Implementation of PQ-tree Algorithms

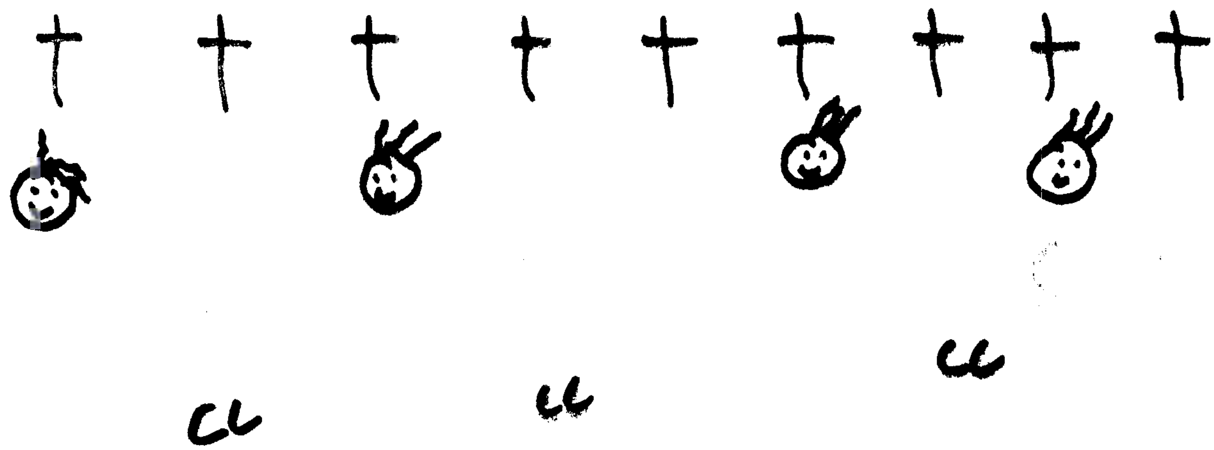
M. J. Fischer

R. E. Ladner

S. M. Young

Applications:

1. Relative dating



2. data organization

3. Testing for interval graphs

4. Testing for graph planarity

PQ-trees

Kellogg Booth
George Lueker

Contiguous Ordering problem

Given $S_1, S_2, \ldots, S_k \subseteq U$ find a linear ordering of U (if there is one) such that within the ordering each set S_i is consecutive.

Example: $U = \{1, 2, \ldots, 6\}$

$S_1 = \{1, 5, 6\}$

$S_2 = \{2\ 5\ 6\}$

$S_3 = \{1, 5\}$

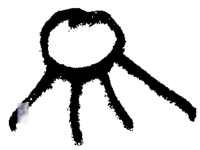
$S_4 = \{3, 4\}$

3 4 2 6 5 1

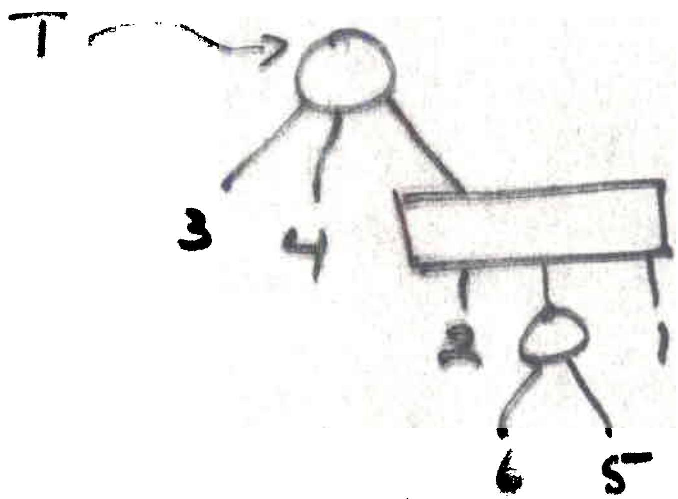
There are 7 more.

PQ-tree

leaves — distinctly labeled

P-nodes  at least 2 children

Q-nodes  at least 3 children



$T = \langle 3\ 4\ [2\ \langle 6\ 5 \rangle\ 1] \rangle$

$f(T) = 342651$ frontier of T

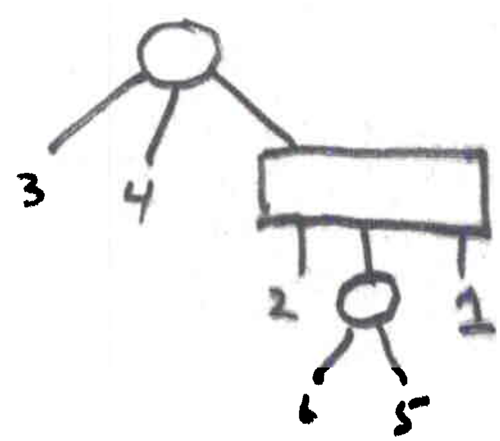
$\ell(T) = \{1, 2, 3, 4, 5, 6\}$ leaves of T

Allowable transformations

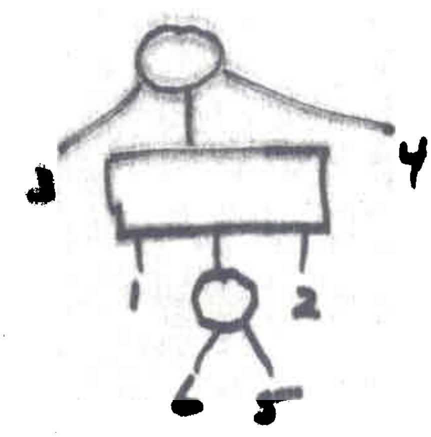
1. Permute the children of a P-node.
2. Reverse the children of a Q-node.

Equivalent trees

$T \equiv T'$ if T' can be reached from T by a sequence of allowable transformations



3 4 2 6 5 1



3 1 6 5 2 4

$$\delta(T) = \{ \rho(T') : T' \equiv T \}$$

permutations of $\ell(T)$ stored by T

T a PQ-tree $S \subseteq \ell(T)$

T is <u>reducible</u> from S if

$$\delta(T) \cap \delta(\langle\langle a_1 \cdots a_i \rangle a_{i+1} \cdots a_n \rangle)$$

is non-empty where

$S = \{a_1, \dots, a_i\}$.

---

we want a tree $A(T,S)$ s.t.

$$\delta(A(T,S)) = \delta(T) \cap \delta(\langle\langle a_1 \cdots a_i \rangle a_{i+1} \cdots a_n \rangle)$$

<u>The algorithm</u>

$T \leftarrow \langle u_1 \cdots u_m \rangle$, $U = \{u_1 \cdots u_m\}$

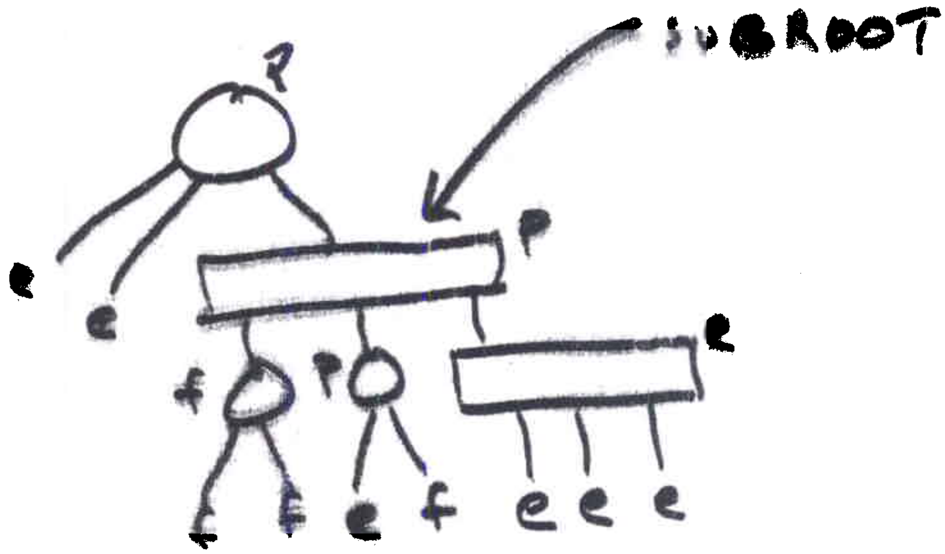
<u>for</u> $i = 1$ <u>to</u> k <u>do</u> $T \leftarrow A(T, S_i)$.

FULL: All descendant leaves are in S

EMPTY: All descendant leaves are in $\overline{S}$

PARTIAL: Neither full nor empty

SUBROOT: root of smallest subtree containing all full nodes



Marked Tree relative to S

The Algorithm

1. Mark the full nodes, bottom up.
2. Mark the partial nodes, bottom up. Do not go to root unless necessary.
3. Find subroot. (If tree not reducible then it is discovered by this time)

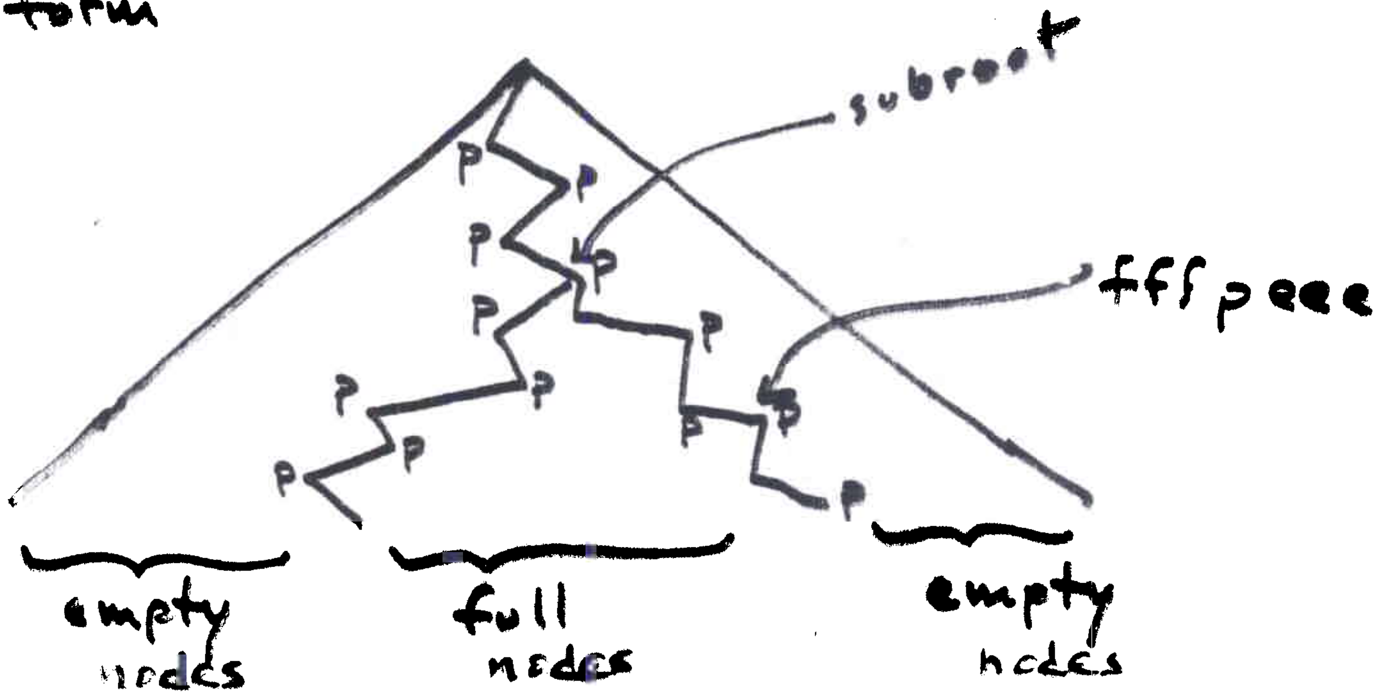
* 4. Modify tree rooted at SUBROOT.

FOREST: $T_1 \cdots T_n$

T_i PQ tree

$\ell(T_i) \cap \ell(T_j) = \phi$ if $i \neq j$

FACT: T is reducible from S iff the marked tree relative to S is equivalent to a tree of the form



$$\text{subroot} - \left\{ \begin{array}{l} \langle p_1 \, \underline{f} \, p_2 \, \underline{e} \rangle \\ [\underline{e}_1 \, p_1 \, \underline{f} \, p_2 \, \underline{e}_2] \end{array} \right\}$$

$$\text{partial nodes} - \left\{ \begin{array}{l} \langle \underline{f} \, p \, \underline{e} \rangle \\ [\underline{f} \, p \, \underline{e}] \end{array} \right\}$$

$\underline{f} \in \mathrm{FULL}^*$

$\underline{e}, \underline{e}_1, \underline{e}_2 \in \mathrm{EMPTY}^*$

$p, p_1, p_2 \in \mathrm{PARTIAL} \cup \{\lambda\}$

P- and Q-node constructors $\langle\, \rangle$, [

$$\langle T_1 \ldots T_n \rangle = \begin{cases} \langle T_1 \ldots T_n \rangle & n \geq 2 \\ T_1 & n = 1 \\ \lambda & n = 0 \end{cases}$$

$$[\, T_1 \ldots T_n \,] = \begin{cases} [\, T_1 \ldots T_n \,] & n \geq 3 \\ \langle T_1\, T_2 \rangle & n = 2 \\ T_1 & n = 1 \\ \lambda & n = 0 \end{cases}$$

$$T_1 \ldots T_n{}^R = T_n \ldots T_1$$

$$M : \text{SUBROOT} \to \text{TREE}$$

$$M(\langle p_1 \underline{f}\, p_2\, \underline{e} \rangle) := \langle [D(p_1)^R \langle \underline{f} \rangle D(p_2)]\, \underline{e} \rangle$$

$$M([\underline{e}_1\, p_1\, \underline{f}\, p_2\, \underline{e}_2]) := [\underline{e}_1\, D(p_1)^R\, \underline{f}\, D(p_2)\, \underline{e}_2]$$

$$M(f) := f$$

$$\delta(M(T)) = \delta(T) \cap \delta(\langle\langle \ell(T) \cap S \rangle\, \ell(T) - S \rangle)$$

$$D : \text{PARTIAL} \to \text{FOREST}$$

$$D(\langle \underline{f}\, p\, \underline{e} \rangle) := \langle \underline{f} \rangle D(p) \langle \underline{e} \rangle$$

$$D([\underline{f}\, p\, \underline{e}]) := \underline{f}\, D(p)\, \underline{e}$$

$$D(\lambda) := \lambda$$

$$\delta(D(T)) = \delta(T) \cap \delta(\langle\langle \ell(T) \cap S \rangle \langle \ell(T) - S \rangle$$

Linear Time $S_1, \dots, S_k \subseteq U$

$m = |U|$

$k = \#$ of subsets

$s = \sum_{i=1}^{k} |S_k|$

$O(m + k + s)$

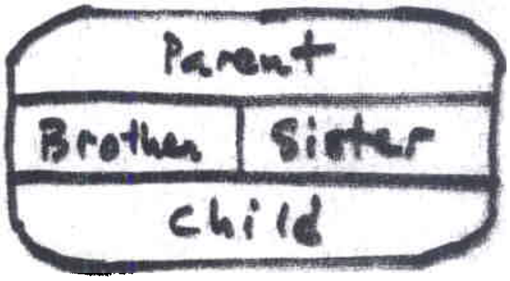
$m \approx k \sim 800$

$s \sim 20{,}000$

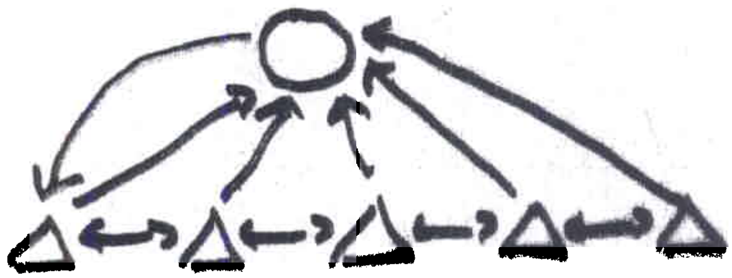
8 secs

Data Representation

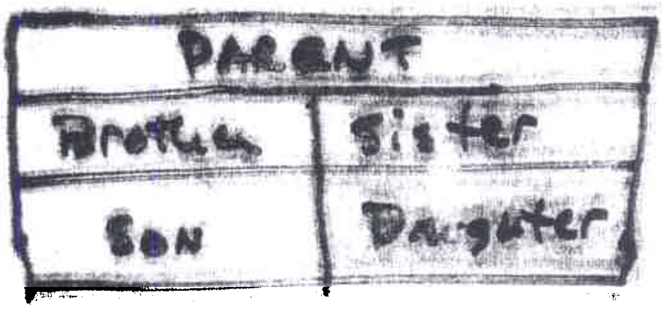
PNODE



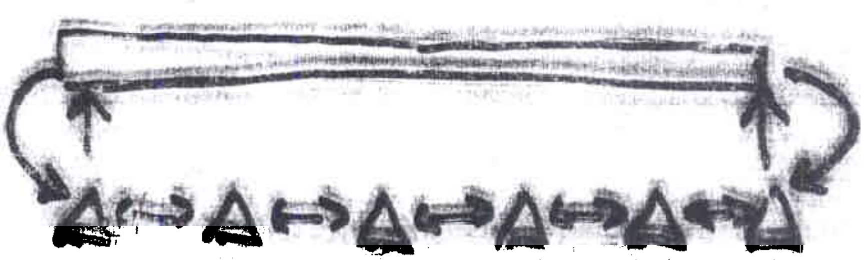
LISTPLACE
PARTIAL
group
partial list
full list



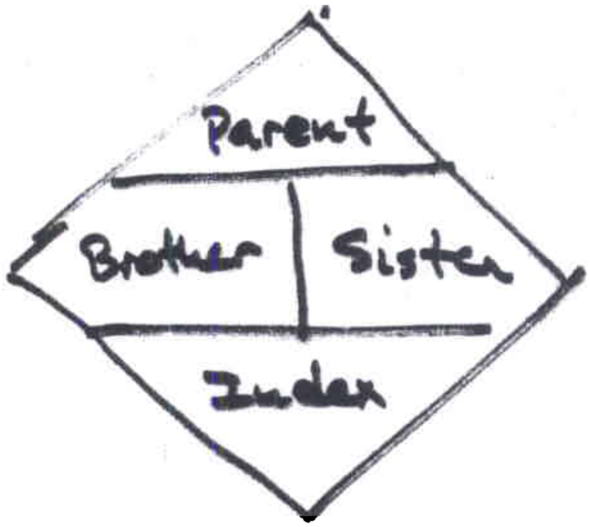
QNODE



Listplace
partial
group.



LEAF



list place
group