

# Εργασία Πολυδιάστατων Δομών Δεδομένων έτους 2021-22

Τριανταφυλλόπουλος Παναγιώτης<sup>1</sup>

<sup>1</sup>Τμήμα Μηχανικών Η/Υ και Πληροφορικής, Πανεπιστήμιο Πατρών

26 Σεπτεμβρίου 2022

## Περίληψη

Στόχος της εργασίας είναι η υλοποίηση και η μελέτη Δένδρων Διαστημάτων (ΔΔ) και Δένδρων Ευθυγράμμων Τμημάτων (ΔΕΤ), στα πλαίσια του μαθήματος Πολυδιάστατες Δομές Δεδομένων, για interval και stabbing queries αντίστοιχα.

## 1 Περιγραφή Δομών

### 1.1 Δένδρο Διαστημάτων

#### 1.1.1 Σύντομη Περιγραφή Δομής και Υλοποίησης

Το ΔΔ είναι δομή στην οποία αποθηκεύονται διαστήματα σε  $O(n)$  χώρο ενώ μπορεί να απαντά σε ερωτήματα τομής διαστημάτων σε  $O(\log(n))$  χρόνο [1]. Για να διατηρηθεί ο λογαριθμικός χρόνος απάντησης θέλουμε το δέντρο να είναι υψοζυγισμένο. Για τον λόγο αυτό ως 'σκελετό' του δέντρου χρησιμοποιούμε ένα AVL δένδρο [2]. Ταυτόχρονα, για διευκόλυνση στην αναζήτηση, σε κάθε κόμβο μαζί με το διάστημα αποθηκεύουμε δείκτες προς τα παιδιά αν υπάρχουν και το μέγιστο άκρο διαστήματος ( $max[x]$ ) που υπάρχει στο υποδέντρο με ρίζα το  $x$  (επαυξημένο δέντρο) [3]. Στο [3] περιγράφεται επίσης ο αλγόριθμος αναζήτησης διαστήματος ενώ ο αλγόριθμος αναζήτησης διαστήματος που επιστρέφει όλα τα διαστήματα του δέντρου που επικαλύπτονται με ένα διάστημα  $i$  περιγράφεται σε διαφάνειες του Rutgers University που περιέχουν λύσεις των ασκήσεων του [3]. Για τα διαστήματα χρησιμοποιούμε μια κλάση Interval της οποίας τα αντικείμενα περιέχουν δύο στοιχεία, *high* και *low* με  $0 < low < high$ .

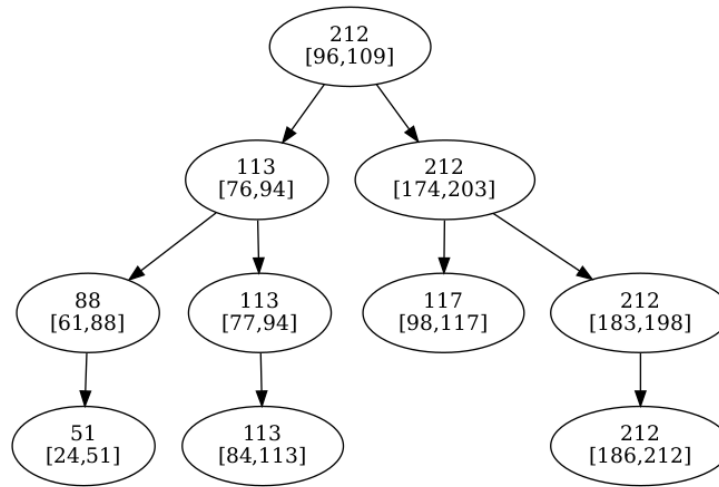
#### 1.1.2 Βασικές Πολυπλοκότητες Δένδρου

Αποδεικνύεται [1–3] ότι το ΔΔ, αν  $n =$  πλήθος διαστημάτων :

- Απαιτεί  $O(n)$  χώρο

- Μπορεί να κατασκευαστεί σε  $O(n * \log(n))$  χρόνο
- Απαντά σε  $O(\log(n))$  όταν επιστρέφει ένα διάστημα, ενώ αν επιστρέφει  $k$  διαστήματα απαντά σε  $O(\min(n, k * \log(n)))$  χρόνο.
- Οι περιστροφές κατά την εισαγωγή γίνονται σε  $O(1)$

Ένα παράδειγμα  $\Delta\Delta$ :



## 1.2 Δένδρο Ευθυγράμμων Τμημάτων

### 1.2.1 Σύντομη Περιγραφή Δομής και Υλοποίησης

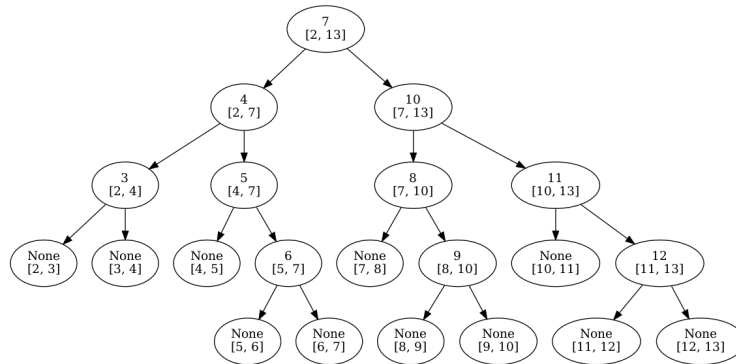
Το  $\Delta\Delta$  είναι μια στατική δομή στην οποία αποθηκεύονται διαστήματα σε  $O(n * \log(n))$  χώρο ενώ μπορεί να απαντά σε ερωτήματα τομής διαστημάτων με ένα συγκεκριμένο σημείο (stab query) σε  $O(\log(n) + k)$  χρόνο, όπου  $k$  το πλήθος των διαστημάτων που επιστρέφει [1, 4]. Στη γενική του μορφή είναι δένδρο εύρεσης με  $2 * n + 1$  φύλλα τα οποία αντιστοιχούν στα διαστήματα  $(-\infty, x_1), [x_1, x_1], (x_1, x_2), \dots, [x_n, x_n], (x_n, +\infty)$  (elementary intervals), σε αύξουσα σειρά από τα αριστερά προς τα δεξιά, με  $x_i$  να είναι το δεξί άκρο του  $i$ -οστού διαστήματος το οποίο ανήκει σε ένα πεπερασμένο σύνολο διαστημάτων  $S$  (locus approach [1]). Κάθε διάστημα  $Int(v)$  που αποθηκεύεται σε έναν κόμβο  $v$  του δένδρου προκύπτει από την ένωση των διαστημάτων των παιδιών του. Στους κόμβους κρατάμε επίσης μία λίστα ( $list(v)$ ) με τα διαστήματα του  $S$  τα οποία περιέχουν το  $Int(v)$ . Ισχύει ότι:  $list(parent) \cap list(child) = \emptyset$ . Κάθε διάστημα μπορεί να υπάρχει σε πολλές λίστες, ενώ όσο πιο μεγάλο είναι σε μέγεθος τόσο πιο ψηλά αποθηκεύεται στο δέντρο [1]. Επειδή ήταν πιο κατανοητή, επιλέχθηκε η υλοποίηση που περιγράφεται στο [4], section 2 Preliminaries και στις διαφάνειες του Department of Computer Graphics and Interaction, Czech Technical University in Prague, Lecture: 'Windowing'.

### 1.2.2 Βασικές Πολυπλοκότητες Δένδρου

Αποδεικνύεται [1, 4] ότι το ΔΕΤ, αν  $n =$  πλήθος διαστημάτων :

- Απαιτεί  $O(n * \log(n))$  χώρο
- Μπορεί να κατασκευαστεί σε  $O(n * \log(n))$  χρόνο
- Απαντά σε  $O(n * \log(n) + k)$  χρόνο αν επιστρέφει  $k$  διαστήματα.

Ένα παράδειγμα ΔΕΤ (δεν τυπώνονται οι λίστες  $\text{list}(v)$ ):



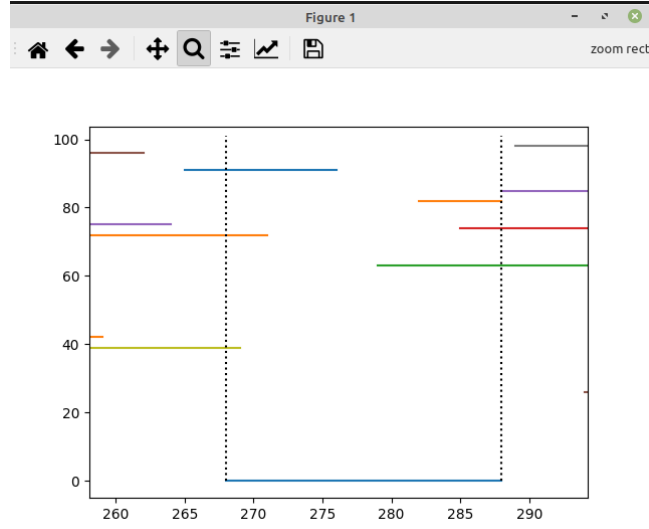
## 2 Περιγραφή Προβλήματος

### 2.1 Interval Query for Interval Tree

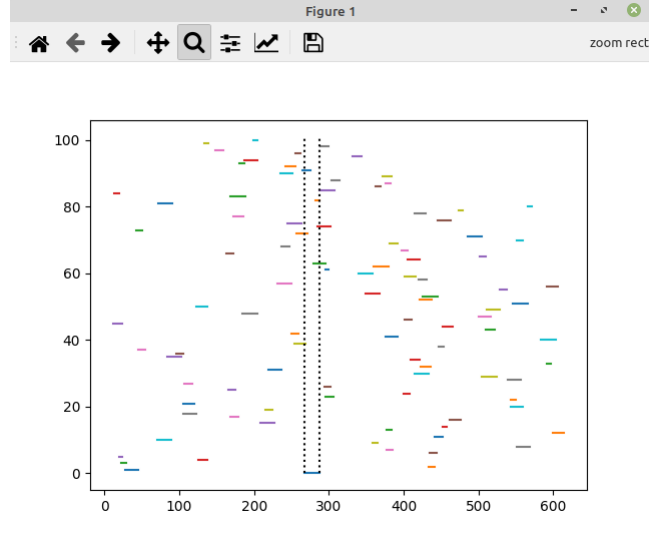
```
Query Interval: [268,288]
First overlap with: [279, 295] max = 368
All overlaps:
Found: [279, 295] max = 368
Found: [254, 269] max = 276
Found: [257, 271] max = 276
Found: [265, 276] max = 276
Found: [285, 302] max = 367
Found: [282, 288] max = 288
Found: [288, 307] max = 367
Ran for 100 intervals
```

Δοθέντος ενός τυχαίου συνόλου διαστημάτων  $S$  και ενός τυχαίου διαστήματος  $q = (x_1, x_2)$ , βρες όλα τα διαστήματα  $i \in S$  με  $i \cap q \neq \emptyset$ . Αρχικά δημιουργούμε  $n$  τυχαία διαστήματα και κτίζουμε το ΔΔ. Στη συνέχεια, παίρνουμε το διάστημα  $q$  και αναζητούμε τα διαστήματα  $i$  με τα οποία επικαλύπτεται. Τα διαστήματα  $i, q$  θεωρούνται κλειστά διαστήματα.

Σχήμα 1: Περιοχή γύρω από το  $q$



Σχήμα 2: Χώρος Διαστημάτων



## 2.2 Stab Query for Segment Tree

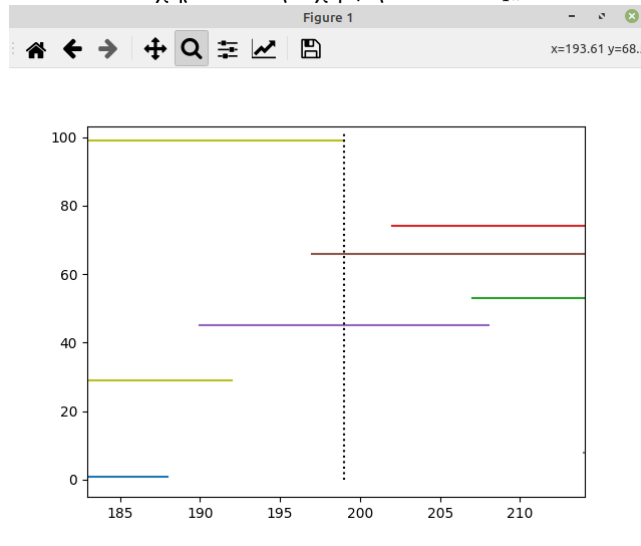
```

Query Point: 199
Found in: [[190, 208]]
Found in: [[197, 216]]
Found in: [[182, 199]]

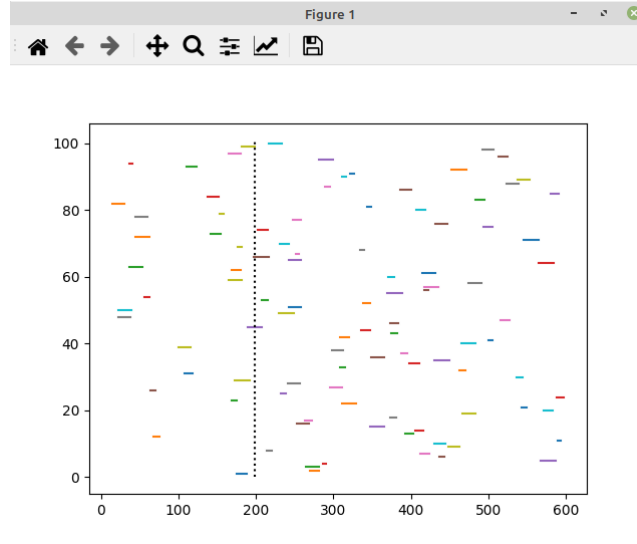
```

Δοθέντος ενός τυχαίου συνόλου διαστημάτων  $S$  και ενός τυχαίου σημείου  $q_x$ , βρες όλα τα διαστήματα  $i = (low_i, high_i) \in S$  με  $q_x \in i$ . Αρχικά δημιουργούμε  $n$  τυχαία διαστήματα και χτίζουμε το ΔΕΤ όπως περιγράψαμε πιο πάνω. Στη συνέχεια, παίρνουμε το σημείο  $q_x$  και αναζητούμε τα διαστήματα  $i$  στα οποία ανήκει. Θεωρούμε πως τα  $i$  είναι κλειστά μόνο στο δεξί τους άκρο.

Σχήμα 3: Περιοχή γύρω από το  $q_x$



Σχήμα 4: Χώρος Διαστημάτων



### 3 Βιβλιογραφία

- [1] A. Tsakalidis, *Advanced Data Structures and Computer-Graphics*. University of Patras, 1995.
- [2] A. Tsakalidis, *Data Structures*. University of Patras, 1995.
- [3] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*. The MIT Press, 2nd ed., 2001.
- [4] L. Wang and X. Wang, “A simple and space efficient segment tree implementation,” *MethodsX*, vol. 6, pp. 500–512, 2019.