Theoretical Computer Science 581 (2015) 97–101

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Note

Low space data structures for geometric range mode query ☆

Stephane Durocher [a], Hicham El-Zein [b], J. Ian Munro [b], Sharma V. Thankachan [c]

[a] Department of Computer Science, University of Manitoba, Winnipeg, Canada
[b] Cheriton School of Computer Science, University of Waterloo, Waterloo, Canada
[c] School of Computational Science and Engineering, Georgia Institute of Technology, Atlanta, USA

ARTICLE INFO

Article history:
Received 4 October 2014
Received in revised form 28 January 2015
Accepted 3 March 2015
Available online 6 March 2015
Communicated by S. Sen

Keywords:
Range queries
Mode
Data structures
Color queries

ABSTRACT

Let $\mathcal{S}$ be a set of n points in d dimensions such that each point is assigned a color. Given a query range $\mathcal{Q} = [a_1, b_1] \times [a_2, b_2] \times \ldots \times [a_d, b_d]$, the geometric range mode query problem asks to report the most frequent color (i.e., a mode) of the multiset of colors corresponding to points in $\mathcal{S} \cap \mathcal{Q}$. When $d = 1$, Chan et al. (2012) [1] gave a data structure that requires $O(n + (n/\Delta)^2/w)$ words and supports range mode queries in $O(\Delta)$ time for any $\Delta \geq 1$, where $w = \Omega(\log n)$ is the word size. Chan et al. also proposed a data structures for higher dimensions (i.e., $d \geq 2$) with $O(s_n + (n/\Delta)^{2d})$ words and $O(\Delta \cdot t_n)$ query time, where s_n and t_n denote the space and query time of a data structure that supports orthogonal range counting queries on the set $\mathcal{S}$. In this paper we show that the space can be improved without any increase to the query time, by presenting an $O(s_n + (n/\Delta)^{2d}/w)$-word data structure that supports orthogonal range mode queries on a set of n points in d dimensions in $O(\Delta \cdot t_n)$ time, for any $\Delta \geq 1$. When $d = 1$, these space and query time costs match those achieved by the current best known one-dimensional data structure.

1. Introduction

Range query problems have proven to be of fundamental importance in computational geometry, both as tools employed to provide efficient solutions to various geometric problems, and also in the study of their optimality with respect to space and query time. In this paper we investigate the range mode query problem in a multi-dimensional setting:

Definition 1 *(Range mode query).* Given $\mathcal{S}$, a set of n points in d dimensions, such that each point is assigned a color, a *range mode query* $\mathcal{Q} = [a_1, b_1] \times [a_2, b_2] \times \ldots \times [a_d, b_d]$ asks for the most frequent color in $\mathcal{S} \cap \mathcal{Q}$.

Although the one-dimensional range mode query problem has received significant attention [3–7], only limited attention has been paid to the multi-dimensional problem. The first solution for the multi-dimensional case was proposed recently by Chan et al. [3]. They gave a data structure that requires $O(s_n + (n/\Delta)^{2d})$ words and supports d-dimensional range mode queries in $O(\Delta \cdot t_n)$ time for any $\Delta \geq 1$, where s_n is the space of an orthogonal range counting data structure in d dimensions with query time t_n. The model of computation is the standard Word RAM model with word size $w = \Omega(\log n)$.

☆ A preliminary version of this work appeared in CCCG 2014 [2]. This work was supported in part by the National Science and Engineering Research Council of Canada and the Canada Research Chairs program.

E-mail addresses: durocher@cs.umanitoba.ca (S. Durocher), helzein@uwaterloo.ca (H. El-Zein), imunro@uwaterloo.ca (J.I. Munro), sharma.thankachan@gmail.com (S.V. Thankachan).

http://dx.doi.org/10.1016/j.tcs.2015.03.011
0304-3975/Published by Elsevier B.V.

Also d is assumed to be constant. In this paper we show that the space of the range mode query data structure can be improved to $O(s_n + (n/\Delta)^{2d}/w)$ words while maintaining the same query time. That is, our data structure achieves the same asymptotic space and query time costs as those of the current best known range mode query data structure for one-dimensional data [1].

1.1. Related work

The first range mode data structure (on arrays) was proposed by Krizanc et al. [4], requiring $O(n)$ words for $O(\sqrt{n}\log\log n)$ query time. They also described data structures that provide constant query time using $O(n^2\log\log n/\log n)$ words, and $O(n^\epsilon \log n)$ query time using $O(n^{2-2\epsilon})$ words. Petersen and Grabowski [5] improved the first bound to constant time and $O(n^2\log\log n/\log^2 n)$ words. Peterson [6] later improved the second bound to $O(n^\epsilon)$ time queries using $O(n^{2-2\epsilon})$ words for any $\epsilon \in (0, 1/2]$. Chan et al. [3] further improved the last bound to $O(n^\epsilon)$ time queries using $O(n^{2-2\epsilon}/\log n)$ words. Using reductions from boolean matrix multiplication, they showed show that query times significantly lower than $\sqrt{n}$ are unlikely for this problem with linear space [3]. Finally, Greve et al. [7] proved a lower bound of $\Omega(\log n/\log(s \cdot w/n))$ time for any data structure that supports range mode query on arrays using s memory cells of w bits in the cell probe model.

Given any fixed $\alpha \in (0, 1]$ and any range $\mathcal{Q}$, the objective of an approximate range mode query is to return an element whose frequency in $\mathcal{S} \cap \mathcal{Q}$ is at least $\alpha \cdot m$, where m denotes the frequency of the mode of $\mathcal{S} \cap \mathcal{Q}$. Bose et al. [8] gave a data structure that requires $O(n/(1-\alpha))$ words and answers approximate range mode queries in $O(\log\log_{1/\alpha}(n))$ time, as well as a data structure that answers queries in constant time when $\alpha \in \{1/2, 1/3, 1/4\}$, using $O(n\log n)$, $O(n\log\log n)$, and $O(n)$ words respectively. Greve et al. [7] improved previous results by giving a data structure that supports range mode queries in $O(1)$ time using $O(n)$ words when $\alpha = 1/3$, and $O(\log(\alpha/(1-\alpha)))$ time using $O(n\alpha/(1-\alpha))$ words when $\alpha \in [1/2, 1)$.

Another related question is the problem of finding a least frequent element (with frequency at least one) in a one dimensional range. Chan et al. [9] gave the first solution with linear space and $O(\sqrt{n})$ query time. Later, Durocher et al. [10] improved the query time to $O(\sqrt{n/w})$. Our improved data structure for range mode query is based on the encoding ideas from [10]. See the recent survey by Skala [11] for further reading.

2. Framework

A point $p \in \mathcal{S}$ is represented by a $(d+1)$-tuple $(p_1, p_2, \ldots, p_d, p_c)$, where for each i, p_i denotes p's coordinate in dimension i, and p_c is the color associated with p. When d is constant, we can map the input set $\mathcal{S}$ to rank space using standard techniques,[1] requiring $O(n)$ words of additional space and an $O(\log n)$ additive increase to query time to map any point in rank space back to its original value. Throughout the paper we assume that points are in rank space. That is for any point $p \in \mathcal{S}$ and any $i \in \{1, \ldots, d\}$, $p_i \in \{0, \ldots, n-1\}$. Moreover if $p \neq q$, then $p_i \neq q_i$. This ensures the following:

Lemma 1. *The number of points of $\mathcal{S}$ in a rectangle $\mathcal{Q} = [\alpha_1, \beta_1] \times [\alpha_2, \beta_2] \times \ldots \times [\alpha_d, \beta_d]$ is at most the minimum element in $\{\beta_i - \alpha_i + 1 \mid 1 \le i \le d\}$.*

Definition 2. Let $\Delta \ge 1$ be an integer. A Δ-box is a region $R = [\alpha_1, \beta_1] \times [\alpha_2, \beta_2] \times \ldots \times [\alpha_d, \beta_d]$, where for all i, $\alpha_i = k_i\Delta$ and $\beta_i = k'_i\Delta$ for any integers k_i and k'_i.

There are $\Theta((n/\Delta)^{2d})$ distinct Δ-boxes in our rank space grid, which includes empty boxes, i.e., boxes with $\alpha_i = \beta_i$ for some $i \in [1, d]$. Each Δ-box $R = [\alpha_1, \beta_1] \times [\alpha_2, \beta_2] \times \ldots \times [\alpha_d, \beta_d]$ can be identified using a unique index, given by:

$$rank(R, \Delta) = \sum_{i=1}^{d} (\alpha_i/\Delta) \cdot \phi^{2i-2} + (\beta_i/\Delta) \cdot \phi^{2i-1},$$

where $\phi = \lfloor n/\Delta \rfloor + 1$. Notice that $rank(R, \Delta)$ can be computed in $O(d)$ time (i.e., constant time when d is a constant) given any R and Δ.

3. Data structure of Chan et al.

In this section we describe the data structure presented by Chan et al. [3]. The data structure relies on the following observation [4]: a mode of $\mathcal{Q}_1 \cup Q_2$ is either a mode of $\mathcal{Q}_1$ or an element in $\mathcal{Q}_2$. Throughout Sections 3 and 4 we assume that d is a constant.

---

[1] For $k = 1, 2, \ldots, d$, let $E_k[0, n-1]$ be an array of length n sorted in ascending order such that the entries in E_k represent the kth coordinates of the points in $\mathcal{S}$. A point $p \in \mathcal{S}$ maps to the point $p'(z_1, z_2, \ldots, z_d, p_c)$ in rank space, where $E_k[z_k]$ is equal to the kth coordinate of p. The total space for maintaining these arrays is $d \cdot n$ words.

A query $\mathcal{Q} = [a_1, b_1] \times [a_2, b_2] \times \ldots \times [a_d, b_d]$ on $\mathcal{S}$ maps to an equivalent query $\mathcal{Q}^* = [a_1^*, b_1^*] \times [a_2^*, b_2^*] \times \ldots \times [a_d^*, b_d^*]$ in rank space, where $E_k[a_k^* - 1] < a_k \le E_k[a_k^*]$ and $E_k[b_k^*] \le b_k < E_k[b_k^* + 1]$. We can obtain $\mathcal{Q}^*$ from $\mathcal{Q}$ in $O(d\log n)$ time by applying $2d$ binary search operations.