VNU. JOURNAL OF SCIENCE. NAT. SCI. t.XI. n°2-1995

OPTIMIZATION FOR ONE-DIMENSIONNAL BINARY SEARCH TREES

Do Duc Giao and A Min TJOA
Institute of Applied Computer Science
and Information Systems, University of Vienna (Austria)

1. Abst.

This paper introduces axiom schemes for binary search trees. Inference rules for binary search trees are specified. A prove of a theorem which shows that each tree can be uniquely transformed into an optimal tree by using the axiom schemes and the rules of inferences are introduced in this paper.

2. Introduction

The notion of a search tree plays an important role in computor science, especially in the theory of data lstructures. For that reason we can find many papers concerned with the theory of search trees in the literature. We noticed that, above all, questions of the optimal construction and inductive generation of search trees and studied, where equivalent transformations of search trees are offen used [1,2,3,4,5,6].

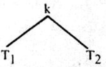
In this paper we will give the fundamentals of such theory and optimization problem for the set of one-dimensional binary search trees with infomations in their leafs.

3. One-dimensionnal binary search trees.

Let D and be the set of documents and the set of the nonegative integers. Let the symbols /</>/,/ τ not be in the set D u K. τ is the empty tree. We denote $D^+ := D \cup \{\tau\}$. Now we define the set TREE of all one- dimensionnal binary search trees with informations in leafs as follows :

Definition 1

1. d is a tree for vrery $d \in D^+$
2. If T_1 and T_2 are trees and $k \in K$, then k,T_1,T_2>or



is a tree

K is the set of keys of the set TREE of all one-dimensionnal binary search trees informations in theri leafs (Definition 1). We define the RESULT (T, ℓ) of searching in the tree $T \in$ TREE with the key $\ell \in K$ by

Definition 2

1. RESULT (d, ℓ): = d for every $d \in D^+$

2. RESULT $(k<T_1,T_2>\ell)$: = RESULT (T_1, ℓ) if $\ell \leq k$

RESULT $(k<T_1,T_2>\ell)$: = RESULT (T_2, ℓ) if $\ell > k$

The base of the following investigation is the definition of equivalence of trees of TREE. In the sense of retrieval theory another equivalent relation for trees is relevant.

Definition 3

Let T_1 and T_2 be trees of the set TREE. T_1 is equivalent to T_2 ($T_1 \approx T_2$) if and only if for every $l \in K$ the equation

RESULT (T_1, ℓ) = RESULT (T_2, ℓ) holds.

In the following by $T_1 \equiv T_2$ ($T_1 \neq T_2$) we denote that the tree T_1 is equality (inequality) to the tree T_2.

4. Derivability for formal equations of the set TREE

Let=be a new primitive symbol. We define the set EQU of formal equation for trees of the set TREE by.

<equation> : = <tree> = <tree>.

First we introduce a suitable notion of derivability for formal equations of the set EQU. Let be $X \subseteq$ EQU and $T_1 = T_2 \in$ EQU.

Definition 4

$T_1 = T_2$ is derivable from X ($X \vdash T_1 = T_2$ if and only if $T_1 = T_2 \in X$ or $T_1 = T_2$ can be constructed in a finite number of steps using elements of X by application of the following elementary rulees inference:

R1. If $T \in$ TREE then $X \vdash T = T$

R2. If $X \vdash T_1 = T_2$, then $X \vdash T_2 = T_1$

R3. If $X \vdash T_1 = T_2$ and $X \vdash T_2 = T_3$, then $X \vdash T_1 = T_2$

R4. If $X \vdash T1 = T1'$, then $X \vdash k<T1, , T2> = k<T1', T2>$

R5. If $X \vdash T_2 = T_2'$, then $X \vdash k<T_1, T_2> = k<T_1, T_2'>$

Now we formulate the syntactic theorem of replacement

Theorem 1

For every T_1, T_0, T_{00}, T_2 of TREE holds if T_2 is the result of a simultaneous replacement of the tree T_0 by the tree at some places in T_2, then : $X \vdash T_0 = T_{00}$, then $X \vdash T_1 = T_2$.

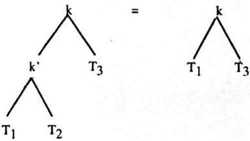
Proof. Induction on the complexity of the tree T_1.

5. Axiom system (AX) of the set TREE.

The problem of axiomatizing the equivalent relation is fundamental for applications in practice. We define the axiom system AX of the set TREE as follows AX:=ax_1 u ax_2 u ax_3 u ax_4, where we define axi (i=1,2,3,4) as follows:

Axiom scheme **ax**$_1$

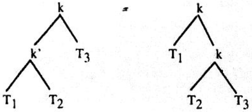
For each T_1, T_2 , T_3 of TREE, and , $k' \in K$ the following formal equation < $k'< T_1, T_2>, T_3>$ = $k< T_1, T_3>$ or



s an axiom if $k \leq k'$

Axiom scheme ax_2

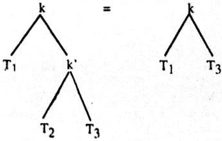
For every T_1, T_2 , T_3 of TREE ınd k, $k' \in K$ the following formal :quation $k<k'<T_1,T_2>,T_3>$ = $k'<-T_1$, ‹$<T_2,T_3>>$or



is an axiomif $k>k'$

Axiom scheme ax_3

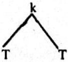
For ever T_1, T_2, T_3 of TREE and k,$k' \in K$ the following formal equation $k<T_1, k'<T_2, T_3>>=k<T_1,-T_3>$or



is an axiom if $k \geqslant k'$

Axiom scheme ax_4

For each $T \in$ TREE and $k \in K$ the following formal equation $k<T,T>=T$ or



= T is an axiom

We can prove the following consistency theorem

Theorem 2

Let T_1 and T_2 be trees of the set TREE. IF AX $\vdash T_1 = T_2$, then $T_1 \approx T_2$

Proof. By Induction on the length of a derivation from AX.

To prove the existence theorem in section 7 we formulate the following lemmas :

Lemma 1

For every T_1, T_2, T_3 of TREE and k, $k' \in K$ and $k<k'$ we have AX$\vdash$ $k<T_1, k'<T_2, T_3>>=k'<k<T_1,T_2>,T_3>$.

Proof. By using the axiom scheme ax2 and the rule R_2

Lemma 2

For every T_0, T_1 of TREE, k, k' $\in$ K and k' >k we have AX $\vdash$ $k< T_0, k'<T_0,T_1>>= k'<T_0, T_1>$.

Proof. By using the lemma 1. The axiom scheme ax4 and the theorem 1.

6.- Normal forms and uniqueness theorem

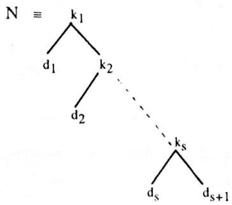
We define the following notion of a normal form of a tree of TREE

Definition 5

A tree N is said to be a normal; from if and only if

1. N $\equiv$ d for each d $\in$ D^+ or
2. $N \equiv k1 <d1, k2<d2...,ks<ds, ds+1>...>$or

Where d1,d2...,ds+1 $\in$ D+ ; di $\neq$ di+1 For every i=1,2...,s; k1,k2...,ks $\in$ K and $k1<k2<...<ks$ (s $\geq$1).



We have the following theorem

Theorem 3 (Uniqueness theorem)

Let N and N' be normal forms $\in$ TREE. If N $\approx$ N', then N $\equiv$ N'

Proof. For N and N' we have the following four cases :

Case 1. N $\equiv$ d and N' $\equiv$ d', where d,d' $\in$ D^+

Here our theorem triveally holds.

Case 2. N $\equiv$ d and $N' \equiv p_1<d_1', p_2< d_2',...p_y <d_y', d'_{y+1}>...>$

where $d'_i \neq d'_{i+1}$ for each i = 1,2,...y and $p_1<p_2<...<p_y$

Then we obviously obtain that N $\approx$ N'.

Case 3. $N \equiv k_1 < d_1, k_2<d_2,...,k_s<d'_s\ d_{s+1}>...>$ and N'$\equiv$ d' where $d_i \neq d_{i+1}$ for every i=1,2...s and $k_1<k_2<...<k_s$.

This case is proved analogously to case 2.

Case 4: $N \equiv k_1<d_1,k_2<d_2,...,k_s<d_s, d_{s+1}>...>$ and

$N' \equiv p_1<d'_1, p_2<d'_2,...p_y<d'_y, d'_{y+1}>...>$

where $d_i \neq d_{i+1}$ for every $i=1,2...s$; $k_1 < k_2 <...< k_s$; $d'_j \neq d'_{j+1}$ for each $j=1,2,...,y$ and $p_1 < p_2 <... p_y$.

In this case we obtain that $N \equiv N'$ i.e. $s = y$(1); $k_1 = p_1, k_2 = p_2, ..., k_s = p_y$(2) and $d_1 = d'_1$, $d_2 = d'_2, ... d_s = d'_y, d_{s+1} = d'_{y+1}$ (3)

Let be $N \approx N'$ (4), for s and y we have the following two cases :

4.1 s = y

In this case we obtain that $k_i = p_i$, $i = 1,2,...,s$. In contrary to the above it is stated as, $k_{i0} \neq p_{i0}$ ($i_0 \in \{1,2,...s\}$). Let $k_{i0} < p_{i0}$.

The case $k_{i0} > p_{i0}$ is proved analogously to case $k_i < p_{i0}$,

Let ℓ_{i0}, $\ell'_{i0} \in K$ and $\ell_{i0} = k_{i0}$, $\ell'_{i0} = k_{i0+1}$, where $k_{i0} = \ell_{i0} < p_{i0}$ and $k_{i0} < k_{i0+1} \leq p_{i0}$, k_{i0+1}. From the definition 2 and (4) it follows that RESULT (N, ℓ_{i0}) = RESULT (N', ℓ_{i0}), i.e. $d_{i0} = d'_{i0}$ (5) and RESULT (N, ℓ_{i0}) = RESULT (N', ℓ'_{i0}),i.e. $d_{i0+1} = d'_{i0}$ (6). From (5) and (6) it follows that $d_{i0} = d_{i0+1}$ and hence a contradiction, i.e. in this case $k_i = p_i$ for every $i=1,2,...,s$ (2). From (4), (1) and (2) it follows that $d_1 = d'_1, ... d_{s+1} = d'_{y+1}$. i.e. $N \equiv N'$

4.2. s ≠ y

Let $s<y$, i.e. $y = s + r$, $r \neq 1$. The case $s>y$ is proved analogously to case $s<y$. In this case is proved analogously to ca-se 4.1: $k_1 = p_1$, $k_2 = p_2, ..., k_s = p_s$ (7).

Let $\ell_1 = p_{s+1}$ and $\ell'_1 = p_{s+2}$ (if $r>1$); $\ell_2 = p_{s+1}$ and $\ell'_2 = p_{s+1} + 1$ (if $r = 1$). From the definition 2 and (4) it follows that RESULT (N, ℓ_1) = RESULT (N', ℓ_1), i.e. $d_{s+1} = d'_{s+1}$ (8) and RESULT (N, ℓ'_1) = RESULT (N', ℓ'_1, i.e. $d_{s+1} = d'_{s+1}$ (9). From (8) an (9) it follows that $d'_{s+1} = d'_{s+2}$ and hence a contradiction, i.e. s=y in case r>1.

In this case r = 1 it follows from the definition 2, (4) that

RESULT (N, ℓ_2) = RESULT (N', ℓ_2), i.e. $d_{s+1} = d'_{s+1}$ (10) and

RESULT (N, ℓ'_2) = RESULT (N', ℓ'_2), i.e. $d_{s+1} = d'_{s+2}$ (11) . $d'_{s+1} = d'_{s+2}$

it follows from (10) and (11) and hence a contradiction, i.e.s = y in the case r=1.

N N' immediately follows from the case 4.1 and 4.2.

7. Existence theorem and axiomatization theorem

First we will prove the theorem which says that each tree of TREE can be uniquely ransformed into a normal form.

Theorem 4 (Existence theorem)

To every tree T N∈ TREE we can construct one and only one normal form N ∈ TREE such hat $T \approx N$ (1) and AXN$\vdash$ T = N(2).

Proof. The part (1) follows from the second assertion of our theorem by applying th theorem 2. This part (2) is proved by induction on the complexity of T.

Initial step

$T \equiv d$ N$\in D^+$. We define N:= d and AX$\vdash$ T = N follows from the rule R_1.

Induction step

$T \equiv k<T_1, T_2>$. Our induction supposition yields AX $\vdash T_1 = N_1$ (1)

AX $\vdash T_2 = N_2$ (2), where the tree N_i is the normal form of the tree T_i (i= 1,2). From (1 and (2) it follows by using the rules R_3, R_4 and R_5 that AX$\vdash$ T = $k<N_1,N_2>$. For N_1 and N we have following cases:

Case 1. $N_1 \equiv d_1$, and $N_2 \equiv d_2$. For AX$\vdash$ T=k< d_1, d_2> we have the following possibilities:

1.1. $d_1 \neq$N d2. In this case we define N: = $k<d_1, d_2>$.

1.2. $d_1 = d_2$. In this case we define N: = N_1 (or N_2) by using the axiom scheme ax_4 an rule R_3.

Case 2. N_1 N≡ d_1 and N_2 N≡ $p_1<d_1$N $p_2<d_2$, ..., $p_y<d_y$; d_{y+1}>...>, where $d'_i \neq d'_{i+1}$ fo every i = 1, 2,...y ; $p_1<p_2<...<p_y$ and

AX$\vdash$ T = $k<d_1$, $p_1<d_1$N, $p_2<d_2$, ...,$p_y<d_y$, d_{y+1}>...>> (3)

For (3) we have following case:

2.1. $k<p_1$

2.1.1. $d_1 \neq d'_1$. In this case we define N : = $k<d_1, N_2>$ and AX$\vdash$ T = N follows from (3 by using the rules R_1 and R_3.

2.1.2. $d_1 = d'_1$. We define N : = N_2 and AX $\vdash$ T = N follows from (3) by using th lemma 2 and the rule R_3.

2.2. $p_i \leq k< p_{i+1}$, i = 1,2,...,y -1.

AX$\vdash$ T = $k<d_1$ $p_{i+1}<d_{i+1}$,...,$p_y<d'_y$ d'_{y+1} >...> follows from (3) by using the axior scheme ax3 and the rule R_3. This case is proved analogously to case 2.

2.3. $k>p_y$. AX$\vdash$ T = $k<d_1$, d'_{y+1}> follows from (3) by using the axiom scheme ax_3 an the rule R_3. This case is proved analogously to case 1.

Case 3. $N_1 \equiv q_1<d_1$, q_2 $<d_2$,..., $q_x<d_x$, d_{x+1}>...>and N_2 N≡ d, where $d_i \neq$Nd_{i+1} for every = 1,2,...,x ; $q_1<q_2<...$ q_x and AX $\vdash$ T = $k<q_1<d_1$, $q_2<d_2$,...,$q_x<d_x$,d_{x+1}>...>,d>(4).

For (4) we have the following cases :

3.1. $k \leq q_1$ AX$\vdash$ T = $k<d_1$, d> follows from (4) by using the axiom scheme ax_1 and th rule R_3.

This case is proved analogously to case 1.

3.2. $q_i<k \le q_{i+1}$, i = 1, 2,... ,x-1.

AX $\vdash$ T = $q_1<d_1$, $q_2<d_2$, ...,$k<d_{i+1}$,d>...> (5) follows from (4) by using the axiom schemes ax_1, ax_1 the theorem 1 and the rule R_3.

For (5) we have the followsing two cases :

3.2.1. d_{i+1} N≠ d. We define N:=q_1 < d_1, q_2 <d_2,...,k<d_{i+1}, d>...> and AX $\vdash$ T = N follows from (5) by using the rules $R_1$and R_3

3.2.2. d_{i+1} = d. AX$\vdash$ T = $q_1<d_1$, $q_2<d_2$,...$q_i<d_i$,d_{i+1}>...> (6) follows the axiom scheme ax4 and the theorem 1.

In this case we define N: = $q_1<d_1$, $q_2<d_2$,...$q_i<d_i$, d_{i+1}>...> and AX$\vdash$ T = N follows from (6) by using the ruler R_1 and R_1.

3.3. k>q_x. AX $\vdash$ T = $q_1<d_1$, $q_2<d_2$....$q_x<d_x$, k<d_{x+1}, d>...> follows from (4) by using the axiom scheme ax_2 and the rule R_3.

This case is proved analogously to case 3.2.

Case 4. N_1 ≡ $q_1<d_1$, q_2 < d_2,... q_x < d_x, d_{x+1} >...> and N_2 ≡N $p_1<d_1$, p_2< d_2,...,$p_y<d_y$, d_{y+1}>...>, where $d_i \neq d_{i+1}$ for each i = 1,2,....x ; $q_1<q_2<...<q_x$; d'_j ≠N d'_{j+1} for every j = 1,2,...,y and $p_1<p_2<...<p_y$.

AX$\vdash$ T = k< $q_1<d_1$, $q_2<d_2$ ....,$q_x<d_x$, d_{x+1}>...>,$p_1<d'_1$,...,p_y< d'_y,d'_{y+1}>...>>(7)For (7) we have the following cases:

4.1.k ≤ q_1. AX $\vdash$ T = k<d_1, $p_1<d'_1$, $p_2<d'_2$,...$p_y<d'_y$, d'_{y+1}>...>> follows from (7) by using the axiom scheme ax1 and the rule R_3. This case is poroed analogously to case 2.

4.2. $q_i<k \le q_{i+1}$, i = 1,2,...x -1.

AX$\vdash$ T = $q_1<d_1$, $q_2<d_2$,...q_i< d_i, k<d_{i+1}, $p_1<d'_1$, p_2 < d'_2,...,$p_y<d'_y$, d'_{y+1}>...>> (8) follows from (7) by using the axiom schemes ax_1, ax_2 and rule R_3.

For (8) we have the following cases:

4.2.1. k<p_1

4.2.1.1. d_{i+1}≠ d'_1 We define N := $q_1<d_1$, $q_2<d_2$,...,$q_i<d_i$, k<d_{i+1}, p_1< d'_1, p_2< d_2,...,$p_y<d'_y$, d'_{y+1}>...> and AX$\vdash$ T=N follows from the rules R_1 and R_3.

4.2.1.2. $d_{i+1}=d'_i$ .AX$\vdash$ T = $q_1<d_1$, $q_2<d_2$,....$q_i<d_i$, $p_1<d'_1$, $p_2<d'_2$,...,$p_y<d'_y$, d'_{y+1}>...>> follows from (8) by using the lemma 2 and the theorem 1.

This case is proved analogously to case 4.2.

4.2.2. $p_i \le k < p_{i+1}$, $i = 1,2,\ldots, y-1$.

$AX \vdash T = q_1 < d_1, q_2 < d_2, \ldots, q_i < d_i \; k < d'_i, p_{i+1} < d'_{i+1}, \ldots, p_y < d'_y, d'_{y+1} > \ldots >>$ follows from (8) by using the axiom scheme ax_3, the theorem 1 and the rule R_3. This case is proved analogously to case 4.2.

4.2.3. $k \ge p_y$.

From (8) it follows by using the axiom scheme ax_3 the theorem 1 and the rule R_3 that: $AX \vdash T = q_1 < d_1, q_2 < d_2, \ldots, q_i < d_i, k < d'_y, d'_{y+1} > \ldots >$. This case is proved analogously to case 4,2.

4.3. $k > q_x$.

$AX \vdash T = q_1 < d_1, q_2 < d_2, \ldots, q_x < d_x, k < d_{x+1}, p_1 < d'_1, p_2 < d'_2, \ldots, p_y < d'_y, d'_{y+1} > \ldots >>$ follows from (7) by using the axiom scheme ax_2, the theorem 1 and the rule R_3. This case is proved analogously to case 4.2.

The uniqueness it follows from the theorem 2 and 3.

Now we are going to prove the completeness and axiomtization theorems.

Theorem 5 (Completeness theorem).

Let T_1 and T_2 be trees of TREE. IF T_1 N≈ T_2, then $AX \vdash T_1 = T_2$.

Proof. Let $T_1 \approx T_2$ (1). By the theorem 4 there are normal forms N_1 and N_2 such that $T_1 \approx N_1$ (2), $AX \vdash T_1 = N_1$ (3), $T_2 \approx N_2$ (4) and $AX \vdash T_2 = N_2$ (5) holds. From (1), (2) and (4) it follows that $N_1 \approx N_2$ (6). From (6) we get $N_1 = N_2$ by using the theorem 3 and hence rule R_1 leads us to: $AX \vdash N_1 = N_2$ (7). This result implies $AX \vdash T_1 = T_2$ by applying the rules R_2, R_3 to (3), (5) and (7)

Theorem 6 (Axiomatization theorem).

Let T_1, T_2 be trees of TREE. $T_1 \approx T_2$ if and only if $Ax \vdash T_1 = T_2$

Proof. By using the theorem 2 and 5.

8. Reduced forms and optimization theorem

First we define the following notions

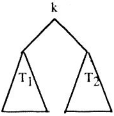
For every tree T of TREE we define:

$\gamma(T)$: = the number of all nodes and leafs of T and

Deep (d) : = the number of arcs of the way from the root to leaf d of T.

Definition 6.

Let R be a tree of TREE. R is said to be a reduced from of TREE if and only if R ≡ d ∈ D^+ or R≡



for every subtrees $T_1 \equiv k_1 < T_{11}, T_{12} >$ and $T_2 \equiv k_2 < T_{21}, T_{22} >$ it holds: $k_1 < k$; $k_2 > k$; $T_{11} \neq T_{12}$; $T_{21} \neq T_{22}$; and $/Deep(d_i) - Deep(d_j)/ \leq$ for each $d_i, d_j (i \neq j)$.

Where k, k_1, k_2 are keys of the set K and d_i, d_j are leafs of R.

Definition 7

A tree T_0 of TREE is said to be an optimal if and only if

$\gamma(T_0) = \min \{\gamma(T) : T \in TREE \text{ and } T \approx T_0\}$.

holds.

Theorem 7 : (Optimization theorem)

To each tree T ∈ TREE we can construct one and only one reduced from R such that.

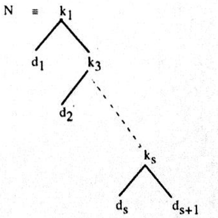
1. $T \approx R$
2. $AX \vdash T = R$
3. $\gamma(R) = \min \{\gamma(T') : T' \in TREE \text{ and } T' \approx R\}$

Proof.

The part (1) follows from the part (2) by using the theorem 2. To every tree T ∈ TREE we can construct a normal from N such that $T \approx N$ (4) and $AX \vdash T = N$ (5) by using the theorem 4.

If $N \equiv d \in D^+$, then we define R: = d and here our theorem trivially holds.

If $N \neq d \in D^+$, i.e. $N \equiv k_1$

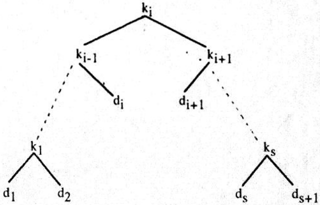


(6)

where $d_1, d_2, ..., d_{s+1} \in D^+$; $d_i \neq d_{i+1}$ for every i=1,2,...,s ; $k_1, k_2, ..., k_s \in K$ and $k_1 < k_2 < ... < k_s$ $(s \geq 1)$.

From(6) it follows by using the asiom scheme ax_2, the theorem 1, and the rules R_1, R_3 that : $AX \vdash N =$

Where $i = \left|\frac{S}{2}\right|$ and $k_1 < ... < k_{i-1} < k_i < k_{i+1} < ... < k_s$.

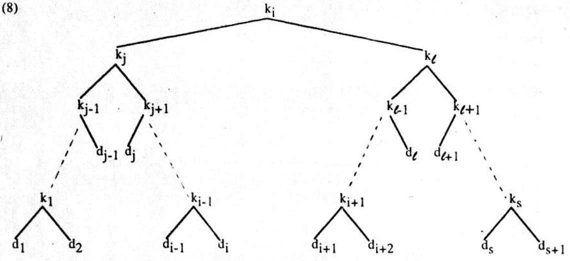


(7)

From (7) it follows by using the axiom scheme ax_2, the theorem 1 and the rule R_2 in the left - and right subtrees of the rot k_i :

$AX \vdash N =$

(8)



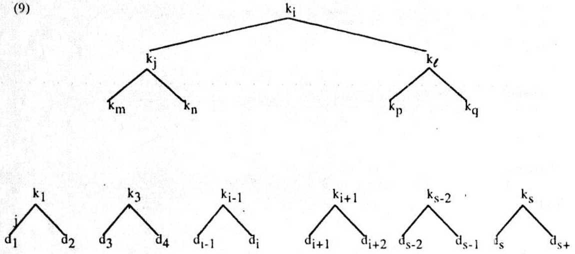
where $j = \left|\frac{i-1}{2}\right|; \ell = \left|\frac{s-i}{2}\right|$; and $k_j < k_i < k_\ell$; $k_{j-1} < k_j < k_{j+1} < k_{\ell-1} < k_{\ell+1}$;

$k_1 < ... < k_j - 1 < k_j < k_{j+1} < ... < k_{i-1} < k_i < k_{i+1} < ... \leq k_{\ell-1} < k_i < k_{\ell+1} < ... < k_s$.

From (8) it follows in a finite number of steps by using the axiom scheme ax_2, the theorem 1 and the rule R_3 that.

$AX \vdash N =$

(9)



where $m = \left|\frac{j-1}{2}\right|, n = \left|\frac{i-j}{2}\right|, p = \left|\frac{\ell-i}{2}\right|, q = \left|\frac{s-\ell}{2}\right|, ...$

$k_j < k_i < k_1$; $k_m < k_j < k_n < k_i < k_p < k_1 < k_q$; ...; $d_1 \neq d_2$; $d_3 \neq d_4$; ...; $d_{i-1} \neq d_i$; $d_{i+1} \neq d_{i+2}$; ...; $d_{s-2} \neq d_{s-1}$; $d_s \neq d_{s+1}$ and from the definition of the number i, j, l, m, n, p, q, ... it follows that.

$|Deep(d) - Deep(d')| \leq 1$ for every d and d' of the set $\{d_1, d_2, ..., d_{s+1}\}$.

In this case we define R: = the right tree of the formal equation in Figure (9). From (9) it follows AX $\vdash$ T = R by using (5) R_1 and R_3. The result (5) if and only if $\exists\ T \equiv T_1, T_2, ..., T_n \equiv N$ such that $T_j \approx T_{j+1}$ and AX $\vdash T_j = T_{j+1}$, j=1,2,...n-1. Where, the tree T_{j+1} is the result by using the axiom schemes in the tree T_j and $\gamma(T) = \gamma(T_1) \geq \gamma(T_2) \geq ... \geq \gamma(T_n) = \gamma(N)$, i.e.

$\gamma(N) = \min \{\gamma(T_1), \gamma(T_2), ..., \gamma(T_n)\}$. Let $T_i \in$TREE, $T_i \notin \{T_1, T_2, ... T_n\}$ and $\gamma(T_i) < \gamma(N)$ (10) and $T_i \approx N$(11).

To T_i we can construct a normal from N_i such that $T_i \approx N_i$ (12) and AX $\vdash T_i = N_i$(13) by using the theorem 4.

From(11) and(12) it follows that $N \approx N_i$(14). $N \equiv N_i$(15) follows from(14) by using the theorem 3, i.e $\gamma(N) = \gamma(N_i)$ (16).

The result AX $\vdash T_i = N_i$ if and only if $\exists\ T_i \equiv T_{i1}, T_{i2}, ..., T_{im} \equiv N_i$ such that $T_{ij} \approx T_{ij+1}$ and AX $\vdash T_{ij} = T_{ij+1}$, j=1, 2, ..., m-1.

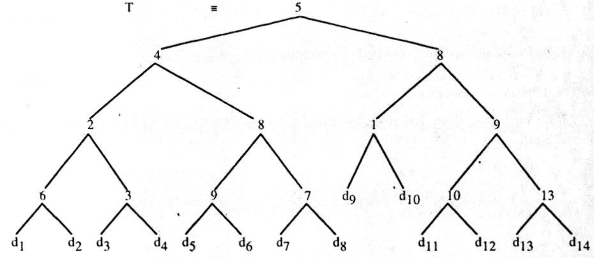
where $\gamma(T_i) = \gamma(T_{i1}) \geq \gamma(T_{i2}) \geq \ldots \geq \gamma(T_{im}) = \gamma(N_i)$ (17). From (16) and (17) it follows $\gamma(T_i) \geq \gamma(N)$ and hence a contradiction, i.e.

$\gamma(N) = \min \{ \gamma(T' : T' \in$ TREE and $T' \approx R \}$ (18). For the definition of the axiom scheme ax_2 we have $\gamma(N) = \gamma(R)$ (19), and $N \approx R$ it follows that

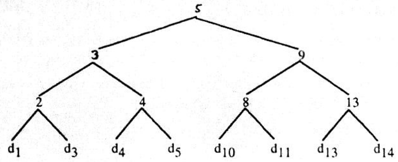
$\gamma(R) = \min \{ \gamma(T') : T' \in$ TREE and $T' \approx R \}$.

A example

Let



be a tree of TREE, where $d_i \neq dj$ for every $i \neq j$ and i,j = 1,2, ..., 14. To this tree we can construct a normal form N≡ 2< d_1, 3< d_3, 4< d_4, 5< d_5, 8< d_{10}, 9< d_{11}, 13< d_{13}, d_{14} > ... > and a reduced from R≡



where $T \approx N \approx R$, $\gamma(N) = \gamma(R) = 15 = \min \{ \gamma(T') : T' \in$ TREE and $T' \approx R \}$. [6,7,8,9]

9. Conclusion and further research

The efforts to optimize one - dimensional binary search trees as introduced in this paper are quite useful for pracical applications, especially for the representation of range queries, where the information about secondary keys defined on ranges are organized as a binary search tree.

The next investigations which are in preparation are dealing with the optimization of n-dimensional binary search trees.

REFERENCES

1. **Knuth, D.E. :** Optimum binary search trees, Acta Inf. I,1971

2. **Mehlhorn, K. :** Nearly optimal binary search trees. Acta Inf. 5(1975).

3. **Lee,C.C., Lee, D.D. and Wong, C.K. :** Generating binary trees of bounded height. Acta Inf.23 (1986), 529 - 544.

4. **Sprugnoli, R. :** The generation of binary trees as a numerical problem. J.of Association for Computing Machinery, Vol. 39,2 (4.1992).

5. **Thiele, H.:** On equivalent transformations of one - dimensional binary search trees. Pr. IPI. PAN, 411 (1980), 87-89.

6. **Do Duc Giao :** Optimization for n-dimensional Binary search trees.

   Twelfth European Meeting on Cybernetics and Systems Research- 1994, Vienna, Austria, 4-8 April 1994, Proceedings, pp.1377 - 1384.

7. **Do Duc Giao :** General equivalence relations between the set TREE/(Formula)^n, Y/, TREE/B,Y$^+$/ and TREE/V,Y$^+$/.

8. **Do Duc Giao :** Development of TREE/ (Formula)^n, Y/. General equivalence relations between the sets TREE/U(Formula)^n, Y/, TREE/B,Y$^+$/, and TREE/V,Y$^+$/.

9. Tạp chí khoa học, số 4/1993, Tr.13-17, ĐHTH Hà Nội.

VẤN ĐỀ TỐI ƯU ĐỐI VỚI CÁC CÂY NHỊ NGUYÊN MỘT CHIỀU

*Đỗ Đức Giáo * A Min TJoa*

Viện ứng dụng Khoa học máy tính và các hệ thông tin

Đại học Viên

Đưa ra các khái niệm về cây nhị nguyên một chiều, sự tương đương giữa các cây và khái niệm dẫn được đối với cây.

Kết quả chính là dùng hệ tiêu đề hóa để từ một lớp phân hoạch tương đương giữa các cây, xây dựng được một cây tối ưu (duy nhất) trong lớp phân hoạch trên. Thay cho việc phân loại, tìm kiếm các thông tin trên một lớp các phân hoạch, ta chỉ cần làm việc trên cây tối ưu là đủ.