ARTICLE IN PRESS
Discrete Applied Mathematics ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

Note

Decision trees with minimum average depth for sorting eight elements

Hassan AbouEisha, Igor Chikalov, Mikhail Moshkov *

Computer, Electrical and Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology, Thuwal 23955-6900, Saudi Arabia

ARTICLE INFO

Article history:
Received 12 June 2014
Received in revised form 14 October 2015
Accepted 26 October 2015
Available online xxxx

Keywords:
Sorting
Decision tree
Average depth

ABSTRACT

We prove that the minimum average depth of a decision tree for sorting 8 pairwise different elements is equal to 620160/8!. We show also that each decision tree for sorting 8 elements, which has minimum average depth (the number of such trees is approximately equal to 8.548×10^{326365}), has also minimum depth. Both problems were considered by Knuth (1998). To obtain these results, we use tools based on extensions of dynamic programming which allow us to make sequential optimization of decision trees relative to depth and average depth, and to count the number of decision trees with minimum average depth.

1. Introduction

The problem of sorting n pairwise different elements from a linearly ordered set is one of the model problems in algorithm theory [6]. For solving this problem, we use binary decision trees [4,8] where each step is a comparison of two elements. The minimum number of nodes in such a tree is equal to $2(n!) - 1$. For $n = 2, \ldots, 11$, the minimum depth of a decision tree for sorting n elements is equal to the well known lower bound $\lceil \log_2(n!) \rceil$ [10]. For $n = 2, 3, 4, 5, 6, 9, 10$, the minimum average depth of a decision tree for sorting n elements is equal to the known lower bound $\varphi(n)/n!$, where $\varphi(n) = (\lceil \log_2 n! \rceil + 1) \cdot n! - 2^{\lceil \log_2 n! \rceil}$ is the minimum external path length in an extended binary tree with $n!$ terminal nodes [6]. Césary [3] proved that, for $n = 7$ and $n = 8$, there are no decision trees for sorting n elements whose average depth is equal to $\varphi(n)/n!$. Kollár [7] found that the minimum average depth of a decision tree for sorting 7 elements is equal to 62416/7!. We find that the minimum average depth of a decision tree for sorting 8 elements is equal to 620160/8!.

Another open problem considered by Knuth [6] is the existence of decision trees for sorting n elements which have simultaneously minimum average depth and minimum depth. As it was mentioned by Knuth in [6], if a decision tree for sorting n elements has average depth equal to $\varphi(n)/n!$ then this tree has depth equal to $\lceil \log_2 n! \rceil$. Therefore, for $n = 2, 3, 4, 5, 6, 9, 10$, each decision tree for sorting n elements, which has minimum average depth, has also minimum depth. We extended this result to the cases $n = 7$ (Kollár in [7] did not consider this question) and $n = 8$. For $n = 2, \ldots, 8$, we counted also the number of decision trees for sorting n elements which have minimum average depth. In particular, for $n = 8$, the number of such trees is approximately equal to 8.548×10^{326365}. We recalculate known values of the minimum depth for $n = 2, \ldots, 8$ and minimum average depth for $n = 2, \ldots, 7$ to make sure that the new results are valid.

To obtain these results, we use tools based on extensions of dynamic programming [2,5,9] which allow us to make sequential optimization of decision trees relative to depth and average depth, and to count the number of decision trees with minimum average depth. The considered algorithms are not, of-course, brute-force algorithms (it is impossible to work

* Corresponding author. Tel.: +966 12 808 0334; fax: +966 12 802 1291.
E-mail address: mikhail.moshkov@kaust.edu.sa (M. Moshkov).

http://dx.doi.org/10.1016/j.dam.2015.10.030
0166-218X/

Please cite this article in press as: H. AbouEisha, et al., Decision trees with minimum average depth for sorting eight elements, Discrete Applied Mathematics (2015), http://dx.doi.org/10.1016/j.dam.2015.10.030

Table 1
Results for sorting $n = 2, 3, 4, 5$ elements.

n	2	3	4	5
$h(P_n)$	1	3	5	7
$h_{avg}(P_n)$	2/2!	16/3!	112/4!	832/5!
$\varphi(n)/n!$	2/2!	16/3!	112/4!	832/5!
$\lvert Opt_{h_{avg}}(P_n)\rvert$	1	12	27, 744	2, 418, 647, 040
$\lvert\mathcal{P}(n)\rvert$	3	19	219	4231

directly with 8.548×10^{326365} optimal trees for $n = 8$). However, they require the work with big number of subproblems (for $n = 8$, the number of subproblems is equal to 431, 723, 379). We describe the notion of subproblems in Section 2. The construction of the directed acyclic graph (DAG) representing the ordering of subproblems is discussed in Section 3.1.

The functions provided by the system DAGGER depend on the number of nodes and edges of the DAG corresponding to the input problem. From empirical study, our system can work with DAGs of size around fifty million nodes.

The paper consists of four sections. Section 2 contains main results, Section 3 – description of tools, and Section 4 – short conclusions.

2. Main results

Let $x_1, \ldots, x_n$ be pairwise different elements from a linearly ordered set. We should find a permutation $(p_1, \ldots, p_n)$ from the set P_n of all permutations of the set $\{1, \ldots, n\}$ such that $x_{p_1} < \cdots < x_{p_n}$. Each nonempty subset Q of the set P_n can be considered as a subproblem of the initial sorting problem P_n with inputs $x_1, \ldots, x_n$ for each of which there exists a permutation $(p_1, \ldots, p_n) \in Q$ such that $x_{p_1} < \cdots < x_{p_n}$. We give all required definitions not for P_n but for an arbitrary subset (subproblem) Q of P_n.

We denote by $I(n)$ the set of all inequalities of the form $x_i < x_j$ such that $(i, j) \in \pi(n) = \{(i, j) : 1 \le i, j \le n, i \ne j\}$. We say that the permutation $p = (p_1, \ldots, p_n)$ is compatible with the inequality $x_i < x_j$ if and only if i precedes j in p. For $s_1, \ldots, s_m \in I(n)$, we denote by $Q(s_1)\ldots(s_m)$ the set of all permutations from Q which are compatible with all inequalities $s_1, \ldots, s_m$.

For solving the subproblem Q, we use binary decision trees in which terminal nodes are labeled with permutations from Q. Each nonterminal node is labeled with a comparison $x_i : x_j$ of two elements where $(i, j) \in \pi(n)$. Two edges start in this node which are labeled with results of the comparison $x_i < x_j$ and $x_j < x_i$, respectively.

We denote by $E(Q)$ the set of comparisons $x_i : x_j$ such that $(i, j) \in \pi(n)$, $Q(x_i < x_j) \ne \emptyset$ and $Q(x_j < x_i) \ne \emptyset$.

Let Γ be a decision tree and v be a node of Γ. We denote $Q(v) = Q(s_1)\ldots(s_m)$ where $s_1, \ldots, s_m$ are all inequalities attached to the edges in the path from the root of Γ to v (if v is the root of Γ then $Q(v) = Q$).

We will say that Γ solves the subproblem Q if each node v of Γ satisfies the following conditions:

1. If $|Q(v)| = 1$ and $Q(v) = \{p\}$ then v is a terminal node labeled with the permutation p;
2. If $|Q(v)| > 1$ then the node v is a nonterminal node which is labeled with a comparison $x_i : x_j$ from the set $E(Q(v))$.

We consider three cost functions for decision trees. Let Γ be a decision tree for solving the subproblem Q. We denote by $h(\Gamma)$ the depth of Γ which is the maximum length of a path from the root to a terminal node, by $l(\Gamma)$ – the external path length in Γ (the sum of lengths of all paths from the root to terminal nodes of Γ), and by $h_{avg}(\Gamma)$ – the average depth of Γ which is equal to $l(\Gamma)/|Q|$ (one can show that each decision tree for solving Q has $|Q|$ terminal nodes).

We denote by $h(Q)$ the minimum depth, by $l(Q)$ – the minimum external path length, and by $h_{avg}(Q)$ – the minimum average depth of decision trees for solving the subproblem Q. Note that $h_{avg}(Q) = l(Q)/|Q|$.

For $n = 2, \ldots, 8$, the values $h(P_n)$ and $h_{avg}(P_n) = l(P_n)/n!$ can be found in Tables 1 and 2. For the considered values of n, the parameter $h(P_n)$ is equal to its lower bound $\lceil \log_2(n!) \rceil$, and the parameter $h_{avg}(P_n)$ is equal to its lower bound $\varphi(n)/n!$ for $n = 2, \ldots, 6$.

Let Γ be a decision tree for solving the subproblem Q and ψ be one of the cost functions h, l, h_{avg}. We will say that Γ is optimal relative to ψ if $\psi(\Gamma) = \psi(Q)$. We denote by $Opt_\psi(Q)$ the set of decision trees for the subproblem Q which are optimal relative to ψ.

It is clear that $Opt_{h_{avg}}(P_n) = Opt_l(P_n)$. Based on results of computer experiments we obtain that $Opt_{h_{avg}}(P_n) \subseteq Opt_h(P_n)$ for $n = 2, \ldots, 8$. For $n = 2, \ldots, 8$, we count also the cardinality of the set $Opt_{h_{avg}}(P_n)$.

A nonempty subproblem $Q \subseteq P_n$ is called a separable subproblem of P_n if there exists a subset I of the set of inequalities $I(n)$ such that Q is the set of all permutations from P_n which are compatible with each inequality from I. In particular, $Q = P_n$ if $I = \emptyset$. We denote by $\mathcal{P}(n)$ the set of all separable subproblems of P_n. The cardinality of the set $\mathcal{P}(n)$ for $n = 1, \ldots, 8$ can be found in Tables 1 and 2.

All computations were done using our software system Dagger [1] for optimization of decision trees and rules. This system is based on extensions of dynamic programming that allow us to describe the set of all decision trees for the initial problem, to make sequential optimization relative to different cost functions and to count the number of optimal decision trees for some cost functions. The work of Dagger involves the construction and transformations of a directed acyclic graph