Stronger Tradeoffs for Orthogonal Range Querying in the Semigroup Model

Swaroop N. Prabhakar

The Institute of Mathematical Sciences, HBNI, Chennai, 600113, India
npswaroop@imsc.res.in

Vikram Sharma

The Institute of Mathematical Sciences, HBNI, Chennai, 600113, India
vikram@imsc.res.in

Abstract

In this paper, we focus on lower bounds for data structures supporting orthogonal range querying on m points in n-dimensions in the semigroup model. Such a data structure usually maintains a family of "canonical subsets" of the given set of points and on a range query, it outputs a disjoint union of the appropriate subsets. Fredman showed that in order to prove lower bounds in the semigroup model, it suffices to prove a lower bound on a certain combinatorial tradeoff between *two parameters*: (a) the total sizes of the canonical subsets, and (b) the total number of canonical subsets required to cover all query ranges. In particular, he showed that the arithmetic mean of these two parameters is $\Omega(m \log^n m)$. We strengthen this tradeoff by showing that the *geometric mean* of the same two parameters is $\Omega(m \log^n m)$.

Our second result is an alternate proof of Fredman's tradeoff in the one dimensional setting. The problem of answering range queries using canonical subsets can be formulated as factoring a specific boolean matrix as a product of two boolean matrices, one representing the canonical sets and the other capturing the appropriate disjoint unions of the former to output all possible range queries. In this formulation, we can ask what is an optimal data structure, i.e., a data structure that minimizes the sum of the two parameters mentioned above, and how does the balanced binary search tree compare with this optimal data structure in the two parameters? The problem of finding an optimal data structure is a non-linear optimization problem. In one dimension, Fredman's result implies that the minimum value of the objective function is $\Omega(m \log m)$, which means that at least one of the parameters has to be $\Omega(m \log m)$. We show that both the parameters in an optimal solution have to be $\Omega(m \log m)$. This implies that balanced binary search trees are near optimal data structures for range querying in one dimension. We derive intermediate results on factoring matrices, not necessarily boolean, while trying to minimize the norms of the factors, that may be of independent interest.

2012 ACM Subject Classification Information systems → Multidimensional range search

Keywords and phrases range querying, lower bounds, matrix factorization, Lagrange dual function

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2018.45

Acknowledgements We are thankful to Dr. Prashant Batra and Dr. Arijit Ghosh for pointing out the relevant literature on the eigenvalues of tridiagonal matrices, and range querying, respectively. We would also like to thank the anonymous refrees for their careful comments; in particular, one of the reviewers who spotted an error in our argument in Lemma 4.

38th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2018).
Editors: Sumit Ganguly and Paritosh Pandya; Article No. 45; pp. 45:1–45:14
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

Orthogonal range querying is one of the fundamental problems in computational geometry. The **range querying** problem is the following: Given a set X of m points in $\mathbb{R}^n$ and a *range* set $\mathcal{R}$ of subsets of points in $\mathbb{R}^n$, the goal is to pre-process the set X into a data structure so that given a query range $R \in \mathcal{R}$, the set of points in $X \cap R$ can be output efficiently. For *orthogonal range querying*, a range is simply an axis aligned box in $\mathbb{R}^n$. In this paper, we only consider the problem of orthogonal range querying. Sometimes, we are also interested in the number of points in the set $X \cap R$. The case where we output all the points in $X \cap R$ is called range reporting and the case where we only report the number of points in $X \cap R$ is called range counting. Other types of queries include whether or not $X \cap R$ is empty and so on. To capture these different types of queries in the range querying framework, it is typical to associate with every point $X_i \in X$ a weight w_i, where w_i comes from a commutative semigroup $(S, +)$[1]. Then, for every query range R, the output is $\sum_{X_i \in X \cap R} w_i$. For instance, for the orthogonal range reporting problem, we can take the semigroup $(2^X, \cup)$ and set $w_i = \{X_i\}$.; for the range counting problem, we can take the semigroup $(\mathbb{N}, +)$ and set $w_i = 1$.

Data structures for range querying typically store certain **canonical subsets** of the input set X and on a query range R, the query algorithm comes up with a set of disjoint canonical subsets such that their union is exactly $X \cap R$. The performance of a data structure for range querying is measured by the time spent in answering a query, the space requirement of the data structure and also the preprocessing cost involved in building the data structure. Often, the preprocessing is ignored as the data structure is built only once. In the dynamic setting where operations such as delete and insert are permitted, update time is also important. Most data structures for geometric problems are described in the real RAM model [16] and the pointer-machine model [1, 2]. A popular data structure for orthogonal range querying is the *range tree* which was introduced by Bentley [3]; for an exposition, see [4, chap. 5]. For orthogonal range reporting on m points in n dimensions, the range tree can be built in time $O(m \log^{n-1} m)$ and every query can be answered in time $O(\log^n m + k)$, where k is the number of points in the output. The query time though can be improved to $O(\log^{n-1} m + k)$ through a technique called fractional cascading [7, 14]. These upper bounds have been subsequently improved for range querying in various computation models [1, 2].

Fredman gave some of the first lower bounds on orthogonal range querying in the semigroup model [10, 11]. These lower bounds are in the dynamic setting where insertions and deletions are allowed. More specifically, in [11], he showed that for any m, there is a sequence of m operations consisting of insert, delete and querying such that the time required for this sequence is $\Omega(m \log^n m)$ in the semigroup model. The crux of Fredman's lower bound argument lies in exploiting a certain combinatorial tradeoff between the sizes of the canonical sets and the number of canonical sets needed to answer all the query ranges [15, p. 69, Lemma 9]. To state this more precisely, we set up some definitions and notations.

From here on, we take the set X to be the n-dimensional grid of m points, i.e.,

$$X := \left\{1, \ldots, \left\lfloor m^{1/n} \right\rfloor\right\}^n .$$

The n coordinates of the point X_i are represented as X_{ij}, $j = 1, \ldots, n$. A **one-sided range**

[1] Another algebraic structure from which weights are assigned are groups [12, 9], but in this paper we restrict ourselves to the case where weights come from a semigroup.

query on the set X takes as input a $Y \in \mathbb{R}^n$ and outputs

$$R_Y := \{X_i \in X : X_i \leq Y\},$$

where $X_i \leq Y$ iff $X_{ij} \leq Y_j$, for all $j \in [n]$. In this paper, range queries will always be one-sided. Corresponding to m points in X, we have m range queries whose outputs are $R_j := R_{X_j}$, for $j \in [m]$. A set $\mathcal{D} := \{W_1, \dots, W_r\}$ of subsets of X is a **data structure for answering range queries on** X if every output to a range query on X is represented as a *disjoint union* over the canonical sets $W_1, \dots, W_r$. Let $\langle R_j \rangle_{\mathcal{D}}$ denote the set of indices of W_k's used in the representation of R_j. Fredman showed the following result:

▶ **Proposition 1.** *If $\mathcal{D}$ is a data structure that answers range queries on X then*

$$\sum_{k=1}^{r} |W_k| + \sum_{j=1}^{m} |\langle R_j \rangle_{\mathcal{D}}| = \Omega(m \log^n m).$$

This tradeoff between the sizes of canonical sets and the number of canonical sets needed for covering all the query ranges is the central theme of this paper. Many more lower bounds on orthogonal range querying in different computation models are also known (see [1, p. 7] and [2, p. 11]).

The tradeoff in Proposition 1 gives us a lower bound on the arithmetic mean of the total size of the canonical sets and the number of canonical sets needed for covering query ranges. But in practice, data structures such as range trees need $\Theta(m \log^n m)$ many canonical sets for orthogonal range querying on m points and the total size of these sets is $\Theta(m \log^n m)$ as well. In view of this fact, we prove the following stronger result:

▶ **Theorem 2.** *If $\mathcal{D}$ is a data structure that answers range queries on X then*

$$\left(\sum_{k=1}^{r} |W_k|\right)\left(\sum_{j=1}^{m} |\langle R_j \rangle_{\mathcal{D}}|\right) = \Omega(m^2 \log^{2n} m).$$

From the AM-GM inequality it is clear that Theorem 2 implies Proposition 1. Theorem 2 also implies that any data structure $\mathcal{D}$ that is *tight* with respect to Proposition 1, i.e.,

$$\sum_{k=1}^{r} |W_k| + \sum_{j=1}^{m} |\langle R_j \rangle_{\mathcal{D}}| = \Theta(m \log^n m), \tag{1}$$

must satisfy:

$$\sum_{k=1}^{r} |W_k| = \Theta(m \log^n m) \text{ and } \sum_{j=1}^{m} |\langle R_j \rangle_{\mathcal{D}}| = \Theta(m \log^n m). \tag{2}$$

The proof of Theorem 2 will be given in Section 2.

From (2), we see that the balanced binary search tree is an optimal data structure in the *boolean* setting where the outputs are represented as disjoint unions over canonical sets. This leaves open the possibility of existence of a more efficient data structure that does not take disjoint unions of its canonical subsets but takes their weighted sum in order to represent an output. In such a relaxed setting, Proposition 1 and Theorem 2 are not applicable. Can balanced binary search tree be an optimal data structure even in this setting? In Section 3, we give a positive answer to this question.

In order to account for data structures that take weighted sums of their canonical sets, we will reinterpret range querying differently from Proposition 1. In the proof of Proposition 1 [15, p. 69, Lemma 9 and 10], the problem of range querying is interpreted in a graph theoretic setting, namely expressing a bipartite graph as a "product" of two bipartite graphs. This can also be interpreted in terms of matrices [6, Sec. 2.2]. Let $U_{m\times r}$ be the incidence matrix of the set X with the canonical sets W_k's, i.e, $U_{ik} = 1$ iff $X_i \in W_k$. Similarly, define $V_{r\times m}$ to be the incidence matrix of the canonical sets W_k's and the outputs R_j's. Let $R_{m\times m}$ be the matrix whose columns are the characteristic vectors of the sets R_j's. To give a proof of Proposition 1, it suffices to derive a lower bound on the optimal value of the following optimization problem:

$$\min\left(||U||_F^2 + ||V||_F^2\right) \text{ subject to } UV = R, \tag{3}$$

where $R \in \{0,1\}^{m\times m}$, $U \in \{0,1\}^{m\times r}$ and $V \in \{0,1\}^{r\times m}$ and $||.||_F$ refers to the Frobenius norms of the respective matrices.

In this optimization based formulation of the problem, the objective function aims to minimize the sum of the two parameters we are interested in: The total size of the canonical sets, $||U||_F^2$ and the total number of canonical sets needed to cover all the query ranges, $||V||_F^2$. Every data structure that supports range querying in one dimension is a feasible solution to the problem above. When the entries of the matrices are restricted to be boolean, Proposition 1 implies that the optimal value of the objective function is $\Omega(m \log m)$. Hence, from (2), we see that for an optimal solution $(U_{\text{bool}}, V_{\text{bool}})$ of (3) we must have,

$$||U_{\text{bool}}||_F^2 = \Theta(m \log m) \text{ and } ||V_{\text{bool}}||_F^2 = \Theta(m \log m).$$

To extend these bounds for data structures that take weighted sums of their canonical sets, we consider the relaxation of the problem in (3) where the matrix entries are allowed to be arbitrary reals. For an optimal solution (U^*, V^*) of this relaxation, we show that

$$||U^*||_F^2 = \Omega(m \log m) \text{ and } ||V^*||_F^2 = \Omega(m \log m).$$

The lower bounds above imply that the balanced binary search tree is near optimal **not only** in the boolean framework but also in a more relaxed setting.

The main idea in proving the lower bounds above is to use the Lagrangian dual of the relaxation and show that

$$||U^*||_F^2 = \text{Trace}((R^t R)^{1/2}) \text{ and } ||V^*||_F^2 = \text{Trace}((R^t R)^{1/2}), \tag{4}$$

where we take the principal square-root of a matrix [13]. The result in (4) holds for an **arbitrary** matrix R (see Theorem 3) and is the key technical ingredient in our proof. Then, by taking R to be the lower triangular all ones matrix, which corresponds to range querying in one dimension, and by using some well established results on explicit forms for the eigenvalues of tri-diagonal matrix (in this case $(R^t R)^{-1}$), we show that

$$\text{Trace}((R^t R)^{1/2}) = \Omega(m \log m).$$

We believe that our proof gives more understanding on the optimality of Fredman's lower bound by relating it to some intrinsic parameters of the matrix R, which is a natural representation of the range query problem in one-dimension. To the best of our knowledge, our proof is more general than the existing proofs in the literature; e.g., [17] works only in the boolean setting. Whether our proof technique can be generalized to obtain an alternative proof of Proposition 1 in all dimensions remains an open and interesting question.

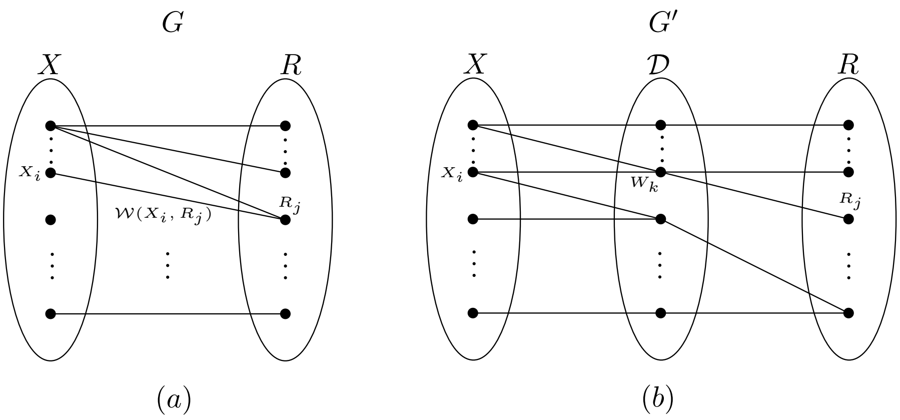

Figure 1 (a): Bipartite graph G with the vertex sets X, R and the edge set E. (b): Tripartite graph G' with vertex sets X, $\mathcal{D}$ and R.

2 Tradeoff between Sizes of Canonical Sets and Outputs to Query Ranges

In this section, we will prove Theorem 2. The proof of the theorem follows by altering the argument in the proof of Proposition 1. Before we proceed, we first present some high level details of the proof of Proposition 1. For the complete details, we refer to [15, p. 69].

The argument relies on interpreting range querying in a graph theoretic setting: Consider the weighted bipartite graph $G(X \cup R, E)$, where $R := \{R_1, R_2, \ldots, R_m\}$ and the edge set $E := \{(X_i, R_j) : X_i \in R_j\}$; see Figure 1(a) for illustration. The edge $(X_i, R_j) \in E$ is assigned the weight

$$\mathcal{W}(X_i, R_j) := \frac{1}{\prod_{\kappa=1}^{n} (X_{j\kappa} - X_{i\kappa} + 1)}. \tag{5}$$

The graph G can be "factored" into a tripartite graph G' whose vertex set is $\{X \cup \mathcal{D} \cup R\}$. There is an edge (X_i, W_k) iff $X_i \in W_k$ and there is an edge (W_k, R_j) iff $k \in \langle R_j \rangle_{\mathcal{D}}$; see Figure 1(b) for an illustration. Note that the edges of G are a disjoint union over the sets $\{(X_i, R_j) : X_i \in W_k, k \in \langle R_j \rangle_{\mathcal{D}}\}$, for all $k \in [r]$, as every R_j is a disjoint union of W_k's. For every set W_k, define

$$I_k := \{X_i \in X : X_i \in W_k\}$$

i.e., the edges of G' incident on W_k from the left and

$$O_k := \{R_j \in R : k \in \langle R_j \rangle_{\mathcal{D}}\},$$

i.e., the edges of G' incident on W_k from the right. Therefore,

$$|I_k| = |W_k| \text{ and } \sum_{j=1}^{m} |\langle R_j \rangle_{\mathcal{D}}| = \sum_{k=1}^{r} |O_k|.$$

At a high level, the proof of Proposition 1 can be broken down into two steps [15, p. 69, Lemma 9 and p. 71, Lemma 10]:

Step 1. For every $k \in [r]$,

$$\sum_{\substack{X_i \in I_k \\ R_j \in O_k}} \mathcal{W}(X_i, R_j) = \sum_{\substack{X_i \in I_k \\ R_j \in O_k}} \frac{1}{\prod_{\kappa=1}^{n} (X_{j\kappa} - X_{i\kappa} + 1)} \leq (2\pi)^n (|I_k| + |O_k|), \tag{6}$$

where n is the dimension of the points in X. The outline of the proof is as follows. Let $M_j = \max\{X_{ij} : X_i \in I_k\}$. Define the **associate sets** of W_k as

$$B := \{(M_1 - X_{i1}, M_2 - X_{i2}, \ldots, M_n - X_{in}) : X_i \in I_k\}$$

and

$$C := \{(X_{j1} - M_1, X_{j2} - M_2, \ldots, X_{jn} - M_n) : R_j \in O_k\}.$$

Since every term of the form

$$(X_{j\kappa} - X_{i\kappa} + 1) = (M_\kappa - X_{i\kappa} + X_{j\kappa} - M_\kappa + 1),$$

the summation in (6) is equal to

$$\sum_{\substack{X_i \in I_k \\ R_j \in O_k}} \frac{1}{\prod_{\kappa=1}^{n} (X_{j\kappa} - X_{i\kappa} + 1)} = \sum_{\substack{u \in B \\ v \in C}} \frac{1}{\prod_{\kappa=1}^{n} (u_\kappa + v_\kappa + 1)},$$

where u_κ and v_κ are non-negative integers. Then, by applying a generalized version of Hilbert's inequality for points with natural numbers as their coordinates, one obtains

$$\sum_{\substack{u \in B \\ v \in C}} \frac{1}{\prod_{\kappa=1}^{n} (u_\kappa + v_\kappa + 1)} \leq (2\pi)^n (|B| + |C|) = (2\pi)^n (|I_k| + |O_k|). \tag{7}$$

Step 2. The second step is to show that the total sum of weights over all edges in E satisfies

$$\sum_{(X_i, R_j) \in E} \mathcal{W}(X_i, R_j) = \sum_{k=1}^{r} \sum_{\substack{X_i \in I_k \\ R_j \in O_k}} \mathcal{W}(X_i, R_j) = \Omega(m \log^n m). \tag{8}$$

By summing the inequality in (6) over all W_k and applying the lower bound from (8), we get the claim in Proposition 1.

We now give the proof of Theorem 2.

Proof. Since $\sum_{k=1}^{r} |W_k| = \sum_{k=1}^{r} |I_k|$ and $\sum_{j=1}^{m} |\langle R_j \rangle_{\mathcal{D}}| = \sum_{k=1}^{r} |O_k|$, we will show that

$$\sum_{k=1}^{r} |I_k| \cdot \sum_{k=1}^{r} |O_k| = \Omega(m^2 \log^{2n} m).$$

To prove this, consider a pair of canonical sets, W_k and W_ℓ. Using the same weight function as in (5) on the edges of G, we have

$$\sum_{\substack{X_i \in I_k \\ R_j \in O_k}} \sum_{\substack{X_c \in I_\ell \\ R_d \in O_\ell}} \mathcal{W}(X_i, R_j) \cdot \mathcal{W}(X_c, R_d) = \sum_{\substack{X_i \in I_k \\ R_j \in O_k}} \sum_{\substack{X_c \in I_\ell \\ R_d \in O_\ell}} \frac{1}{\prod_{\kappa=1}^{n} (X_{j\kappa} - X_{i\kappa} + 1)(X_{d\kappa} - X_{c\kappa} + 1)}. \tag{9}$$

Define the **associate sets** B, C of W_k as in Step 1; sets B' and C' are defined analogously for W_ℓ. Notice that $|B| = |I_k|$, $|C| = |O_k|$, $|B'| = |I_\ell|$ and $|C'| = |O_\ell|$. Using the associate sets, equation (9) can be expressed as

$$\sum_{\substack{X_i \in I_k \\ R_j \in O_k}} \sum_{\substack{X_c \in I_\ell \\ R_d \in O_\ell}} \mathcal{W}(X_i, R_j) \cdot \mathcal{W}(X_c, R_d) = \sum_{\substack{u \in B \\ v \in C}} \sum_{\substack{u' \in B' \\ v' \in C'}} \frac{1}{\prod_{\kappa=1}^{n} (u_\kappa + v_\kappa + 1)(u'_\kappa + v'_\kappa + 1)}. \quad (10)$$

The importance of the associate sets of W_k and W_ℓ is that they have non-negative coordinates and they are in some sense independent of the actual coordinates of the points in X, since difference choices of the point set X give the same associate sets. We now use the upper bound from (7) to upper bound the RHS of (10). Since the RHS of (10) can be interpreted as a function over $2n$ dimensional points, we define the following sets in $\mathbb{R}^{2n}$

$$\mathcal{B} := \{(u, u') : u \in B, u' \in B'\}, \mathcal{C} := \{(v, v') : v \in C, v' \in C'\}$$

and

$$\mathcal{B}' := \{(u, v') : u \in B, v' \in C'\}, \mathcal{C}' := \{(v, u') : v \in C, u' \in B'\}.$$

The pair of sets $(\mathcal{B}, \mathcal{C})$ and $(\mathcal{B}', \mathcal{C}')$ allow us to express the RHS of (10) in two different ways as:

$$\sum_{\substack{u \in B \\ v \in C}} \sum_{\substack{u' \in B' \\ v' \in C'}} \frac{1}{\prod_{\kappa=1}^{n} (u_\kappa + v_\kappa + 1)(u'_\kappa + v'_\kappa + 1)} = \sum_{\substack{\mathcal{U} \in \mathcal{B} \\ \mathcal{V} \in \mathcal{C}}} \frac{1}{\prod_{\kappa=1}^{2n} (\mathcal{U}_\kappa + \mathcal{V}_\kappa + 1)}$$
$$= \sum_{\substack{\mathcal{U}' \in \mathcal{B}' \\ \mathcal{V}' \in \mathcal{C}'}} \frac{1}{\prod_{\kappa=1}^{2n} (\mathcal{U}'_\kappa + \mathcal{V}'_\kappa + 1)}.$$

From (7), the RHS of the equalities above can be upper bounded as

$$\sum_{\substack{\mathcal{U} \in \mathcal{B} \\ \mathcal{V} \in \mathcal{C}}} \frac{1}{\prod_{\kappa=1}^{2n} (\mathcal{U}_\kappa + \mathcal{V}_\kappa + 1)} \leq (2\pi)^{2n} (|\mathcal{B}| + |\mathcal{C}|) \text{ and } \sum_{\substack{\mathcal{U}' \in \mathcal{B}' \\ \mathcal{V}' \in \mathcal{C}'}} \frac{1}{\prod_{\kappa=1}^{2n} (\mathcal{U}'_\kappa + \mathcal{V}'_\kappa + 1)} \leq (2\pi)^{2n} (|\mathcal{B}'| + |\mathcal{C}'|).$$

So, from the two inequalities above and (10) we obtain

$$\sum_{\substack{X_i \in I_k \\ R_j \in O_k}} \sum_{\substack{X_c \in I_\ell \\ R_d \in O_\ell}} \mathcal{W}(X_i, R_j) \cdot \mathcal{W}(X_c, R_d) \leq (2\pi)^{2n} \min\{|\mathcal{B}| + |\mathcal{C}|, |\mathcal{B}'| + |\mathcal{C}'|\}.$$

$$= (2\pi)^{2n} \min\{|B||B'| + |C||C'|, |B||C'| + |B'||C|\},$$
$$= (2\pi)^{2n} \min\{|I_k||I_\ell| + |O_k||O_\ell|, |I_k||O_\ell| + |I_\ell||O_k|\}$$

Therefore, for an arbitrary pair W_k, W_ℓ, we have

$$\sum_{\substack{X_i \in I_k \\ R_j \in O_k}} \sum_{\substack{X_c \in I_\ell \\ R_d \in O_\ell}} \mathcal{W}(X_i, R_j) \cdot \mathcal{W}(X_c, R_d) \leq (2\pi)^{2n} (|I_k||O_\ell| + |I_\ell||O_k|). \quad (11)$$

Note that every edge (X_i, R_j) in E maps to a unique path (X_i, W_k, R_j) in the graph G'. Hence the sum of the LHS of (11) over all possible pairs of W_k and W_ℓ gives us the sum of the product of weights of all possible pairs of edges (X_i, R_j) and (X_c, R_d) in E. Hence from (8) we obtain that

$$\sum_{\substack{W_k \\ W_\ell}} \sum_{\substack{X_i \in I_k \\ R_j \in O_k}} \sum_{\substack{X_c \in I_\ell \\ R_d \in O_\ell}} \mathcal{W}(X_i, R_j) \cdot \mathcal{W}(X_c, R_d) = \sum_{\substack{(X_i, R_j) \in E \\ (X_c, R_d) \in E}} \mathcal{W}(X_i, R_j) \cdot \mathcal{W}(X_c, R_d) = \Omega(m^2 \log^{2n} m).$$

(12)

Similarly, summing the RHS of (11) over all pairs of W_k and W_ℓ and using the fact that $|I_k| = |W_k|$ and $\sum_{k=1}^{r} |O_k| = \sum_{j=1}^{m} | \langle R_j \rangle_{\mathcal{D}} |$, we get

$$\sum_{W_k}\sum_{W_\ell}(2\pi)^{2n}(|I_k||O_\ell| + |I_\ell||O_k|) = 2\cdot(2\pi)^{2n}\left(\sum_{j=1}^{m} | \langle R_j \rangle_{\mathcal{D}} |\right)\left(\sum_{k=1}^{r}|W_k|\right).$$

Therefore, from (11), (12) and the equality above, we conclude that

$$\left(\sum_{j=1}^{m} | \langle R_j \rangle_{\mathcal{D}} |\right)\left(\sum_{k=1}^{r}|W_k|\right) = \Omega(m^2 \log^{2n} m).$$ ◀

3 Optimality of the Balanced Binary Search Tree

In this section, we will show the optimality of the balanced binary search tree in a relaxed framework where the data structures are allowed to take a weighted sum of their canonical subsets. Let $X := \{1, 2, \ldots, m\}$. We again consider the set of one sided range queries: For $j \in X$, output $R_j := \{i \in X : i \le j\}$. Let $\mathcal{D} := \{W_1, W_2, \ldots, W_r\}$ be an arbitrary data structure that answers range queries on X. Proposition 1 in one dimension reduces to:

$$\left(\sum_{k=1}^{r}|W_k|\right) + \left(\sum_{j=1}^{m} | \langle R_j \rangle_{\mathcal{D}} |\right) = \Omega(m \log m). \tag{13}$$

To extend the lower bound above for data structures that are allowed to take weighted sums of their canonical subsets, we will reinterpret range querying in a different setting. The problem of range querying can be interpreted in a linear algebraic setting as follows: Consider the 0/1 matrix U whose rows are indexed by the m numbers and columns are indexed by the r sets, W_k's. The (i, j)th entry is one iff the number i is a member of W_j. Consider the range query that asks for all the numbers less than or equal to the jth number. The output is a union of at most, say ℓ sets. Then, R_j, which is an m-dimensional vector with ones from the jth position onwards can be expressed as a linear combination of at most ℓ columns of U. Let $\mathbf{v}_j$ be this linear combination, i.e.

$$R_j = U\mathbf{v}_j$$

where $\mathbf{v}_j$ is a 0/1 vector. Since there are m distinct range queries, we have $\mathbf{v}_1, \ldots, \mathbf{v}_m$ such vectors. If V is the matrix with these vectors as its columns, then our observation regarding these m range queries can be succinctly represented by the following matrix equation

$$UV = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 1 & 1 & 1 & \cdots & 1 \end{bmatrix} =: R, \tag{14}$$

where R is the lower triangular matrix with all ones on and below the diagonal and the rows of R are indexed by the numbers $m, m-1, \ldots, 1$ and the columns by $R_m, R_{m-1}, \ldots, R_1$.

Also, $U \in \{0,1\}^{m\times r}$ and $V \in \{0,1\}^{r\times m}$. Now,

$$\sum_{i=1}^{r} |W_i| = ||U||_F^2 \quad \text{and} \quad \sum_{j=1}^{m} |\langle R_j\rangle| = ||V||_F^2,$$

where $||A||_F$ denotes the Frobenius norm of the matrix A. So, in terms of factorizations of R as a product of U and V, proving the claim in (13) is equivalent to lower bounding the optimal value of the following optimization problem

$$\begin{aligned} &\min\left(||U||_F^2 + ||V||_F^2\right) \\ &\text{subject to } UV = R, U \in \{0,1\}^{m\times r}, V \in \{0,1\}^{r\times m}. \end{aligned} \tag{15}$$

In order to consider data structures that may take weighted sum of their canonical subsets instead of disjoint unions, we focus on the following relaxation of the problem in (15): Given an arbitrary matrix $T \in \mathbb{R}^{m\times m}$,

$$\begin{aligned} &\min\left(||U||_F^2 + ||V||_F^2\right) \\ &\text{subject to } UV = T, U \in \mathbb{R}^{m\times r}, V \in \mathbb{R}^{r\times m}. \end{aligned} \tag{16}$$

It is clear that a lower bound on the optimal value of (16), when T is taken to be R, is also a lower bound on the optimal value of (15). So, we first prove that

▶ **Theorem 3.** *Any optimal solution (U^*, V^*) for the optimization problem in (16) satisfies:*

$$||U^*||_F^2 = \text{Trace}((T^tT)^{1/2}) \text{ and } ||V^*||_F^2 = \text{Trace}((T^tT)^{1/2}),$$

where the matrix $(T^tT)^{1/2}$ is defined to be the principal square root of the matrix T^tT [13, p .20, Theorem 1.29].

Proof. The Lagrangian dual function associated with the problem in (16) is defined as [5, p. 216]

$$\inf_{U,V} L(U,V,\Lambda) = \inf_{U,V}\left(||U||_F^2 + ||V||_F^2 + \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{r}(T_{ij} - U_{ik}\cdot V_{kj})\lambda_{ij}\right), \tag{17}$$

where $\Lambda \in \mathbb{R}^{m\times m}$. The Lagrange dual problem is now defined as

$$\max_{\Lambda}\left(\inf_{U,V} L(U,V,\Lambda)\right), \tag{18}$$

where $\Lambda \in \mathbb{R}^{m\times m}$. Any optimal solution (U^*, V^*) for the primal problem satisfies the following inequality:

$$||U^*||_F^2 + ||V^*||_F^2 \geq \max_{\Lambda}\left(\inf_{U,V} L(U,V,\Lambda)\right).$$

From the inequality above, we see that it suffices to lower bound the optimal value of the dual problem in order to prove the required claim. Consider the function

$$\inf_{U,V} L(U,V,\Lambda).$$

Applying the optimality conditions, we take the partial derivative of $L(U,V,\Lambda)$ with respect to variables in U to get the following matrix equation:

$$\nabla_U L(U,V,\Lambda) = 2U^t - V\Lambda^t = 0. \tag{19}$$

Similarly, taking derivative with respect to variables in V, we get

$$\nabla_V L(U, V, \Lambda) = 2V - U^t\Lambda = 0. \tag{20}$$

From (19) and (20), we have

$$V\Lambda^t = 2U^t \quad \text{and} \quad U^t\Lambda = 2V. \tag{21}$$

Since the dual problem is convex and aims to maximize the Lagrangian dual function in (17) with respect to Λ, we apply the first order condition to $L(U, V, \Lambda)$ with respect to Λ to get

$$UV = T.$$

By left multiplying the first equation in (21) by U, we get

$$\begin{aligned} UV\Lambda^t &= 2UU^t \\ T\Lambda^t &= 2UU^t \text{ since } UV = T. \end{aligned}$$

Using the equality above and the fact that $||U||_F^2 = \text{Trace}(UU^t)$, we get

$$||U||_F^2 = \frac{1}{2}\text{Trace}(T\Lambda^t).$$

Similarly, we can show that

$$||V||_F^2 = \frac{1}{2}\text{Trace}(\Lambda^t T).$$

Therefore, for an optimal solution Λ of the dual problem, we have

$$||U||_F^2 = \frac{1}{2}\text{Trace}(T\Lambda^t), \quad ||V||_F^2 = \frac{1}{2}\text{Trace}(\Lambda^t T).$$

Since $\text{Trace}(T\Lambda^t) = \text{Trace}(\Lambda^t T)$, it suffices to show that the trace of $\Lambda^t T$ is $2 \cdot \text{Trace}((T^t T)^{1/2})$ to prove the theorem.

We begin by multiplying the transpose of the second equation in (21) with the first equation in (21) to obtain

$$\Lambda^t UV\Lambda^t = 4(UV)^t.$$

Since $UV = T$, we see that any optimal solution for the dual problem must satisfy:

$$\Lambda^t T\Lambda^t = 4T^t.$$

Multiplying the equality above by T from the right, we get

$$(\Lambda^t T)^2 = 4T^t T.$$

Since $T^t T$ is a positive semidefinite matrix [2], it is diagonalizable. Assuming Q to be the $m \times m$ matrix whose columns are the eigenvectors of $T^t T$ and $\gamma_1 \geq \gamma_2 \geq \cdots \geq \gamma_m$ to be the eigenvalues of $T^t T$, we can express the equality above as

$$(\Lambda^t T)^2 = 4Q^{-1}\Gamma Q$$

[2] Here we use the fact that any matrix A that can be written as $A = B^t B$, is positive semidefinite.

where Γ is the diagonal matrix with γ_k's being the kth diagonal entry. Therefore, we have

$$(\Lambda^t T) = 2Q^{-1}\Gamma^{1/2}Q,$$

where $Q^{-1}\Gamma^{1/2}Q$ is defined to be the *principle square root of* T^tT whose eigenvalues are $\sqrt{\gamma_1}, \sqrt{\gamma_2}, \dots, \sqrt{\gamma_m}$ and for all k, $\sqrt{\gamma_k} \in \mathbb{R}_{\geq 0}$. So,

$$\mathrm{Trace}(\Lambda^t T) = 2\mathrm{Trace}((T^tT)^{1/2}) = 2\sum_{k=1}^{m}\sqrt{\gamma_k}.$$

Hence, we conclude that

$$||U^*||_F^2 = \mathrm{Trace}((T^tT)^{1/2}) \quad ||V^*||_F^2 = \mathrm{Trace}((T^tT)^{1/2}).$$ ◀

We bound the trace of $(R^tR)^{1/2}$ in the following result:

▶ **Lemma 4.** *Let R be the matrix as in* (14)*. The trace of the principal square root of R^tR satisfies*

$$\mathrm{Trace}((R^tR)^{1/2}) = \sum_{k=1}^{m}\sqrt{\gamma_k} = \Omega(m\log m),$$

where γ_k, $k \in [m]$, are the eigenvalues of the matrix R^tR.

Proof. To compute γ_k's, consider the inverse matrix

$$(R^tR)^{-1} = R^{-1}(R^{-1})^t.$$

The inverse of the matrix R is the bidiagonal matrix

$$R^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 \\ -1 & 1 & 0 & 0 & \dots & 0 \\ 0 & -1 & 1 & 0 & \dots & 0 \\ 0 & 0 & -1 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 \end{bmatrix}$$

So, the matrix $R^{-1}(R^{-1})^t$ is the following tridiagonal matrix

$$R^{-1}(R^{-1})^t = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & \dots & 0 \\ -1 & 2 & -1 & 0 & 0 & \dots & 0 \\ 0 & -1 & 2 & -1 & 0 & \dots & 0 \\ 0 & 0 & -1 & 2 & -1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \dots & -1 \\ 0 & 0 & 0 & 0 & 0 & \dots & 2 \end{bmatrix},$$

which obtained as a special case of tridiagonal matrices of the form

$$\begin{bmatrix} a+d & b & 0 & 0 & 0 & \dots & 0 \\ b & a & b & 0 & 0 & \dots & 0 \\ 0 & b & a & b & 0 & \dots & 0 \\ 0 & 0 & b & a & b & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \dots & b \\ 0 & 0 & 0 & 0 & 0 & \dots & a+c \end{bmatrix},$$

by substituting $a = 2$, $b = -1$, $d = -1$, and $c = 0$. From [8, p. 27], we know that the roots of the characteristic polynomial of the matrix above is given by

$$a + 2b\cos\theta \tag{22}$$

where θ varies over the m zeros of the following function

$$\frac{\sin(m+1)\theta - \frac{c+d}{b}\sin m\theta + \frac{cd}{b^2}\sin(m-1)\theta}{\sin\theta}.$$

Substituting $a = 2$, $b = d = -1$, and $c = 0$ in the formula above, we obtain the following expression

$$\frac{\sin(m+1)\theta - \sin m\theta}{\sin\theta}.$$

Simplifying the formula above using the sum-to-product identity[3], we get

$$\frac{2\sin\frac{(m+1)\theta - m\theta}{2}\cos\frac{(m+1)\theta + m\theta}{2}}{\sin\theta} = \frac{2\sin(\theta/2)\cos(m\theta+\theta/2)}{2\sin(\theta/2)\cos(\theta/2)} = \frac{\cos(m\theta+\theta/2)}{\cos(\theta/2)}.$$

The expression above vanishes at the values

$$\left(\frac{2k-1}{2m+1}\right)\pi,$$

where $k = 1, 2, \ldots, m$. Substituting in (22), we see that the eigenvalues of $R^{-1}(R^{-1})^t$ are

$$2\left(1 - \cos\left(\frac{2k-1}{2m+1}\right)\pi\right) = 4\sin^2\left(\frac{2k-1}{4m+2}\right)\pi,$$

for $k = 1, 2, \ldots, m$, where we use the identity $(1 - \cos x) = 2\sin^2 x/2$ above. So, the eigenvalues of R^tR are

$$\gamma_k = \frac{1}{4\sin^2\left(\frac{2k-1}{4m+2}\right)\pi},$$

for $k = 1, 2, \ldots, m$, and, therefore, the trace of the principal square root of R^tR is

$$\text{Trace}((R^tR)^{1/2}) = \sum_{k=1}^{m}\sqrt{\gamma_k} = \sum_{k=1}^{m}\frac{1}{2\sin\left(\frac{2k-1}{4m+2}\right)\pi}.$$

Since for $k = 1, \ldots, m$, the reciprocal of the sine functions is a monotonically decreasing concave function, the summation above can be lower bounded as follows:

$$\text{Trace}((R^tR)^{1/2}) = \sum_{k=1}^{m}\frac{1}{2\sin\frac{(2k-1)\pi}{4m+2}} \geq \int_1^m \frac{dx}{2\sin\frac{(2x-1)\pi}{4m+2}} = \int_1^m \frac{\csc\frac{(2x-1)\pi}{4m+2}}{2}dx.$$

Substituting $y = \frac{(2x-1)\pi}{4m+2}$ and using the fact that $\int \csc y \cdot dy = \ln|\tan(y/2)|$, we obtain

$$\text{Trace}((R^tR)^{1/2}) \geq \frac{4m+2}{4\pi}\left(\ln\left(\tan\frac{(2m-1)\pi}{(8m+4)}\right) - \ln\left(\tan\frac{\pi}{(8m+4)}\right)\right).$$

---

[3] Namely, $\sin A - \sin B = 2\sin\frac{(A-B)}{2}\cos\frac{(A+B)}{2}$

As m tends to infinity, the term $\tan((2m-1)\pi/(8m+4))$ tends to one. Therefore, we have

$$\text{Trace}((R^tR)^{1/2}) = \Omega\left(m \ln\left(\cot \frac{\pi}{(8m+4)}\right)\right).$$

From the Taylor series of the cotangent function, we know that for $m \geq 1$, $\cot \frac{\pi}{(8m+4)} = \Theta(m)$. Therefore,

$$\text{Trace}((R^tR)^{1/2}) = \Omega(m \log m). \qquad \blacktriangleleft$$

We note that from Theorem 3 and Lemma 4, we have a stronger conclusion than in (13). From (13), we can only infer that one of the two parameters is $\Omega(m \log m)$ whereas for data structures such as balanced binary search trees, both the parameters are $\Theta(m \log m)$. Our proof shows that

$$||U_{\text{BST}}||_F = \Theta(||U^*||_F) \text{ and } ||V_{\text{BST}}||_F = \Theta(||V^*||_F),$$

where (U^*, V^*) is an optimal solution for the problem in (15) and $(U_{\text{BST}}, V_{\text{BST}})$ are the matrices U and V corresponding to the balanced binary search tree. Therefore, binary search trees are optimal with respect to both the parameters, The total size of the canonical sets and the total number of canonical sets needed for covering all the query ranges. Also, our proof implies that balanced binary search trees are near optimal in a more relaxed framework where the data structures are allowed to take weighted sums of their canonical subsets.

4 Conclusion

In this paper, we have shown that there is a stronger tradeoff between the sizes of canonical sets and the outputs to query ranges than the one shown by Fredman. In Section 3, we show the optimality of balanced binary search trees in a more general setting where we allow weighted combinations of the canonical sets in the data structure. A natural continuation would be to generalize this proof to higher dimensions. One can start by bounding the integrality gap between an optimal solution in the boolean setting and an optimal solution of the relaxation. In one dimension, our proof shows that this gap is at most a constant.

We also believe that the optimization problem introduced in Section 3 is interesting in its own right. For instance, the lower bound for the average complexity of the partial sums problem [12] in the one dimensional setting can be obtained from the lower bound on the optimization problem in (15).

References

1. Pankaj Agarwal. Range Searching. Technical report, Duke University, 2016. URL: `https://users.cs.duke.edu/~pankaj/publications/surveys/rs3ed.pdf`.
2. Pankaj Agarwal and Jeff Erickson. Geometric Range Searching and its Relatives. Technical report, University of Illinois, 1997. URL: `http://jeffe.cs.illinois.edu/pubs/pdf/survey.pdf`.
3. Jon Louis Bentley. Multidimensional Divide-and-Conquer. *Commun. ACM*, 23(4):214–229, April 1980. `doi:10.1145/358841.358850`.
4. Mark de Berg, Otfried Cheong, Marc van Kreveld, and Mark Overmars. *Computational Geometry: Algorithms and Applications*. Springer-Verlag TELOS, Santa Clara, CA, USA, 3rd ed. edition, 2008.
5. Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.

6 Richard A. Brualdi and Dragos Cvetkovic. *A Combinatorial Approach to Matrix Theory and Its Applications*. Chapman and Hall/CRC, 1st edition, 2008.

7 Bernard Chazelle and Leonidas J. Guibas. Fractional Cascading: A Data Structuring Technique. *Algorithmica*, 1(1):133–162, November 1986. doi:10.1007/BF01840440.

8 J. F. Elliott. The characteristic roots of certain real symmetric matrices. Master's thesis, University of Tennessee, 1953.

9 M. Fredman and M. Saks. The Cell Probe Complexity of Dynamic Data Structures. In *Proceedings of the Twenty-first Annual ACM Symposium on Theory of Computing*, STOC '89, pages 345–354, New York, NY, USA, 1989. ACM. doi:10.1145/73007.73040.

10 M. L. Fredman. The Inherent Complexity of Dynamic Data Structures which Accommodate Range Queries. In *21st Annual Symposium on Foundations of Computer Science (sfcs 1980)*, pages 191–199, October 1980. doi:10.1109/SFCS.1980.47.

11 Michael L. Fredman. A Lower Bound on the Complexity of Orthogonal Range Queries. *J. ACM*, 28(4):696–705, October 1981.

12 Michael L. Fredman. The Complexity of Maintaining an Array and Computing its Partial Sums. *J. ACM*, 29(1):250–260, January 1982. doi:10.1145/322290.322305.

13 N. Higham. *Functions of Matrices*. Society for Industrial and Applied Mathematics, 2008. URL: https://epubs.siam.org/doi/abs/10.1137/1.9780898717778.

14 G. S. Lueker. A Data Structure for Orthogonal Range Queries. In *19th Annual Symposium on Foundations of Computer Science (sfcs 1978)*, pages 28–34, October 1978. doi:10.1109/SFCS.1978.1.

15 Kurt Mehlhorn. *Data Structures and Algorithms 3: Multi-dimensional Searching and Computational Geometry*. Springer-Verlag New York, Inc., New York, NY, USA, 1984.

16 Franco P. Preparata and Michael I. Shamos. *Computational Geometry: An Introduction*. Springer-Verlag, 1985.

17 A. Toni. Lower Bounds on Zero–One Matrices. *Linear Algebra and its Applications*, 376:275–282, 2004. doi:10.1016/j.laa.2003.06.018.