Discrete Applied Mathematics **36** (1992): 87–92

Narrowness, Path-width, and their Application in Natural Language Processing

András Kornai

Department of Linguistics, Stanford University, Stanford, CA 94305, USA;
and
Institute of Linguistics, Hungarian Academy of Sciences, Budapest, Hungary

Zsolt Tuza

Computer and Automation Institute, Hungarian Academy of Sciences
H-1111 Budapest, Kende u. 13-17, Hungary

Abstract

In the syntactic theory of Tesnière (1959) the structural description of sentences are given as graphs. We discuss how the graph-theoretic concept of path-width is relevant in this approach. In particular, we point out the importance of graphs with path-width ≤ 6 in connection with natural language processing, and give a short proof of the characterization theorem of trees with path-width k.

1 The linguistic background

Following the pioneering work of Tesnière [Te], the field of *dependency grammar* evolved at a steady pace. For results and references, see [Ma] and [Me]. In the present note we concentrate on one particular dependency model, put forth by Kálmán and Kornai [KK], although our observations are applicable for a wider range of dependency formalisms where the syntactic description of a sentence is given as an ordered graph (with vertices corresponding to *words* and arcs corresponding to *dependencies*).

In this model a grammatical derivation starts with a *dependency graph* which encodes the major syntactic relations that can obtain among words (e.g. attribution, modification, or possession) by labelled arcs that run between the vertices (words). A grammatical derivation of a sentence begins with a *dependency graph* and ends with a linear sequence of nodes, corresponding to the temporal order in which the words of the sentence are uttered or written.

The essential feature of the model is a relatively small storage unit called the *shack,* which differs from ordinary stack (LIFO) memory in several respects. First, the shack is finite – it is assumed that it can hold at most 6 or 7 vertices at any given moment. Second, the shack is unordered (random access). Third, elementary memory cells of the shack are indistinguishable – this means that the shack cannot store two or more copies of the same element. A fourth property, not considered in this paper, is the "bladder-like" nature of the shack. In modeling the production of actual sentences, the shack hardly ever contains more than 4 items, and research on human sentence production (Yngve [Y], Church [Ch]) suggests that in a realistic model overloading the shack results in the loss of the *entire* memory content, rather than in the loss of the last element. This property is successfully captured in connectionist symbol manipulation models such as Touretzky and Hinton [TH].

During the derivation the graph is moved from a permanent storage space – which we will call the *inner memory (IM)* – to the *outer memory (OM)* via the *shack*. The order in which items are moved from IM to the shack – called *in-sequence* – or from the shack to OM – called *out-sequence* – is arbitrary, but the following condition must be met:

STMC (Short Term Memory Constraint) – A vertex can be moved from the shack to OM only if all of the vertices connected to it by an arc are also in the shack or already in OM.

The STMC captures the idea that the structural relations obtaining between those parts of the sentence which are already spoken and those which are not must be ke pt in the short-term memory of the speaker (see Fodor et al. [FBG]). Similarly, in order to understand the full content of the sentence, the listener has to remember (i.e. store in the short-term memory) all words having dependencies to the unspoken part. The bounded capacity of this short-term memory is a special case of the general phenomenon known in psychology as "Miller's Law" [Mi].

2 Narrowness of graphs

We use standard terminology and notation of graph theory. Graphs considered here are undirected, with no loops or multiple edges.

Let us note first that every sequence $(v_1, \dots, v_n)$ of the vertices of a graph $G = (V, E)$, when viewed as an in-sequence, together with the STMC defines a minimum demand on shack capacity in the following way. In the i-th step, put vertex v_i from the IM into the shack; move all $v_j (j \le i)$ with no neighbors $v_k, k > i$, from the shack to the OM; then go to the $(i+1)$-st step. The maximum number of vertices in the shack during this process gives a lower bound for the capacity of the shack needed for that particular in-sequence. This maximum, denoted by $\nu(v_1, \dots, v_n)$, will be called the *narrowness* of the in-seque nce in question.

Definition. [TK] The *narrowness* $\nu(G)$ of a graph $G = (V, E)$ is the minimum value of $\nu(v_1, \dots, v_n)$ taken over all permutations $(v_1, \dots, v_n)$ of the vertex set V.

A "dual" version of the definition might be introduced starting with an *out*-sequence $(w_1, \dots, w_n)$ as a permutation on V. For each $v \in V$ there is a smallest subscript i such that $v = w_i$ or v is adjacent to w_i. Putting v into the shack just before w_i is moved to the OM, the largest number of vertices in the shack during this process – called the narrowness of the out-sequence in question – gives a lower bound on the shack capacity needed. The next observation shows, however, that this latter approach leads to the same definition of narrowness.

Proposition 2.1. For any graph G, there is an in-sequence of narrowness k if and only if there is an out-sequence of narrowness k. **Proof.** Observe that the in-sequence $(v_1, \ldots, v_n)$ and the out-sequence $(w_1, \ldots, w_n)$ satisfying $v_i = w_n + 1 - i$ have the same narrowness. □

3 Related invariants

In this section we point out that narrowness has equivalent interpretations in terms of several graph invariants of a different nature, introduced in the literature. Recall from [RS] that the *path-width* $\pi(G)$ of a graph $G = (V, E)$ is the smallest value $min_X max_{1 \le i \le |X|} |X_i| - 1$, where the minimum is taken over all "path decompositions" of G. (A *path decomposition* is a set system $X = \{X_1, \ldots, X_t\}$ with the following properties: $\bigcup_{1 \le i \le t} X_i = V$, for every edge $xy \in E$ there is an i with $X_i \supset \{x, y\}$, and for all $i < j < k, X_j \supset X_i \cap X_k$. The close relation between narrowness and path-width is expressed in the following statement.

Proposition 3.1 For every graph G with at least one vertex, $\nu(G) = \pi(G) + 1$. **Proof.** To show $\nu(G) \le \pi(G) + 1$, let $\{X_1, \ldots, X_t\}$ be a path decomposition of width $\pi(G)$ in G. Putting $X_i \setminus X_{i-1}$ into the shack and then $X_i \setminus X_{i+1}$ from the shack to the OM for $1 \le i \le t (X_0 = X_{t+1} = \emptyset)$, the contents of the shack during this process always are subsets of some X_i, so that no more than $\pi(G) + 1$ positions are required in the shack.

To prove $\pi(G) + 1 \le \nu(G)$, let $(v_1, \ldots, v_n)$ be an in-sequence of narrowness $\nu(G)$. Define X_i as the set of vertices in the shack at the moment when v_i has just been put there from the IM $(1 \le i \le n)$. Since each edge appears in the shack in some step, a path decomposition with $max_{1 \le i \le n} |X_i| \le \nu(G)$ is obtained. □

The problem of determining $\nu(G)$ is equivalent or closely related to many others as well, including the "gate matrix layout problem" [O+], vertex and edge separators [L], search number [P], interval thickness [KF], node search number [KP2], and minimum demands in some types of "pebble games", too. Some of those studies were motivated by practical problems in VLSI design and other important applications; for further references, see the recent survey [Mo2] and also [KP1].

4 Trees

So far the only general class of graphs for which narrowness is well-characterized is the class of trees (and forests). The importance of the result to be discussed in this section is demonstrated by the fact that its various equivalents and consequences were discovered by many authors independently [Di,EST,Sch,TUK,TK]. Let us note that [TK] was one of the earliest presentations of the theorem, and that our proof is one of the shortest.

For a vertex v, an edge e, and a subgraph F of a graph G denote by $G-v, G-e, G/e$, and $G-F$ the graph obtained by deleting v, deleting e, contracting e, and deleting the vertices of F, respectively. Define the classes $\mathbf{T}_k$ of trees ($k \geq 2$) recursively as follows. Let $\mathbf{T}_2 = \{K_2\}$ (i.e. just one graph with two vertices and one edge), and let $T \in \mathbf{T}_{k+1}$ if and only if T has a vertex v of degree 3 such that each of the three connected components of $T-v$ belongs to $\mathbf{T}_k$.

Theorem 4.1. A tree T has $\nu(T) \geq k (k \geq 2)$ if and only if T is contractible to some $T* \in \mathbf{T}_k$. In particular, edge contraction does not increase $\nu(T)$, and every tree T satisfying $\nu(T/e) < \nu(T)$ for all edges e has precisely $(5 \cdot 3^{\nu(T)-2} - 1)/2$ vertices.

The crucial notion in our proof (closely related in flavor to the one of [Sch]) is the concept of *p-strong edges.* We say that an edge e is *p-strong* in a tree T if both connected components of $T-e$ have narrowness $\geq p$. For $p \geq 1$, $T(p)$ denotes the subgraph formed by the p-strong edges of T (hence $T(1) = T$).

Lemma 4.2. If $T(p)$ is non-empty, then it is connected, and each component T' of $T - T(p)$ has $\nu(T') < p$.

Proof. Any two non-adjacent edges e', e'' of $T(p)$ are joined by a path P in T. For any edge $e \in P$, each component C of $T-e$ entirely contains a component of $T-e'$ or $T-e''$, implying $\nu(C) \geq p$ since e' and e'' both are p-strong. Thus, $T(p) \supset P$ and consequently $T(p)$ is connected. For $e \notin T(p)$, one of the two components of $T-e$ contains $T(p)$, so that the other must have narrowness $< p$ for e is assumed not to be p-strong. □

Proof of Theorem 4.1. We apply induction on narrowness, and on the number of vertices, too. For $k=2$ the statements are obvious, so we assume $\nu(T) = k \geq 3$. Let p be the largest integer for which $T(p)$ is non-empty. If $T(p)$ is a path, say of vertices $x_1, \ldots, x_t$, then we claim $\nu(T) = p$ holds. Indeed, an in-sequence of narrowness p is obtained as follows. Put x_1 into the shack first. If x_i is in the shack, but x_{i+1} isn't, then take in-sequences of narrowness at most $p-1$ –by Lemma 4.2– for those components of $T - T(p)$ one by one which are joined to x_i (at any step, all vertices in the shack, other than x_i, belong to the same component). Then put x_{i+1} into the shack and move x_i to the OM.

If $T(p)$ is not a path, then $\nu(T) = p+1$ holds. To prove $\nu(T) \leq p+1$, let $e \in T(p)$ be an edge such that a component $T[e]$ of $T-e$ has $\nu(T[e]) > p$, and $T[e]$ is minimal under inclusion. (The other component has $\nu \leq p$ by the choice of p. Moreover, if such an e does not exist, then the inequality $\nu(T) = p+1$ is obvious.) By the minimality of $T[e]$, $\nu(T') \leq p$ holds for every component T' of $T-v$ for $v := e \cap T[e]$. Hence, an in-sequence of narrowness $\leq p+1$ can be found, starting with v, and taking those components one by one.

To prove $\nu(T) \geq p+1$, let $v \in T(p)$ be a vertex of degree ≥ 3, $T_i (i = 1, 2, 3)$

three components of narrowness p in $T - v$, and $(v_1, \ldots, v_n)$ an in-sequence with $\nu(v_1, \ldots, v_n) = \nu(T)$. There are subscripts $j(i)$ such that the shack contains at least p vertices of T_i when $v_{j(i)}$ appears in the shack. In addition, assuming $j(1) < j(2) < j(3)$, the shack contains at least one vertex of $T'' := \{T_1\} \cup \{T_3\} \cup \{v\}$ when $v_{j(2)}$ is moved into the shack, since T'' is connected and $v_{j(2)}$ separates $v_{j(1)}$ from $v_{j(3)}$ in the in-sequence. Thus, $\nu(T) \geq p + 1$ follows.

If $T(p)$ is not a path, i.e. $p = k-1$, then the three components T_i are contractable to some members of $\mathbf{T}_{k-1}$, and those parts with v provide a $T* \in \mathbf{T}_k$. Otherwise, if $T(p) \neq \emptyset$ is a path, $p = k$, then contract $T(p)$ to just one vertex. Since the graph obtained still has narrowness k, (for it contains a component of $T-e$ as a subgraph, where e is an endedge of $T(p)$, the existence of a $T*$ follows by induction on the number of vertices.

If contraction could increase $\nu(T)$, say $k = \nu(T) < \nu(T/e)$ held for some edge e, then the contracted tree T/e of T would be contractible to a member of $\mathbf{T}_{k+1}$, implying the contradiction $\nu(T) > k$. Finally, the recursion $f(k) = 3f(k-1) + 1$ with $f(2) = 2$ implies that the members of $\mathbf{T}_k$ have precisely $(5 \cdot 3^{k-2} - 1)/2$ vertices. $\square$

In the previous proof we tacitly applied the obvious fact that vertex (and edge) deletion does not increase $\nu(G)$. We note that the same property holds for edge contractions as well – not only for trees, proved above, but also for every graph, see [Tu]. According to the facts described in Section 1, for applications in linguistics it would be of definite interest to know the structure of graphs with narrowness ≤ 7 (i.e. with path-width ≤ 6). Further open problems related to narrowness and path-width are raised in [Tu].

References

[Ch] K. Church, On memory limitations in natural language processing, Technical Report TR-245, MIT Laboratory for Computer Science (1980)

[DI] P. Duchet and G. Ikorong, La largeur d'un arbre, lecture at 4'eme Colloque International en Théorie des Graphes et Combinatoire, Marseille-Luminy, F rance (1990)

[EST] J.A. Ellis, I.H. Sudborough, and J.S. Turner, Graph separation and search number, preprint, Northwestern Univ. and Washington Univ. (1984); extended version to appear.

[FBG] J. Fodor, Th. Bever, and M.F. Garrett, *The Psychology of Language,* McGraw-Hill, New York (1974)

[KF] T. Kashiwabara and T. Fujisawa, NP-completeness of the problem of finding a minimum-clique-number interval graph containing a given graph as a subgraph, in: *Proc. ISCAS* (1979) 657-660.

[KK] L. Kálmán and A. Kornai, Pattern matching, manuscript, Institute of Linguistics, Hungarian Academy of Sciences (1984), and lecture at the University of Vienna Syntax Workshop (1985)

[KP1] L.M. Kirousis and Ch.H. Papadimitriou, Searching and pebbling, *Theoret. Computer Sci.* **47** (2) (1986) 205-218.

[KP2] L.M. Kirousis and Ch.H. Papadimitriou, Interval graphs and searching, *Discrete Math.* **55** (2) (1985) 181-184.

[L] T. Lengauer, Black-white pebbles and graph separation, *Acta Informatica* **16** (1981) 465-475.

[MS] F. Makedon and I.H. Sudborough, Minimizing width in linear layouts, in: *Proc. 10th ICALP*, Lecture Notes in Computer Science **154**, Springer-Verlag (1983) 478-490.

[Ma] P.H. Matthews, *Syntax*, Cambridge University Press (1981)

[Me] I.A. Mel'cuk, *Dependency Syntax: Theory and Practice*, State University Press of New York, Albany (1987)

[Mi] G.A. Miller, The magical number seven, plus or minus two: some limits on ou r capacity for processing information, *Psych. Rev.* **63** (1956) 81-97.

[Mo2] R.H. Mo2hring, Graph problems related to gate matrix layout and PLA folding, in: G. Tinhofer et al., eds., *Computational Graph Theory, Computing Suppl.* **7**, Springer-Verlag, Wien (1990) 17-51.

[O+] T. Ohtsuki, H. Mori, E.S. Kuh, T. Kashiwabara, and T. Fujisawa, One-dimensional logic gate assignment and interval graphs, *IEEE Trans. Circuits Syst.* **CAS-26** (1979) 675-684.

[P] T.D. Parsons, The search number of a connected graph, in: *Proc. 9th Southeastern Conf. on Combinatorics, Graph Theory, and Computing*, Utilitas Math. Publ., Winnipeg (1978) 549-554.

[R] P.A. Reich, The finiteness of natural language, *Language* **45** (1969) 831-843.

[RS] N. Robertson and P.D. Seymour, Graph minors. I. Excluding a forest, *J. Combinatorial Theory Ser. B* **35** (1) (1983) 39-61.

[Sch] P. Scheffler, A linear algorithm for the pathwidth of trees, in: R. Bodendiek and R. Henn, eds., *Topics in Combinatorics and Graph Theory*, Physica-Verlag, Heidelberg (1990) 613-620.

[TUK] A. Takashini, Sh. Ueno, and Y. Kajitani, A characterization of the cycle-free k-path in terms of forbidden minors, lecture at 2nd Japan Conference on Graph Theory and Combinatorics, Hakone, Japan (1990)

[Te] L. Tesnière, *Eléments de Syntaxe Structurale,* Klincksieck, Paris (1959)

[TH] D.S. Touretzky and G.E. Hinton, Symbols among the neurons: details of a connectionist inference architecture, in: *Proc. 9th Intern. Joint Conf. on Artificial Intelligence* (1985) 238-243.

[Tu] Zs. Tuza, Problems related to path-width and narrowness, manuscript (1990)

[TK] Zs. Tuza and A. Kornai, Narrow graphs in natural language processing, lecture at Csillag Pál Mathematical Colloquium, Balatonlelle, Hungary (1984)

[Y] V. Yngve, The depth hypothesis, in: R. Jakobson, ed., *The Structure of Language and its Mathematical Aspects*, American Mathematical Society, Providence RI (1961) 130-138.