Representing Graphs by Knuth Trees

JOHN L. PFALTZ

University of Virginia, Charlottesville, Virginia

ABSTRACT. By means of the Knuth transform, arbitrary rooted trees may be represented compactly as binary trees. In this paper it is shown that the domain of this transform may be extended to a much wider class of graphs, while still maintaining its fundamental properties. Graphs, G, belonging to this extended domain are characterized first in terms of properties of an induced graph, G^*, and then in terms of local properties of G itself. A classic kind of "forbidden" subgraph theorem characterizes nonrepresentable graphs. Finally, it is shown that any directed graph can be modified to make it representable under the transform.

KEY WORDS AND PHRASES: data structure, directed graph, representation, transformation, tree

CR CATEGORIES: 3.73, 5.31

Donald Knuth in [2] presents a lovely transformation (which we will call the Knuth transform) by which an arbitrary rooted tree, T_M, may be represented as a *binary* rooted tree, T_R, as in Figure 1. It is important because by using such a transform, one can represent any rooted tree, with unbounded out-degree, by a linked data structure in which the cells that represent points of the data structure may be of fixed size. In fact, only two link fields are necessary to completely represent the tree structured relationship between the points. Given such a useful[1] transformation it is natural to ask, "Can we apply the Knuth transformation to a larger class of data structures?" A little experimentation is sufficient to convince us, at least intuitively, that in some cases it can be effectively applied (Figure 2), and in others it cannot (Figure 3). What distinguishes these two graphs?

First we must backtrack and define certain concepts with more precision. A *directed graph*, denoted $G = (P, E)$, is a set P of points (or data items) together with a binary relation, E, (or set of ordered pairs) defined on P. Graphs (we will henceforth drop the adjective "directed") are valuable as mathematical models of both abstract data structures and their computer representations. Thus we make a distinction between G_M, the graph which models an abstract data structure, and G_R, which models (or describes) its computer representation. In many cases the representation is virtually a one-to-one isomorphic copy of the abstract model and the distinction is unimportant. But at other times the distinction may be crucial. A representation is *faithful* if from G_R alone a computer procedure can reproduce all of the information that existed in the abstract model.

Let $x \in P$ be a point of a graph. By its set of *right neighbors*, $R(x)$ (often called the

This work was supported in part by the National Science Foundation under Grant NSF-GJ31626.

Author's address: Department of Applied Mathematics and Computer Science, University of Virginia, Charlottesville, VA 22901.

[1] Representation by fixed size cells with few link fields is of especial value for in-core representations. If external storage is employed just the reverse, large cells with many links, may be preferable. B-trees are one example [3].

set of sons or descendants), we mean $R(x) = \{y \mid (x, y) \in E\}$. Sets, such as $R(x)$, may be represented in a computer by many techniques. One of the simplest is as a linked list in which the explicit order that individual cells are linked in the list may, or may not, have any significance. If a point of the graph may belong to several sets of interest, then a computer representation of set membership may require one link field for each set to which the cell may belong. But if a point can belong to at most one set, then readily at most a single link field will be necessary.

The Knuth transformation $f : G_M \to G_R$ may now be informally described by the following: to each point x in G_M, associate a corresponding cell c_x in G_R. If $\{x, y, \cdots, z\} = R(w)$ for some w, link $c_x, c_y, \cdots, c_z$ as a set. Link each cell c_w to the set (list) representing $R(w)$. (Knuth calls these BROTHER and SON links.) In Figure 1 we show this transformation using dashed edges to denote set links and solid edges to denote those links that represent the functional relation $w \to R(w)$. While G_R may be regarded as a binary tree, it seems more accurate to regard it as a collection of sets together with a functional relation between elements w and their associated sets $R(w)$. Regardless of the significance attached to them, no more than two link fields per cell are needed, since in a rooted tree any point x can belong to at most one set $R(w)$. Now suppose G_M is neither rooted nor a tree, as in Figure 2. Representation under the Knuth transform still appears to "work," in that each cell needs at most two link fields. But the transformation "fails" on the graph G_M of Figure 3, even though it is a tree. Either c_f must contain an additional link field to distinguish its membership in the sets $\{e, f\}$ and $\{f, g\}$, or else e will "appear" (as shown) to be an element of $R(c)$. If the former representation is employed, then graphs, and even trees, may be found that require cells with an arbitrarily large number of "set" link fields, thereby vitiating the value of the transformation. The latter representation is not faithful.

While it is customary to say that the Knuth transformation, f, maps the set $\{T_M\}$ of all rooted trees[2] onto the set $\{T_R\}$ of binary rooted trees, this is not strictly true. T_R is not really a tree since it has two *distinct* kinds of "edges." It will pay to be more precise. A point set P with two, or more, binary relations $E_1, \cdots, E_n$ defined on it we will call a *hypergraph*, denoted by $G = (P, E_1, \cdots, E_n)$.[3] By a *path*, $\rho_{E_i}(x, y)$, we mean a sequence of points $\langle y_0, \cdots, y_n \rangle$, $n \leq 0$, such that $x = y_0$, $z = y_n$, and $(y_{j-1}, y_j) \in E_i$ for $1 \leq j \leq n$. Intuitively it is a sequence of points that can be accessed by traversing edges (possibly none) in E_i. Readily the existence of paths is a transitive relation.

Definition. A hypergraph $G' = (P', E'_1, E'_2)$ is said to be the *Knuth transform* of the graph $G = (P, E)$ if there exists 1-1 onto function $\varphi : P \to P'$ (we let x' denote $\varphi(x)$) such that

(i) $(x, y) \in E$ implies that for some unique $y_0 \in R(x)$, $(x', y_0') \in E_1'$ and $\rho_{E_2'}(y_0', y')$;

(ii) $(x', y_0') \in E_1'$ and $\rho_{E_2'}(y_0', y')$ implies $y = \varphi^{-1}(y') \in R(x)$;

(iii) E_1' and E_2' are partial functions.

One can verify that this definition includes the transform given by Knuth in [2]. Readily any representation of G_M by such a transform is faithful, since conditions (i) and (ii) imply that a computer search process which traverses the E_1'-link (if any) in the cell c_x to the cell c_{y_0} and then "follows" E_2'-links (if any) will retrieve *all* and *only* those cells corresponding to points in $R(x)$. Thus all edges (x, y) of the original relation E are reconstructable. Finally, since E_1' and E_2' are partial functions, for any $x' \in P'$ there exists at most a single $y' \in P'$ such that $(x', y') \in E'_1$, and a single z' such that $(x', z') \in E_2'$. Hence paths in E_2' from any cell c_x must be unique (thereby making the search procedure trivial), and no more than two link fields per cell will be needed in the representation of the hypergraph G'.

[2]As Knuth notes, this domain is actually the set of all "forests" of rooted trees.

[3]The term hypergraph has various other connotations; among them a point set, P, together with an n-ary relation E on P.

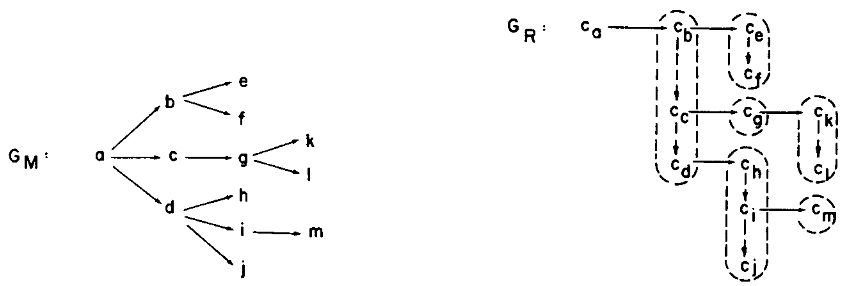

FIG. 1. The Knuth transformation, $f: G_M \to G_R$, applied to a rooted tree

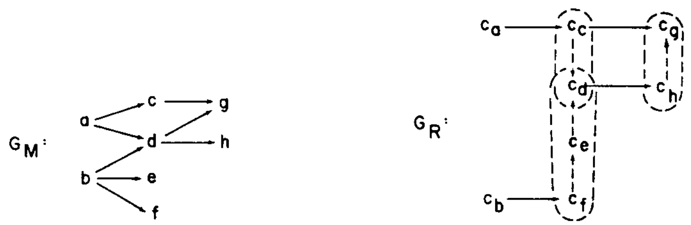

FIG. 2. The Knuth transform, $f: G_M \to G_R$, applied to an acyclic graph

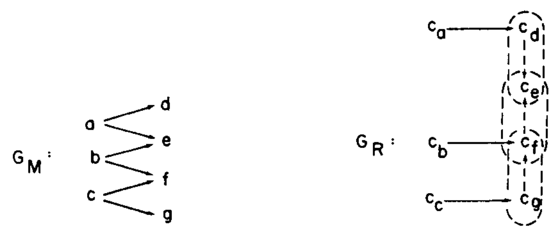

FIG. 3. An invalid application of the Knuth transform to a nonrooted tree

Let $G_M = (P, E)$ be any abstract data structure. G_M induces a second graph G^* defined as follows. Let $\mathcal{R}$ be the collection of sets $\{R(x) \mid x \in P\}$. Augment $\mathcal{R}$ with all finite intersections of elements of $\mathcal{R}$ (in the same manner that one forms a topological base from a subbase) to form $\mathcal{R}^*$. The set $\mathcal{R}^*$ is partially ordered by set inclusion to form $G^* = (\mathcal{R}^*, \subseteq)$. Since the empty set $\varnothing \subseteq R(x)$ for all x, $\varnothing$ is a least element (or root) of G^*, implying G^* is connected, whether or not G_M was.

THEOREM. *The Knuth transform, $f: G_M \to G_R$, of a finite graph $G_M = (P, E)$ exists if and only if $G^* = (\mathcal{R}^*, \subseteq)$ is a tree.*

PROOF. We first prove necessity. Assume G^* is not a tree. Since G^* has a least element, $\varnothing$, there exists R_1, R_2, and $R_3 \in R^*$ such that $R_1 \subset R_3$, $R_2 \subset R_3$, and $R_1 \neq R_1 \cap R_2 \neq R_2$ (see [4]). Let $a \in R_1 \sim R_2$, and let $b \in R_2 \sim R_1$. By construction of $\mathcal{R}^*$, there exist x, $y \in G$ such that $a \in R(x) \sim R(y)$ and $b \in R(y) \sim R(x)$. Now suppose G' were a Knuth transform of G_M. By property (i) there exist a_0' and b_0' in G' such that $(x', a_0') \in E_1'$ and $\rho_{E_2'}(a_0', a')$; $(y', b_0') \in E_1'$ and $\rho_{E_2'}(b_0', b')$. Further, $a, b \in R_3$ implies for some $z \neq x, y$ that $a, b \in R(z)$. Thus there exists c_0' such that

$$(z', c_0') \in E_1', \ \rho_{E_2'}(c_0', a'), \text{ and } \rho_{E_2'}(c_0', b').$$

Since by property (iii) E_2' is a function (it can be regarded as a successor function with initial element c_0'), we must have either $\rho_{E_2'}(a', b')$ or $\rho_{E_2'}(b', a')$. If the former case holds, then by transitivity of the path relation $\rho_{E_2'}(a_0', a')$ and $\rho_{E_2'}(a', b')$ implies $\rho_{E_2'}(a_0', b')$. This together with $(x', a_0') \in E_1'$ and $b \notin R(x)$ yields a contradiction to property (ii). If the latter case holds, we get a similar contradiction. Hence there exists no such Knuth transform, and the condition of the theorem is necessary.

Conversely, assume $\mathcal{R}^*$ is a tree. We demonstrate existence of a Knuth transform by construction. For $y \in P$, let y' denote its image under $\varphi : P \to P'$. Define the relations E_1' and E_2' as follows.

Case 1. If $y \notin R(x)$ for any $x \in P$, then $(x', y') \notin E_1'$ and $(y', z') \notin E_2'$ for all $x', z', \in P'$.

Case 2. $y \in R(x_i)$ for some, possibly several, $x_i \in P$. Consider $Z = \bigcap\{R(x_i) \mid y \in R(x_i)\}$. $Z \in \mathcal{R}^*$. Since $\mathcal{R}^*$ is a tree, there exists a *unique* $W \in \mathcal{R}^*$ such that $\varnothing \subseteq W \subset Z$. Let $Y = Z \sim W$. By construction, $y \in Y$. Arbitrarily order the elements of Y, say $\{y_0, \cdots, y_r\}$. We may assume that the elements of W have been ordered $\{w_0, \cdots, w_s\}$. Let $(y_j', y_{j+1}') \in E_2'$ for $0 \leq j \leq r-1$; $(y_r', w_0') \in E_2'$ iff $W \neq \varnothing$; and $(x_i', Y_0') \in E_1'$ iff $Z = R(x_i)$.

It is apparent that E_1' and E_2' so defined are partial functions. Let $(x, y) \in E$, so $y \in Y$. If $Y = R(x)$, then $(x', y_0') \in E_1'$ and $\rho_{E_2'}(y_0', y')$, thereby satisfying property (i). If $Y \neq R(x)$, then there exists a unique sequence (since $\mathcal{R}^*$ is a tree) $Y \subset Y_1 \subset \cdots \subset Y_n = R(x)$. The sequence $(x', y_{0,n}') \in E'_1$, $\rho_{E_2'}(y_{0,n}', y_{0,n-1}') \cdots \rho_{E_2'}(y_{0,1}', y_0')$, and $\rho_{E_2'}(y_0', y')$ then satisfies property (i) by transitivity of ρ. Property (ii) is similarly verified. □

Figures 4 and 5 illustrate this theorem by presenting the induced graphs G^* of the graphs of Figures 2 and 3, respectively. In the first case G_2^* is a tree and G_2 is representable under a Knuth transform; in the latter case it is not.

The following corollaries give a "local" criterion for Knuth representability in terms of G_M, not G^*.

COROLLARY 1. *A Knuth transform of* $G = (P, E)$ *exists if and only if for all* $x, y, z \in P$, $R(x) \cap R(y)$ *and* $R(x) \cap R(z)$ *nonempty implies either* $R(x) \cap R(y) \subseteq R(x) \cap R(z)$, *or* $R(x) \cap R(z) \subseteq R(x) \cap R(y)$.

PROOF. It is easily verified that, since G^* has a least element, G^* is a tree if and only if this condition holds. □

COROLLARY 2. *If* G_M *is a rooted tree (or forest of rooted trees), then the Knuth transformation of* G_M *exists.*

PROOF. The proof follows immediately from the observation that in G_M, $R(x) \neq R(y)$ implies $R(x) \cap R(y) = \varnothing$. □

$$R_2^* = \begin{Bmatrix} R(a) = \{c, d\}, \ R(b)\ \{d, e, f\}, \ R(c) = \{g, h\}, \ R(d) = \{h\}, \\ R(e) = R(f) = R(g) = R(h) = \phi, \ X = R(a) \cap R(b) = \{d\} \end{Bmatrix}$$

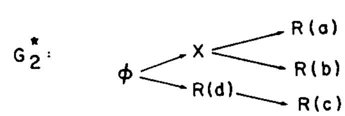

FIG. 4. G_2^* induced by the graph G_M of Figure 2

$$R_3^* = \begin{Bmatrix} R(a) = \{d, e\}, \ R(b) = \{e, f\}, \ R(c) = \{f, g\}, \\ R(d) = R(e) = R(f) = R(g) = \phi, \ X = R(a) \cap R(b) = \{e\}, \\ Y = R(b) \cap R(c) = f \end{Bmatrix}$$

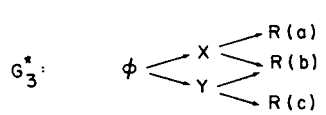

FIG. 5. G_3^* induced by the graph G_M of Figure 3

As noted earlier, in a linked representation of a set, such as $R(x)$, the actual order of linking may be arbitrary. The construction employed in the theorem obviously takes advantage of this freedom to order the links in such a way that subsets become "terminal segments" of the ordering on the entire set. The condition that $\mathfrak{R}^*$ be a tree insures that one can establish such an ordering consistent with the inclusion properties of the sets $R(x)$ in G_M. It is precisely this device which permits us to extend the domain of the transformation.

But in the context of some applications, one may wish to impose some other ordering on the elements of $R(x)$. For example, if G_M denotes the parse tree of some string, then it is natural to impose a "left-to-right" ordering on $R(x)$ corresponding to the position of the terminal, or nonterminal, symbol within the string. In such a context the ordering within the representation must reflect the ordering of the abstract model, and in consequence the domain of the Knuth transform is greatly reduced. The proof of Corollary 2 can be used to show that all rooted trees still belong to this restricted domain. But the reader may verify that the condition, $R(x) \neq R(y)$ implies $R(x) \cap R(y) = \varnothing$, also holds in the case of two-terminal parallel series networks, as well as a number of unclassifiable graphs such as those shown in Figure 6. Thus, these, too, are Knuth representable, even under these more restricted ordering conditions. Notice also that G_M need not be acyclic, although most of our examples have been.

Corollary 1 may be restated in the form of a "forbidden subgraph" theorem of similar vein to the Kuratowski theorem [1] which characterizes those graphs admitting a planar representation. *A graph G_M is Knuth representable if and only if it does not contain the graph of Figure 7 as a subgraph.* But there is an important difference. In most such characterizations the forbidden subgraph may not exist as a homeomorph. In this case, be-

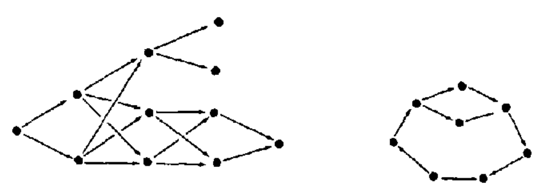

FIG. 6. Two Knuth representable graphs, G_M even, assuming an externally imposed order on the sets $\{R(x)\}$

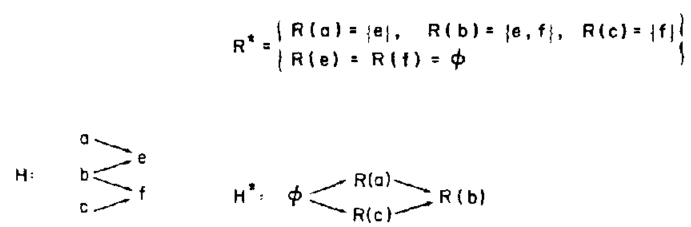

FIG. 7. The forbidden subgraph

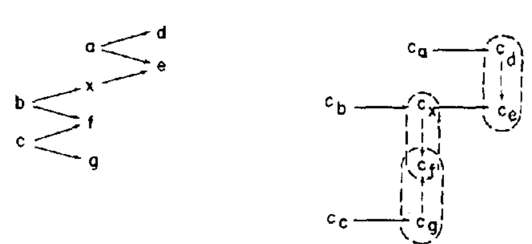

FIG. 8. The forbidden subgraph of Figure 3 "broken" by the addition of one additional point, x

cause of the "local" nature of Corollary 1, the literal subgraph itself must not exist. The graph of Figure 3 contains this forbidden subgraph, but by adding the single point "x" to the graph, one gets the graph of Figure 8. These two distinct graphs are homeomorphs, but the latter is Knuth representable! This observation provides an effective method of extending the technique of Knuth representation to include *all* directed graphs. Forbidden subgraphs in G_M may be "broken" by simply adding dummy points. The corresponding cells in the representation G_R are then tagged and the routines that operate on G_R may be modified in a straightforward way to treat these extra points as if they were "invisible." Thus at the cost of a few extra cells and slight additional complexity to the accessing routines, the ability to represent any graph with a fixed number of link fields per cell is obtained.

Finally, we note that in this discussion it has been assumed that, gi ven $x \in P$, one wants to access the set of all right neighbors, $\{y_i\}$, such that $(x, y_i) \in E$. But there exist situations in which one wants to be able to "work backwards" through the data structure, that is, given $y \in P$, to find all left neighbors, $L(y) = \{x_j\}$, such that $(x_j, y) \in E$. A lovely result, also due to Knuth [2], shows that if G_M is a rooted tree, then by appropriate tagging this can be accomplished with no additional link fields. If G_M is not a rooted tree, some other approach must be used, possibly threading the cells with respect to a reverse topological sort. Another alternative is to employ a symmetric representation of the sets $L(y)$ and the functional relation $y \rightarrow L(y)$ as has been already developed. A complete symmetric representation of the data structure G_M can then be implemented using cells with no more than four link fields to fully represent the relation E.

REFERENCES

(Note. References [3] and [4] are not cited in the text.)

1. HARARY, F. *Graph Theory.* Addison-Wesley, Reading, Mass., 1969.
2. KNUTH, D. *The Art of Computer Programming, Vol. 1: Fundamental Algorithms.* Addison-Wesley, Reading, Mass., 1968.
3. KNUTH, D. *The Art of Computer Programming, Vol. 3: Sorting and Searching.* Addison-Wesley, Reading, Mass., 1973.
4. PFALTZ, J., FINDLER, N., AND BERNSTEIN, H. *Four High Level Extensions of Fortran IV.* Spartan Books, Washington, D.C., 1972.

RECEIVED JULY 1974