

## Properties on Leftchain Trees

Xiang, Limin

Department of Computer Science and CommunicationEngineering, Faculty of Information Science and Electrical Engineering : Graduate Student

Ushijima, Kazuo

Department Computer Science and CommunicationEngineering, Faculty of Information Science and Electrical Engineering, Kyushu University

<https://doi.org/10.15017/1475359>

---

出版情報 : 九州大学大学院システム情報科学紀要. 2 (1), pp.9-13, 1997-03-26. Faculty of Information Science and Electrical Engineering, Kyushu University

バージョン :

権利関係 :

## Properties on Leftchain Trees

Limin XIANG\* and Kazuo USHIJIMA\*\*

(Received December 24, 1996)

**Abstract:** Properties are well-researched not only on the whole set of binary trees but also on some subsets of binary trees. In this paper, we discuss properties on a subset of binary trees — leftchain trees. The concept of leftchain trees is presented with the background of certain applications and a necessary and sufficient condition is found for leftchain trees. A relation is given on leftchain trees, it is proved that the relation is a boolean lattice on leftchain trees, and rank and unrank functions for leftchain trees are obtained based on the result above. It is pointed out that properties on leftchain trees can be grafted on rightchain trees easily.

**Keywords:** Data structure, Binary tree, Leftchain, Boolean lattice

### 1. Introduction

As one of the important data structures or as a branch of graph theory, binary trees have been an interesting research objective for many researchers. Properties are researched on the whole set of binary trees, such as enumeration of binary trees<sup>1),7)</sup> and reconstruction of binary trees<sup>5),6),9)</sup>. Properties are also researched on some subsets of binary trees, such as AVL trees<sup>3)</sup> and red-black trees<sup>4)</sup>. In this paper, we will discuss properties on a new subset of binary trees — leftchain trees.

Leftchain trees can be regarded as a model for bus-structures, resources allocation, or other applications. In section 2, we give the definition of leftchain trees and prove a necessary and sufficient condition for leftchain trees. In section 3, we give a relation on leftchain trees, prove that the relation is a boolean lattice, and code leftchain trees with binary numbers. In section 4, we point out that by symmetry properties on leftchain trees can be grafted on rightchain trees.

### 2. Leftchain Trees

In this section, we give the definition of leftchain trees, and prove a necessary and sufficient condition for leftchain trees.

#### Definition 1.

A path  $p = p_1 p_2 \dots p_l$  in a binary tree is called a leftchain<sup>6)</sup> (rightchain) if and only if

- (a)  $p_1$  has no left (right) son, and

(b) for  $1 \leq i < l$ ,  $p_i$  is the left (right) son of its successor  $p_{i+1}$ , and

(c)  $p_l$  is either the root or the right (left) son of its parent.

$p_l$  is called the header of the leftchain (rightchain.)

**Example 1.** The example of leftchains and rightchains is shown in Fig. 1.

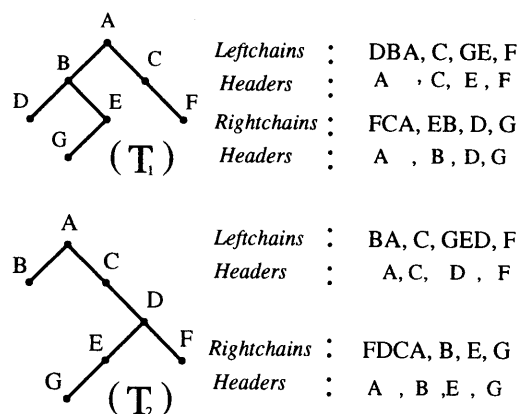


Fig.1 Leftchains and rightchains

#### Definition 2.

A nonempty binary tree  $T$  is called a leftchain tree if and only if headers of all leftchains of  $T$  are on the rightchain of which the header is the root of  $T$ .

The set is denoted by  $LCT_n$  of leftchain trees with  $n$  nodes.

#### Example 2.

$T_2$  of Example 1 is a leftchain tree, while  $T_1$  of

\* Department of Computer Science and Communication Engineering, Graduate Student

\*\* Department of Computer Science and Communication Engineering

**Example 1** is not.

**Definition 3.**

The  $i$ - $p$  sequence<sup>7)</sup> ( $p$ - $i$  sequence<sup>5)</sup>) of a binary tree with  $n$  nodes is an ordered  $n$ -tuple formed by labelling  $n$  nodes of the tree from 1 to  $n$  in inorder (preorder) and reading these labels in preorder (inorder.)

The  $i$ - $p$  sequence ( $p$ - $i$  sequence) of a binary tree  $\mathbf{T}$  is denoted by  $i\text{-}p(\mathbf{T})$  ( $p\text{-}i(\mathbf{T})$ ), and labels of a binary tree for the  $i$ - $p$  sequence ( $p$ - $i$  sequence) are called  $i$ -labels ( $p$ -labels.)

An  $i$ - $p$  sequence (or a  $p$ - $i$  sequence) can be regarded as a permutation of  $(1, 2, \dots, n)$ , and  $i$ - $p$  sequences are also called tree permutations<sup>7)</sup>.

**Example 3.**

$$i\text{-}p(\mathbf{T}_1) = (5, 2, 1, 4, 3, 6, 7),$$

$$p\text{-}i(\mathbf{T}_1) = (3, 2, 5, 4, 1, 6, 7),$$

$$i\text{-}p(\mathbf{T}_2) = (2, 1, 3, 6, 5, 4, 7), \text{ and}$$

$p\text{-}i(\mathbf{T}_2) = (2, 1, 3, 6, 5, 4, 7)$ , here,  $\mathbf{T}_1$  and  $\mathbf{T}_2$  are binary trees of **Example 1**.

**Definition 4.**

A permutation  $p = (p_1, p_2, \dots, p_n)$  of  $(1, 2, \dots, n)$  is a partition permutation if and only if

$$p = (L_1, L_1 - 1, \dots, 1, L_2, L_2 - 1, \dots, L_1 + 1, \dots, L_k, L_k - 1, \dots, L_{k-1} + 1),$$

here,  $1 \leq L_1 < L_2 < \dots < L_k = n$ .

The set is denoted by  $\mathbf{P}_n$  of partition permutations with  $n$  components.

The set of partition permutations  $\subseteq$  the set of tree permutations, since a characterization of tree permutations ( $i$ - $p$  sequences) is that  $(p_1, p_2, \dots, p_n)$  is a tree permutation if and only if there do not exist  $p_i, p_j$ , and  $p_k$  such that  $p_k < p_i < p_j$  for  $i < j < k$  <sup>7)</sup>.

**Example 4.**

In **Example 3**,  $i\text{-}p(\mathbf{T}_2) = (2, 1, 3, 6, 5, 4, 7)$  is a partition permutation, here,  $L_1 = 2 < L_2 = 3 < L_3 = 6 < L_4 = 7$ , while  $i\text{-}p(\mathbf{T}_1) = (5, 2, 1, 4, 3, 6, 7)$  is not.

The relations among leftchain trees,  $i$ - $p$  sequences,  $p$ - $i$  sequences, and partition permutations are given by the following **Theorem 1** and its **Corollary**.

**Theorem 1.**

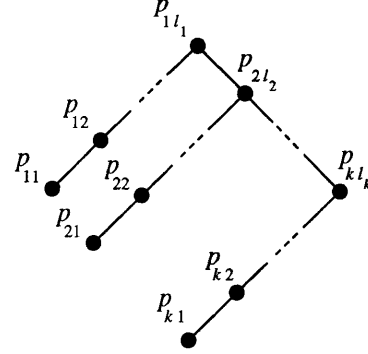
A nonempty binary tree  $\mathbf{T} \in LCT_n \iff i\text{-}p(\mathbf{T}) =$

$p\text{-}i(\mathbf{T})$ .

**Proof.**

" $\implies$ ": Let  $\mathbf{T}$  have  $k$  leftchains, and from root on the rightchain they are

$$p_{11}p_{12}\dots p_{1l_1}, p_{21}p_{22}\dots p_{2l_2}, \dots, p_{k1}p_{k2}\dots p_{kl_k}.$$



**Fig.2** The leftchain tree

The leftchain tree  $\mathbf{T}$  is shown in **Fig. 2**. Thus, by the definitions of  $i\text{-}p(\mathbf{T})$  and  $p\text{-}i(\mathbf{T})$ ,

$$\begin{aligned} i\text{-}p(\mathbf{T}) &= (L_1, L_1 - 1, \dots, 1, \\ &\quad L_2, L_2 - 1, \dots, L_1 + 1, \\ &\quad \dots, \\ &\quad L_k, L_k - 1, \dots, L_{k-1} + 1) \\ &= p\text{-}i(\mathbf{T}), \end{aligned}$$

here,  $L_i = \sum_{j=1}^i l_j$ , for  $1 \leq i \leq k$ .

" $\impliedby$ ": Let  $i\text{-}p(\mathbf{T}) = (u_1, u_2, \dots, u_n) = p\text{-}i(\mathbf{T}) = (v_1, v_2, \dots, v_n)$ , then  $u_i = v_i$  for  $1 \leq i \leq n$ .

Using induction on  $n$ .

(1) When  $n = 1$ , the result is true obviously.

(2) Assume when  $n < k$  ( $k > 1$ ) the result is true.

Let  $u_1 = m$ ,  $d_1(u_i)$  be the node with the label  $u_i$  under  $i$ -labels, and  $d_2(v_i)$  be the node with the label  $v_i$  under  $p$ -labels. Since  $d_1(u_1)$  is the root and  $d_1(u_1) = d_1(m) = d_2(v_m) = d_2(1)$ ,  $v_m = 1 = u_m$ . By the definition of  $i$ - $p$  sequence, under  $i$ -labels labels of nodes on the left subtree are less than the label of the root, and labels of nodes on the right subtree are bigger than the label of the root. Therefore,

*Case 1.*  $m = 1$ , the root  $d_1(u_1)$  has no left subtree, i.e.,  $d_1(u_1)$  is a leftchain of  $\mathbf{T}$ .

*Case 2.*  $m > 1$ ,  $d_1(u_1)$  has the left subtree with nodes  $d_1(u_2), \dots, d_1(u_m)$  and rooted by  $d_1(u_2)$ . Since  $i\text{-}p(\mathbf{T}^L) = (u_2, \dots, u_m)$ , where,  $\mathbf{T}^L$  is the left subtree of  $d_1(u_1)$ , and  $u_m = 1$ ,  $d_1(u_2)$  has no right subtree, and so on for  $d_1(u_j)$  ( $2 \leq j \leq m$ ), i.e.,  $d_1(u_m)d_1(u_{m-1})\dots d_1(u_1)$  is a leftchain of  $\mathbf{T}$ .

If  $\mathbf{T}$  (rooted by  $d_1(u_1)$ ) has no right subtree, the result holds. If  $\mathbf{T}$  has the right subtree  $\mathbf{T}^R$ , then

$$\begin{aligned} i\text{-}p(\mathbf{T}^R) &= (u_{m+1} - m, u_{m+2} - m, \dots, u_k - m) = \\ &= (v_{m+1} - m, v_{m+2} - m, \dots, v_k - m) = p\text{-}i(\mathbf{T}^R), \text{ by} \end{aligned}$$

the assumption of induction,  $\mathbf{T}^R \in LCT_{k-m}$ , so,  $\mathbf{T} \in LCT_k$ .

Therefore, from (1) and (2) the result holds.

□

From the proof of **Theorem 1**, it is not difficult to obtain the following result.

### Corollary

(1) A nonempty binary tree  $\mathbf{T} \in LCT_n \iff i.p(\mathbf{T}) \in \mathbf{P}_n$ .

(2)  $i.p(\mathbf{T}) = p.i(\mathbf{T}) \iff i.p(\mathbf{T}) \in \mathbf{P}_n$ .

### 3. Properties

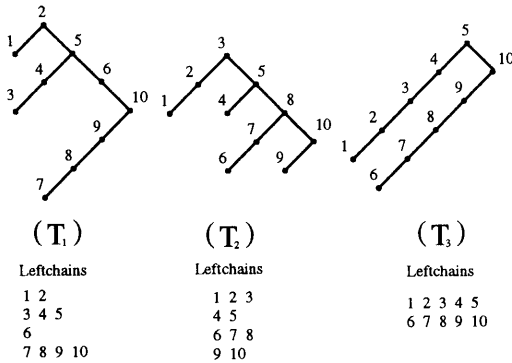
In this section, we first give a relation  $\trianglelefteq_C$  on  $LCT_n$ , and prove  $\langle LCT_n, \trianglelefteq_C \rangle$  is a partial order relation, and then by several bijective functions we obtain the result that  $\langle LCT_n, \trianglelefteq_C \rangle$  is order isomorphic with  $\langle \mathcal{P}(\mathcal{N}_{n-1}), \subseteq \rangle$ , where,  $\mathcal{N}_{n-1}$  is the set of positive integers less than  $n$ , and  $\mathcal{P}(\mathcal{N}_{n-1})$  is the power set of  $\mathcal{N}_{n-1}$ . Thereby, it is known that  $\langle LCT_n, \trianglelefteq_C \rangle$  is a boolean lattice and  $|LCT_n| = 2^{n-1}$ . Lastly, based on the result above the one-to-one correspondence is established between  $LCT_n$  and  $\mathbf{B}_{n-1}$ , here,  $\mathbf{B}_{n-1}$  is the set of binary numbers with  $n-1$  bits.

#### Definition 5

Given  $\mathbf{T}_1, \mathbf{T}_2 \in LCT_n$ ,  $\mathbf{T}_1$  has  $k$  leftchains,  $\mathbf{T}_2$  has  $l$  leftchains, and  $l \leq k$ .  $\mathbf{T}_2$  is called a composition of  $\mathbf{T}_1$ , which is denoted by  $\mathbf{T}_2 \trianglelefteq_C \mathbf{T}_1$ , if and only if, under  $i$ -labels, any leftchain of  $\mathbf{T}_2$  is composed of some leftchain(s) of  $\mathbf{T}_1$ .

#### Example 5

The example of  $\trianglelefteq_C$  is shown in **Fig. 3**.



**Fig.3** Example for  $\trianglelefteq_C$

$\mathbf{T}_3 \trianglelefteq_C \mathbf{T}_1$ , and  $\mathbf{T}_3 \trianglelefteq_C \mathbf{T}_2$ , while  $\mathbf{T}_2 \not\trianglelefteq_C \mathbf{T}_1$ .

**Theorem 2.**  $\langle LCT_n, \trianglelefteq \rangle$  is a partial order relation.

#### Proof.

(1) For any  $\mathbf{T} \in LCT_n$ , by the definition of  $\trianglelefteq_C$ ,  $\mathbf{T} \trianglelefteq_C \mathbf{T}$ , so  $\langle LCT_n, \trianglelefteq_C \rangle$  is reflexive.

(2) For any  $\mathbf{T}_1, \mathbf{T}_2 \in LCT_n$ , if  $\mathbf{T}_2 \trianglelefteq_C \mathbf{T}_1$  and  $\mathbf{T}_1 \trianglelefteq_C \mathbf{T}_2$ , by the definition of  $\trianglelefteq_C$ ,  $\mathbf{T}_1 = \mathbf{T}_2$ , so  $\langle LCT_n, \trianglelefteq_C \rangle$  is anti-symmetric.

(3) For any  $\mathbf{T}_1, \mathbf{T}_2, \mathbf{T}_3 \in LCT_n$ , if  $\mathbf{T}_2 \trianglelefteq_C \mathbf{T}_1$ , and  $\mathbf{T}_3 \trianglelefteq_C \mathbf{T}_2$ , by the definition of  $\trianglelefteq_C$ , under  $i$ -labels, any leftchain of  $\mathbf{T}_3$  is composed of some leftchain(s) of  $\mathbf{T}_2$ , and any leftchain of  $\mathbf{T}_2$  is composed of some leftchain(s) of  $\mathbf{T}_1$ . So, under  $i$ -labels, any leftchain of  $\mathbf{T}_3$  is composed of some leftchain(s) of  $\mathbf{T}_1$ , i.e.,  $\langle LCT_n, \trianglelefteq_C \rangle$  is transitive.

From (1), (2) and (3),  $\langle LCT_n, \trianglelefteq_C \rangle$  is a partial order relation.

□

It can be proved directly that  $\langle LCT_n, \trianglelefteq_C \rangle$  is a complemented distributive lattice (boolean lattice), but for simplicity, we will prove the result by establishing an order isomorphic relation between  $\langle LCT_n, \trianglelefteq_C \rangle$  and  $\langle \mathcal{P}(\mathcal{N}_{n-1}), \subseteq \rangle$ .

Based on **Theorem 1** and its **Corollary**, a leftchain tree with  $n$  nodes can be represented by an element of  $\mathbf{P}_n$ . In fact, for  $p \in \mathbf{P}_n$

$p = (L_1, L_1 - 1, \dots, 1, L_2, L_2 - 1, \dots, L_1 + 1, L_k, L_k - 1, \dots, L_{k-1} + 1)$  can be simplified as an ordered  $k$ -tuple  $(L_1, L_2 - L_1, \dots, L_k - L_{k-1})$ , or as another ordered  $k$ -tuple  $(L_1, L_2, \dots, L_k)$ .

#### Definition 6.

(1)  $\mathcal{T}_n = \{(a_1, a_2, \dots, a_k) | k \in \mathcal{N}_n, a_i \in \mathcal{N}_n \text{ for } 1 \leq i \leq k, \text{ and } \sum_{j=1}^k a_j = n\}$ .

(2)  $\mathcal{S}_n = \{(x_1, x_2, \dots, x_k) | k \in \mathcal{N}_n, x_i \in \mathcal{N}_n \text{ for } 1 \leq i \leq k, \text{ and } 1 \leq x_1 < x_2 < \dots < x_k = n\}$ .

#### Example 6.

$\mathcal{T}_3 = \{(1, 1, 1), (1, 2), (2, 1), (3)\}$ .

$\mathcal{S}_3 = \{(1, 2, 3), (1, 3), (2, 3), (3)\}$ .

The one-to-one correspondence between  $\mathbf{P}_n$  and  $\mathcal{T}_n$  can be established by a bijective function  $\mathcal{A}$  that

$$\mathcal{A} : \mathbf{P}_n \longrightarrow \mathcal{T}_n,$$

$$\mathcal{A}(L_1, L_1 - 1, \dots, 1, L_2, L_2 - 1, \dots, L_1 + 1, \dots, L_k, L_k - 1, \dots, L_{k-1} + 1)$$

$$= (L_1, L_2 - L_1, \dots, L_k - L_{k-1}), \text{ and}$$

$$\mathcal{A}^{-1}(a_1, a_2, \dots, a_k)$$

$$= (L_1, L_1 - 1, \dots, 1, L_2, L_2 - 1, \dots, L_1 + 1, \dots, L_k, L_k - 1, \dots, L_{k-1} + 1),$$

where,  $L_i = \sum_{j=1}^i a_j$ , for  $1 \leq i \leq k$ .

For  $\mathbf{T} \in LCT_n$ , we will denote  $\mathcal{A}(i.p(\mathbf{T}))$  by  $\alpha(\mathbf{T})$

simply. The implication of  $\alpha(\mathbf{T}) = (a_1, a_2, \dots, a_k)$  is that  $\mathbf{T}$  has  $k$  leftchains, and  $k$  leftchains have  $a_1, a_2, \dots, a_k$  nodes from the root of  $\mathbf{T}$  respectively.

The one-to-one correspondence between  $\mathbf{P}_n$  and  $\mathcal{S}_n$  can also be established by a bijective function  $\mathcal{B}$  that

$$\begin{aligned} \mathcal{B} : \mathbf{P}_n &\longrightarrow \mathcal{S}_n, \\ \mathcal{B}(L_1, L_1 - 1, \dots, 1, L_2, L_2 - 1, \dots, L_1 + 1, \dots, L_k, L_k - 1, \dots, L_{k-1} + 1) \\ &= (L_1, L_2, \dots, L_k), \text{ and} \\ \mathcal{B}^{-1}(L_1, L_2, \dots, L_k) \\ &= (L_1, L_1 - 1, \dots, 1, L_2, L_2 - 1, \dots, L_1 + 1, \dots, L_k, L_k - 1, \dots, L_{k-1} + 1). \end{aligned}$$

For  $\mathbf{T} \in LCT_n$ , we will denote  $\mathcal{B}(i\text{-}p(\mathbf{T}))$  by  $\beta(\mathbf{T})$  simply. The implication of  $\beta(\mathbf{T}) = (x_1, x_2, \dots, x_k)$  is that  $\mathbf{T}$  has  $k$  leftchains, and under  $i$ -labels labels for headers of  $k$  leftchains are  $x_1, x_2, \dots, x_k$  from the root of  $\mathbf{T}$  respectively.

The relation between  $\mathcal{T}_n$  and  $\mathcal{S}_n$  can be given by a bijective function  $\gamma$  that

$$\begin{aligned} \gamma : \mathcal{T}_n &\longrightarrow \mathcal{S}_n, \\ \gamma(a_1, a_2, \dots, a_k) &= (x_1, x_2, \dots, x_k), \text{ here, } x_i = \sum_{j=1}^i a_j, \text{ for } 1 \leq i \leq k, \text{ and} \\ \gamma^{-1}(x_1, x_2, \dots, x_k) &= (a_1, a_2, \dots, a_k), \text{ here, } a_1 = x_1 \\ &\text{and } a_i = x_i - x_{i-1} \text{ for } 1 < i \leq k. \end{aligned}$$

Obviously,  $\gamma = \beta \circ \alpha^{-1}$ .

### Definition 7.

Given  $t_1 = (a_1, a_2, \dots, a_k) \in \mathcal{T}_n$ ,  $t_2 = (b_1, b_2, \dots, b_l) \in \mathcal{T}_n$ , and  $l \leq k$ ,  $t_2$  is called a composition of  $t_1$ , which is denoted by  $t_2 \trianglelefteq_T t_1$ , if and only if for any  $i$  ( $1 \leq i \leq l$ ),  $\exists p_i$  ( $1 \leq p_i \leq k$ ) s.t.  $\sum_{j=1}^i b_j = \sum_{j=1}^{p_i} a_j$ .

**Example 7.**  $t_1 = (2, 3, 1, 4)$ ,  $t_2 = (3, 2, 3, 2)$  and  $t_3 = (5, 5)$ .  $t_3 \trianglelefteq_T t_1$  and  $t_3 \trianglelefteq_T t_2$ , while  $t_2 \not\trianglelefteq_T t_1$

From definitions of  $\trianglelefteq_C$ ,  $\trianglelefteq_T$  and  $\alpha$ , there is the result

**Lemma 1.** For  $\mathbf{T}_1, \mathbf{T}_2 \in LCT_n$ ,  $\mathbf{T}_2 \trianglelefteq_C \mathbf{T}_1 \iff \alpha(\mathbf{T}_2) \trianglelefteq_T \alpha(\mathbf{T}_1)$ .

### Definition 8.

Given  $s_1 = (x_1, x_2, \dots, x_k) \in \mathcal{S}_n$ ,  $s_2 = (y_1, y_2, \dots, y_l) \in \mathcal{S}_n$ , and  $l \leq k$ ,  $s_2$  is called a composition of  $s_1$ , which is denoted by  $s_2 \trianglelefteq_S s_1$ , if and only if

$$\{y_1, y_2, \dots, y_l\} \subseteq \{x_1, x_2, \dots, x_k\}.$$

**Example 8.**  $s_1 = (2, 5, 6, 10)$ ,  $s_2 = (3, 5, 8, 10)$  and

$s_3 = (5, 10)$ .  $s_3 \trianglelefteq_S s_1$  and  $s_3 \trianglelefteq_S s_2$ , while  $s_2 \not\trianglelefteq_S s_1$ ,

From definitions of  $\trianglelefteq_T$ ,  $\trianglelefteq_S$  and  $\gamma$ , there is the result

**Lemma 2.** For  $t_1, t_2 \in \mathcal{T}_n$ ,  $t_2 \trianglelefteq_T t_1 \iff \gamma(t_2) \trianglelefteq_S \gamma(t_1)$ .

**Definition 9.**  $\delta$  is a bijective function that

$$\begin{aligned} \delta : \mathcal{S}_n &\longrightarrow \mathcal{P}(\mathcal{N}_{n-1}), \\ \delta(x_1, x_2, \dots, x_k) &= \begin{cases} \emptyset, & k = 1 \\ \{x_1, x_2, \dots, x_{k-1}\}, & k > 1 \end{cases}, \text{ and} \\ \delta^{-1}(A) &= \begin{cases} (n), & \text{case1} \\ (x_1, x_2, \dots, x_{k-1}, n), & \text{case2} \end{cases} \end{aligned}$$

here,  $\text{case1} : A = \emptyset$  and  $\text{case2} : (A = \{x_1, x_2, \dots, x_{k-1}\}) \wedge (x_1 < x_2 < \dots < x_{k-1})$ .

From definitions of  $\trianglelefteq_S$  and  $\delta$ , there is the result

**Lemma 3.** For  $s_1, s_2 \in \mathcal{S}_n$ ,  $s_2 \trianglelefteq_S s_1 \iff \delta(s_2) \subseteq \delta(s_1)$ .

Now, we can give the relation between  $< LCT_n, \trianglelefteq_C >$  and  $< \mathcal{P}(\mathcal{N}_{n-1}), \subseteq >$ .

### Theorem 3.

$< LCT_n, \trianglelefteq_C >$  is order isomorphic with  $< \mathcal{P}(\mathcal{N}_{n-1}), \subseteq >$ .

### Proof.

From **Lemmas 1, 2** and **3**,  $f = \delta \circ \gamma \circ \alpha$  is an order isomorphic function from  $< LCT_n, \trianglelefteq_C >$  to  $< \mathcal{P}(\mathcal{N}_{n-1}), \subseteq >$ .

□

Thus, by **Theorem 3**, we have

### Corollary

(3)  $< LCT_n, \trianglelefteq_C >$  is a boolean lattice.

(4)  $|LCT_n| = 2^{n-1}$ .

Based on **Theorem 3** and its **Corollary**, when  $n > 1$  the one-to-one correspondence can be established between  $LCT_n$  and  $\mathbf{B}_{n-1}$ . Since between  $\mathcal{P}(\mathcal{N}_n)$  and  $\mathbf{B}_n$  there is the one-to-one correspondence  $\epsilon$  that

$$\epsilon : \mathcal{P}(\mathcal{N}_n) \longrightarrow \mathbf{B}_n,$$

$$\epsilon(A) = b_1 b_2 \dots b_n, \text{ here, } b_i = \begin{cases} 0, & i \notin A \\ 1, & i \in A \end{cases} \text{ for } 1 \leq i \leq n, \text{ and}$$

$$\epsilon^{-1}(b_1 b_2 \dots b_n) = \{i | (i \in \mathcal{N}_n) \wedge (b_i = 1)\},$$

when  $n > 1$  the one-to-one correspondence between  $LCT_n$  and  $\mathbf{B}_{n-1}$  can be expressed as  $g$  that

$$g = \epsilon \circ f = \epsilon \circ \delta \circ \gamma \circ \alpha = \epsilon \circ \delta \circ \beta, \text{ and}$$

$$g^{-1} = f^{-1} \circ \epsilon^{-1} = \alpha^{-1} \circ \gamma^{-1} \circ \delta^{-1} \circ \epsilon^{-1} =$$

$$\beta^{-1} \circ \delta^{-1} \circ \epsilon^{-1}.$$

i.e.,  $g$  is a rank function<sup>2),7)</sup> and  $g^{-1}$  is a unrank function<sup>2),7)</sup> for  $LCT_n$ .

### Corollary

(5) For  $\mathbf{T} \in LCT_n$  and  $n > 1$ ,  $\mathbf{T}$  has  $k$  leftchains  $\iff k - 1$  is the number of 1's in  $g(\mathbf{T})$ .

### Proof.

$\mathbf{T}$  has  $k$  leftchains

$$\iff \beta(\mathbf{T}) = (x_1, x_2, \dots, x_{k-1}, n)$$

$$\iff \delta \circ \beta(\mathbf{T}) = \{x_1, x_2, \dots, x_{k-1}\} \text{ and } x_1 < x_2 < \dots < x_{k-1}$$

$$\iff k - 1 \text{ is the number of 1's in } g(\mathbf{T}) = \epsilon(\delta \circ \beta(\mathbf{T})).$$

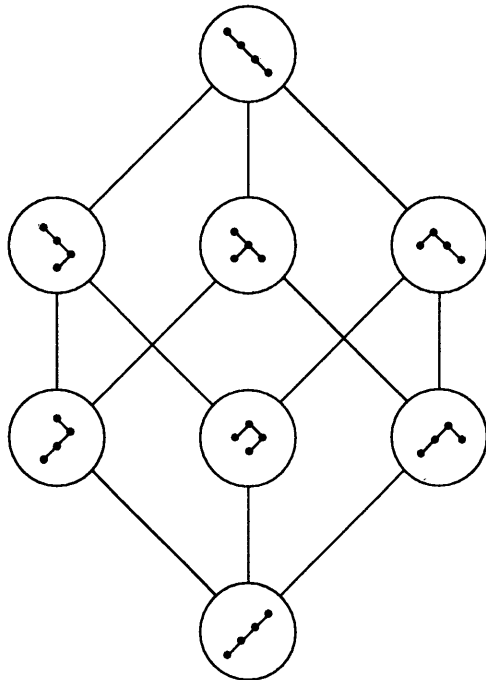
□

In fact, for  $\mathbf{T} \in LCT_n$  and  $n > 1$   $g(\mathbf{T})$  can be formed by labelling all edges on all leftchains with 0's and the rest (all edges on the rightchain) with 1's, and reading these labels in preorder.

**Example 9.**  $g(\mathbf{T}_1) = 010011000$ ,  $g(\mathbf{T}_2) = 001010010$  and  $g(\mathbf{T}_3) = 000010000$ , where,  $\mathbf{T}_1$ ,  $\mathbf{T}_2$  and  $\mathbf{T}_3$  are leftchain trees of **Example 5**.

## 4. Conclusion

The bus-structure with  $k$  branches and  $n$  nodes can be regarded as a leftchain tree with  $k$  leftchains and  $n$  nodes, and the problem can also be abstracted as a leftchain tree with  $k$  leftchains and  $n$  nodes that  $n$  resources are to be allocated among  $k$  users. By symmetry, properties on leftchain trees ( $LCT$ ) can



**Fig.4** the Hasse Diagram

be grafted on rightchain trees ( $RCT$ ) easily.

As the end of this paper, the Hasse Diagram<sup>8)</sup> of  $\langle LCT_n, \triangleleft_C \rangle$  and the comparison of  $LCT_n$ ,  $\mathbf{P}_n$ ,  $\mathbf{T}_n$ ,  $\mathbf{S}_n$ ,  $\mathcal{P}(\mathcal{N}_{n-1})$ ,  $\mathbf{B}_{n-1}$  and  $RCT_n$  are given as in **Fig. 4** and **Fig. 5** respectively for  $n = 4$ .

$LCT_n$	$\mathbf{P}_n$	$\mathbf{T}_n$	$\mathbf{S}_n$	$\mathcal{P}(\mathcal{N}_{n-1})$	$\mathbf{B}_{n-1}$	$RCT_n$
	$i_{-}P(LCT_n)$	$\alpha(LCT_n)$	$\beta(LCT_n)$	$f(LCT_n)$	$g(LCT_n)$	
	(1,2,3,4)	(1,1,1,1)	(1,2,3,4)	{1,2,3}	111	
	(1,2,4,3)	(1,1,2)	(1,2,4)	{1,2}	110	
	(1,3,2,4)	(1,2,1)	(1,3,4)	{1,3}	101	
	(2,1,3,4)	(2,1,1)	(2,3,4)	{2,3}	011	
	(1,4,3,2)	(1,3)	(1,4)	{1}	100	
	(2,1,4,3)	(2,2)	(2,4)	{2}	010	
	(3,2,1,4)	(3,1)	(3,4)	{3}	001	
	(4,3,2,1)	(4)	(4)	$\phi$	000	

**Fig.5** the comparison of  $LCT_n$ ,  $\mathbf{P}_n$ ,  $\mathbf{T}_n$ ,  $\mathbf{S}_n$ ,  $\mathcal{P}(\mathcal{N}_{n-1})$ ,  $\mathbf{B}_{n-1}$  and  $RCT_n$

## References

- 1) V. Bapiraju and V.V.B. Rao, "Enumeration of binary trees", *Inform.Process.Lett.*, **51**, 125-127, 1994.
- 2) E.A. Bender and S.G. Williamson, FOUNDATIONS OF APPLIED COMBINATORICS. Addison-Wesley Publishing Company, ISBN 0-201-51039-1, 69-80, 1991.
- 3) H. Cameron and D. Wood, "Balance in AVL trees and space cost of brother trees", *Theoret.Comput.Sci.*, **127**, 199-228, 1994.
- 4) H. Cameron and D. Wood, "Insertion Reachability, Skinny Skeletons, and Path Length in Red-Black Trees", *Information Science*, **77**, 141-152, 1994.
- 5) N. Gabrani and P. Shankar, "A note on the reconstruction of a binary tree from its traversals", *Inform. Process. Lett.*, **42**, 117-119, 1992.
- 6) V. Kamakoti and C.P. Rangan, "An optimal algorithm for reconstructing a binary tree", *Inform. Process. Lett.*, **42**, 113-115, 1992.
- 7) G.D. Knott, "A Numbering System for Binary Trees", *Comm.ACM*, **20** (2), 113-115, 1977.
- 8) H.F.Jr. Mattson, DISCRETE MATHEMATICS with applications. John Wiley & Sons, Inc., ISBN 0-471-59966-2, 473-473, 1993
- 9) S. Olariu, M. Overstreet and Z. Wen, "Reconstructing a binary tree from its traversals in doubly logarithmic CREW time", *J. Parallel Distrib. Comput.*, **27**, 100-105, 1995.