Proof: Let $\mathcal{G}$ be a directed ε-visibility representation for G. Let $s_1, \ldots, s_k$ be the sources of G and $\sigma(s_1), \ldots, \sigma(s_k)$ the corresponding segments of $\mathcal{G}$. Similarly, let $t_1, \ldots, t_l$ be the sinks of G and $\sigma(t_1), \ldots, \sigma(t_l)$ the corresponding segments of $\mathcal{G}$. From the directed ε-visibility representation $\mathcal{G}$ construct a new directed ε-visibility representation $\mathcal{G}_1$ as follows: The segments $\sigma(s_1), \ldots, \sigma(s_k)$ are moved upwards so that all have the same height. Similarly, the segments $\sigma(t_1), \ldots, \sigma(t_l)$ are moved downwards so that all have the same height. In case that two segments intersect at their endpoints then move one of them in the vertical direction by a small amount ϵ. Then construct a planar drawing Γ of G by shrinking every vertex-segment of $\mathcal{G}_1$ into a point, and bending the edge-segments in order to maintain the adjacencies. Then add a point s' above all vertex points of Γ and join this point with the sources of Γ using straight lines. Similarly, add a point t' below all vertex points of Γ and join the sinks of Γ with this point using straight lines (see Fig. 6). Then the new drawing Γ' is planar and the vertices s' and t' are drawn on the boundary of the external face. Therefore st-graph G' is planar. □



(a) ε-visibility representation of a DAG G

(b) All segments of sources (resp. sinks) have the same height.

(c) Drawing Γ and points s' and t'

Figure 6: An example of the construction of Lemma 5.

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 $C_\varepsilon \subseteq C_w$. From Theorem 2 and Lemma 5 it follows that C_ε is properly included in C_w.

The converse of Lemma 5 can be proved by constructing a directed ε-visibility representation of a planar st-graph. The construction is accomplished by Algorithm 1, which is based on the Algorithm Tessellation [7, 19]. The input to Algorithm 1 is a planar embedding of DAG G'. Note that in order to test for st-planarity and find an appropriate planar embedding of G' we add edge (s', t') and obtain an embedding for the augmented G' graph. This edge constrains the embedding such that vertices s' and t' appear on the boundary of the same face, say the external face. After a planar embedding has been obtained this edge is removed.

Theorem 3 *Let G be a planar st-graph with n vertices. Algorithm 1 constructs a directed ε-visibility representation in $O(n)$ time.*

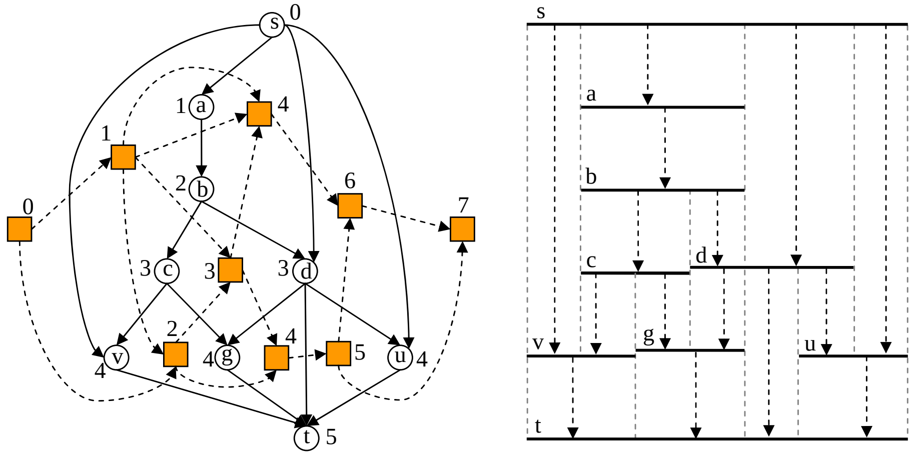
Proof: For any pair of vertices $u, v \in V$, the vertex segments $\sigma(u)$ and $\sigma(v)$ do not overlap since they have distinct y coordinates.

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 $x_L(v) \leq x_L(e) < x_R(e) \leq x_R(v)$. We will show that 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 $\sigma(q)$. For the coordinates of $b'(e)$ we choose $y'_B = y_B(e) = y(v)$, $y'_T = y_T(e) = y(u)$, and $x_L(e) < x'_L < x'_R < x_R(e)$. If there is a directed path in G from vertex q to vertex $orig(e)$ or a directed path from $dest(e)$ to q then $y(q) > y(u)$ or $y(v) > y(q)$, respectively. Therefore

Algorithm 1 Directed ε-visibility representation

```
Input:   a planar st-graph $G = (V, E)$
Output:  a) a directed $\varepsilon$-visibility representation $\mathcal{G}$ of $G$
         b) visibility bands of maximal width between segments $s(u)$ and $s(v)$
         for every $(u, v) \in E$

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 $\epsilon$ be a very small positive number and set $j = 0$
5. 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 $\epsilon \cdot j$
       $j = j + 1$;
6. 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

Figure 7: Example of a run of Algorithm 1. Planar st-graphs G and G^* are labeled by topological numberings Y and X, respectively.

vertex segment $\sigma(q)$ and vertical band $b'(e)$ do not intersect. For this we assume that (see Lemma 2) G^* either has a directed path from $right(e)$ to $left(q)$ or has a directed path from $right(q)$ to $left(e)$. Then either $x'_R < x_R(e) \leq x_L(q)$ or $x_R(q) \leq x_L(e) < x'_L$. In both cases $b'(e)$ does not intersect with $\sigma(q)$.

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. □

The construction of Algorithm 1 implies the following lemma:

Lemma 6 *If DAG G' is a planar st-graph then DAG G admits a directed ε-visibility representation.*

Proof: Suppose that G' is a planar st-graph. Then using Algorithm 1, we construct a directed ε-visibility representation $\mathcal{G}'$ for G'. If we remove the vertex-segments $\sigma(s')$ and $\sigma(t')$ from $\mathcal{G}'$ (along with their incident edge-segments) we obtain a directed ε-visibility representation for G. □

From Lemmas 5 and 6 we have:

Theorem 4 *DAG G admits a directed ε-visibility representation if and only if DAG G' is a planar st-graph.*

4.2 Directed ε-Visibility Representations and Downward Planarity

An alternative characterization of directed ε-visibility representation is based on downward (or upward) planarity.

Let f be a face of a planar drawing (or embedding) of a digraph G and suppose that the boundary of f is traversed counterclockwise. If e_1 and e_2 are two edges incident to or from a vertex v which are encountered in this order along the boundary of f, the triplet $\alpha = (e_1, v, e_2)$ is called an *angle* of f. Note that, e_1 and e_2 may coincide if G is not biconnected. Angle α is called a *switch* of f if e_1 and e_2 are both incoming edges or both outgoing edges of v: in the first case α is also called a *sink-switch* of f, while in the second case it is a *source-switch* of f. Therefore, a source-switch (sink-switch) of f is a source (sink) of f. Note that a source-switch (sink-switch) is not necessarily a source (sink) of G.

A *downward drawing* of a digraph is such that all the edges are represented by directed curves decreasing monotonically in the vertical direction. A digraph has a downward drawing if and only if it is acyclic. A digraph is *downward planar* if it admits a planar downward drawing. A downward planar digraph also admits a planar downward straight-line drawing [8, 11].

Let Γ be a planar straight-line downward drawing of an embedded downward planar digraph G. We say that a sink t (source s) of G is *assigned to* face f of Γ if the angle defined by the two edges of f incident on t (s) is greater than π. Clearly, each sink (source) can be assigned only to one face, while an internal vertex is not assigned to any face. For more information of how this intuitive idea of assignment of vertices to faces can be formally expressed as a perfect c-matching problem see [2, 3, 7].

Lemma 7 *If a DAG G admits a directed ε-visibility representation, then there exists a downward planar drawing of G such that all source and sink vertices are assigned to the external face.*

Proof: Let $\mathcal{G}$ be a directed ε-visibility representation for G. Construct a downward planar drawing of G by shrinking every vertex-segment of $\mathcal{G}$ into a point, and bending the edge-segments in order to maintain the adjacencies. Suppose, for a contradiction, that there is a source s that is not assigned to the external face. Since s is assigned to exactly one face, it follows that it is assigned to an internal face f. This means that the angle defined by the two edges of f incident on s is greater than π and therefore s is not the highest point of the boundary of f. Thus in $\mathcal{G}$ the vertex-segment $\sigma(s)$ is ε-visible from at least one other vertex-segment. Therefore s is not a source (a contradiction). The proof for sink vertices is similar. □

Lemma 8 *If there exists a downward planar drawing of G such that all source and sink vertices are assigned to the external face then G admits a directed ε-visibility representation.*

Proof: The downward planar drawing of G can be easily extended to a planar drawing of G'. Then from Theorem 4 it follows that G admits a directed ε-visibility representation. □

From Lemmas 7 and 8 we have the following theorem:

Theorem 5 *A DAG G admits a directed ε-visibility representation if and only if there exists a downward planar drawing of G such that all source and sink vertices are assigned to the external face.*

4.3 Characterization of One-Dimensional DAGmaps

We are now in a position to prove the converse of Theorem 1

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

Proof: Suppose that G admits a directed ε-visibility representation. Then 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 $\sigma(u)$, with $u \in V$ 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 its internal points do not intersect with 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. Therefore we construct intervals $\{I_u \mid u \in V\}$ and $\{I_e \mid e \in E\}$. With these intervals we can construct rectangles $\{R_u = I_u \times I_{width} \mid u \in V\}$ and $\{R_e = I_e \times I_{width} \mid e \in E\}$ with sides aligned to xy- axes and fixed vertical coordinates of their corners (see Fig. 8). We will show that these rectangles satisfy the DAGmap drawing constraints. Drawing constraint B1 is clearly satisfied. Constraints B2 and B3 are satisfied due to the optimality of the topological numbering X of G^*. □

Combining Theorems 1 and 6 we have the following theorem:

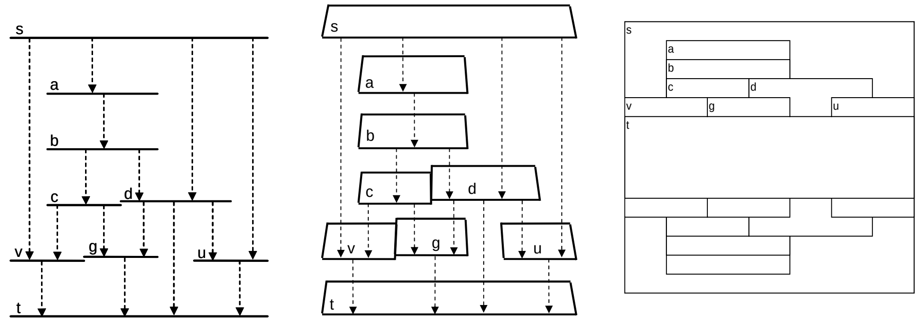
Theorem 7 *A DAG $G = (V, E)$ admits a one-dimensional DAGmap if and only if it admits a directed ε-visibility representation.*

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.

```
1. From DAG $G = (V, E)$ we construct an st-digraph $G' = (V', E')$, where
   $V' = V \cup \{s', t'\}$ and $E' = E \cup \{(s', u) \mid u$ is a source of $G\} \cup \{(u, t') \mid u$
   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.



(a) ε-visibility representation of G

(b) Constructing rectangles with sides aligned to xy-axes and constant width

(c) All rectangles are projected onto xy-plane and then nesting is used to display the hierarchy structure

Figure 8: Steps 4 and 5 of Algorithm 2 when applied to DAG of Fig. 7(a)

Theorem 8 *Algorithm 2 computes a one-dimensional DAGmap of an n-vertex DAG in $O(n)$ time, if such a drawing exists.*

Proof: The proof of Theorem 8 is very similar to the proof of Theorem 3 which proves the correctness of Algorithm 1.

The $O(n)$ time bound follows easily since each step of the algorithm runs in $O(n)$ time. In particular, the planar embedding of Step 3 can be obtained by the algorithm proposed by Bertolazzi et al. [4]. □

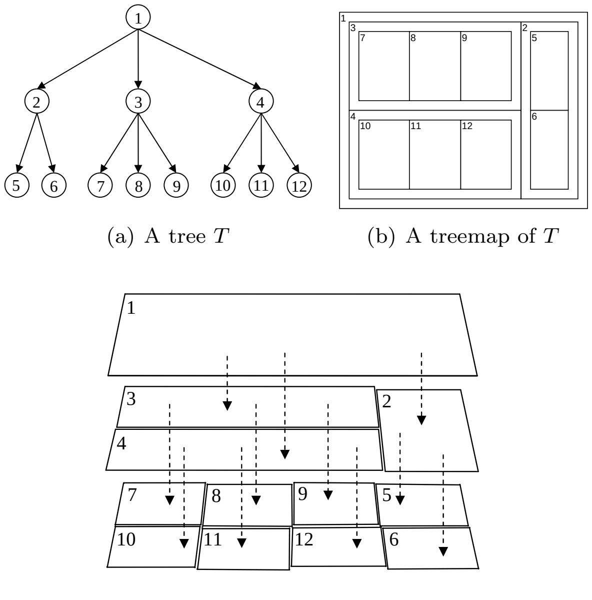
5 DAGmaps and Three-Dimensional ε-Visibility Representations

A treemap determines a three-dimensional ε-visibility representation of a tree T if the 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 rectangles disjoint) (see Fig. 9). Similarly, we construct a directed three-dimensional ε-visibility representation of a DAG G by shifting the vertex rectangles of the DAGmap along the vertical direction in such a way that their z-coordinates are determined by an optimal topological numbering of G (plus a perturbation). This construction leads to Theorem 9. The proof of Theorem 9 is similar to the proof of Theorem 1 and therefore is omitted.

Theorem 9 *If a DAG G admits a DAGmap, then it admits a directed three-dimensional ε-visibility representation.*

Notice that the converse of Theorem 9 does not hold as shown in Fig. 10. The DAG of Fig. 10 does not admit a DAGmap due to adjacency constraint violations. Since the edge rectangles form a partition of the rectangles of the origin and destination vertices, then the sink vertices e, f, g, h, i, j constrain the pairs of source vertices $\{a,b\}$, $\{a,c\}$, $\{a,d\}$, $\{b,c\}$, $\{b,d\}$, $\{c,d\}$ to be drawn in adjacent rectangles. However this configuration is impossible. On the other hand a directed three-dimensional ε-visibility representation is shown on the right side of Fig. 10.

From Theorem 9 and the counter-example of Fig. 10 it follows that the class of DAGs that admit DAGmaps is a proper subset of the class of DAGs that admit directed three-dimensional ε-visibility representations. It is interesting to investigate under what restrictions a directed three-dimensional ε-visibility representation implies a DAGmap. Without loss of generality we focus our analysis on st-graphs. From an st-graph $G = (V, E)$ we construct the st-graph $G_1 = (V_1, E_1)$, such that $V_1 = V \cup E$ and $E_1 = \{(u, e), (e, v) \mid e = (u, v) \in E\}$. In the augmented graph G_1 each edge of G has the entity of a vertex and corresponds to a rectangle in any visibility representation of G_1. Additionally we restrict the arrangement of rectangles of a visibility representation of G_1 such that the projections of the rectangles of the source and sink vertices onto the xy-plane are equal to a given rectangle R and that the projections of the



(a) A tree T

(b) A treemap of T

(c) Three-dimensional ε-visibility of T

Figure 9: A tree T, a treemap of T and a three-dimensional ε-visibility representation of T that is constructed from the treemap of T

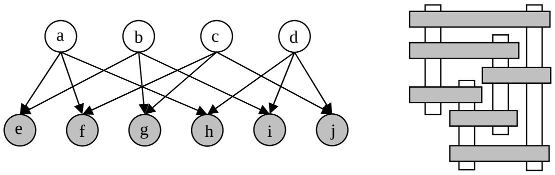


Figure 10: The DAG of this figure does not admit a DAGmap. However it admits a three-dimensional directed ε-visibility representation.

rectangles of all other vertices are restricted inside R. Under these restrictions a three-dimensional ε-visibility representation of G_1 corresponds to a DAGmap of G as Theorem 10 states.

Theorem 10 *An st-graph G admits a DAGmap if and only if the augmented st-graph G_1 admits a directed three-dimensional ε-visibility representation such that the projections of the rectangles of the source and sink vertices onto the xy-plane are equal to a given rectangle R and that the projections of the rectangles of all other vertices are restricted inside R.*

Proof: From a DAGmap of an st-graph G it is easy to construct a directed three-dimensional ε-visibility representation of G_1 by setting appropriately the z-coordinates of the vertex and edge rectangles of the DAGmap of G.

Conversely, from the directed three-dimensional ε-visibility representation of $G_1 = (V_1, E_1)$ we will construct a DAGmap of G. For each vertex $v \in V_1$ let P_v be the corresponding rectangle in the directed three-dimensional ε-visibility representation and let R_v be the projection of P_v onto the xy-plane. Now construct a DAGmap of G such that each vertex $v \in V$ is assigned the rectangle R_v and each edge $e \in E$ is assigned the rectangle R_e (recall that e is an edge of G but is also a vertex of G_1 and corresponds to a rectangle in the visibility representation). Then constraint B1 is satisfied. We will show that constraints B2 and B3 are also satisfied. First assume that constraint B2 is not satisfied. For this suppose that u is a non-source vertex and that R_u is not equal to the union of the rectangles of edges incident to u. Then R_u is not covered by the union of the rectangles of edges incident to u or the union of the rectangles of edges incident to u is not covered by R_u. If R_u is not covered by the union of the rectangles of edges incident to u, then P_u is ε-visible from a rectangle P_w but $(w, u) \notin E_1$, which is a contradiction. Note that there is always at least one such rectangle P_w since the projection onto the xy-plane of the rectangle of the source vertex of G_1 includes the projection of P_u (i.e., $R = R_s \supseteq R_u$). If the union of the rectangles of edges incident to u is not covered by R_u then there exists a rectangle P_w which is ε-visible from a rectangle P_e, where e is a vertex of G_1 and also e is an edge incident to u in G. However $(e, w) \notin E_1$, which is a contradiction. Note that there is always at least one such rectangle P_w since the projection onto the xy-plane of the rectangle of the sink vertex of G_1 includes the projection of P_e (i.e., $R = R_t \supseteq R_e$). Therefore R_u is equal to the union of the rectangles of edges incident to u and constraint B2 is satisfied.

Now, let u be a non-sink vertex. Similarly to the proof of constraint B2 we can show that R_u is equal to the union of the rectangles of edges incident from u. Also suppose that there are two edges e_1, e_2 of G with common origin the vertex u such that $area(R_{e_1} \cap R_{e_2}) \neq 0$. Then rectangles P_{e_1} and P_{e_2}, where e_1 and e_2 are treated as vertices of G_1, are either ε-visible or there are rectangles that block the visibility of P_{e_1} and P_{e_2}. The first case leads to contradiction because neither (e_1, e_2) nor (e_2, e_1) is an edge of G_1. The second case also leads to contradiction because it implies that either P_{e_1} or P_{e_2} is ε-visible from a rectangle P_w, $w \neq u$. However neither (w, e_1) nor (w, e_2) is an edge of G_1. Therefore constraint B3 is also satisfied. □

Following the proof of Theorem 10 we construct a directed three-dimensional ε-visibility representation of G by removing the rectangles of vertices $e \in E \subset V_1$ of the directed three-dimensional ε-visibility representation of G_1. This visibility representation of G implies a DAGmap of G.

6 Discussion

In this paper we show that a DAG admits a one-dimensional DAGmap if and only if it admits a directed ε-visibility representation. We also show that a DAGmap can be transformed into a directed three-dimensional ε-visibility representation of the corresponding DAG. The converse does not always hold, as shown in the counter-example of Fig. 10. Theorem 10 reveals two sufficient conditions on the arrangement of the rectangles of a directed three-dimensional ε-visibility representation under which the converse holds. The first condition is the inclusion of all (projected onto the xy-plane) rectangles in the rectangle of the source (resp. sink) of an st-graph. Note that the characterization of the classes of DAGs that admit a three-dimensional ε-visibility representation under the inclusion property is an open problem. The second condition restricts the visible part between any pair of visible rectangles to be a rectangle. When a DAG is proper layered, then this condition is satisfied in any ε-visibility representation with the inclusion property. However, it may not be satisfied when a DAG has edges that span more than two layers. Therefore, a second open problem is the characterization of the classes of DAGs that comply to both conditions and for whom the converse of Theorem 9 is true.

Acknowledgments

The authors wish to thank the anonymous referees for several useful suggestions that improved the quality and appearance of the paper.

References

[1] B. B. Bederson, B. Shneiderman, and M. Wattenberg. Ordered and quantum treemaps: Making effective use of 2D space to display hierarchies. *ACM Transactions on Graphics*, 21(4):833–854, 2002.

[2] P. Bertolazzi and G. Di Battista. On upward drawing testing of triconnected digraphs. In *Proceedings of the seventh annual symposium on computational geometry*, pages 272–280. ACM, 1991.

[3] P. Bertolazzi, G. Di Battista, G. Liotta, and C. Mannino. Upward drawings of triconnected digraphs. *Algorithmica*, 12(6):476–497, 1994.

[4] P. Bertolazzi, G. Di Battista, C. Mannino, and R. Tamassia. Optimal upward planarity testing of single-source digraphs. *SIAM Journal on Computing*, 27(1):132–169, 1998.

[5] P. Bose, H. Everett, S. P. Fekete, M. E. Houle, A. Lubiw, H. Meijer, K. Romanik, G. Rote, T. C. Shermer, S. Whitesides, and C. Zelle. A visibility representation for graphs in three dimensions. *Journal of Graph Algorithms and Applications*, 2(3):1–16, 1998.

[6] M. Bruls, K. Huizing, and J. van Wijk. Squarified treemaps. In *Data Visualization 2000, Proceedings of the Second Joint Eurographics and IEEE TCVG Symposium on Visualization*, pages 33–42, 2000.

[7] G. Di Battista, P. Eades, R. Tamassia, and I. G. Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall, Upper Saddle River, NJ, 1998.

[8] G. Di Battista and R. Tamassia. Algorithms for plane representations of acyclic digraphs. *Theoretical Computer Science*, 61(2-3):175–198, 1988.

[9] M. Jünger and S. Leipert. Level planar embedding in linear time. *Journal of Graph Algorithms and Applications*, 6(1):67–113, 2002.

[10] G. Kant, G. Liotta, R. Tamassia, and I. G. Tollis. Area requirement of visibility representations of trees. *Information Processing Letters*, 62(2):81–88, 1997.

[11] D. Kelly. Fundamentals of planar ordered sets. *Discrete Mathematics*, 63(2-3):197–216, 1987.

[12] D. G. Kirkpatrick and S. K. Wismath. Determining bar-representability for ordered weighted graphs. *Computational Geometry: Theory and Applications*, 6(2):99–122, 1996.

[13] H. Lü and J. Fogarty. Cascaded treemaps: examining the visibility and stability of structure in treemaps. In *ACM Proceedings of graphics interface 2008*, pages 259–266, 2008.

[14] F. Luccio, S. Mazzone, and C. K. Wong. A note on visibility graphs. *Discrete Mathematics*, 64(2-3):209–219, 1987.

[15] K. Romanik. Directed VR-representable graphs have unbounded dimension. In *GD '94*, volume 894 of *LNCS*, pages 177–181. Springer-Verlag, 1995.

[16] P. Rosenstiehl and R. E. Tarjan. Rectilinear planar layouts and bipolar orientations of planar graphs. *Discrete and Computational Geometry*, 1(1):343–353, 1986.

[17] B. Shneiderman. Tree visualization with tree-maps: 2-d space-filling approach. *ACM Transactions on Graphics*, 11(1):92–99, 1992.

[18] R. Tamassia and I. G. Tollis. A unified approach to visibility representations of planar graphs. *Discrete and Computational Geometry*, 1(1):321–341, 1986.

[19] R. Tamassia and I. G. Tollis. Tessellation representations of planar graphs. In *27th Allerton Conference on Communication, Control, and Computing*, pages 48–57, 1989.

[20] V. Tsiaras, S. Triantafilou, and I. G. Tollis. DAGmaps: Space filling visualization of directed acyclic graphs. *Journal of Graph Algorithms and Applications*, 13(3):319–347, 2009.

[21] J. Valdes, R. E. Tarjan, and E. L. Lawler. The recognition of series parallel digraphs. *SIAM Journal on Computing*, 11(2):298–313, 1982.

[22] S. K. Wismath. Characterizing bar line-of-sight graphs. In *SCG '85: Proceedings of the first annual symposium on computational geometry*, pages 147–152. ACM, 1985.