Runs of Side-Sharing Tandems in Rectangular Arrays

Shoshana Marcus, Dina Sokol, Sarah Zelikovitz

Kingsborough Community College · CUNY · Brooklyn College · College of Staten Island, The City University of New York

2D Side-Sharing Tandems

ab cd	ab cd

Horizontal

ab cd
ab cd

Vertical

Background

Existing algorithms for locating side-sharing tandems are far from optimal on a 2d array that is sparsely populated with side-sharing tandems.

This work: locate the side-sharing tandems in close to linear time, with respect to both the size of the input array and the number of runs of side-sharing tandems that occur.

Contributions

✧Introduce idea of **run of side-sharing tandems**: maximally extended chain of 2d side-sharing tandems.

✧Demonstrate tight upper bounds on the number of runs of side-sharing tandems that can occur in a rectangular array.

✧Develop an efficient algorithm for locating them.

Algorithm

$O((n^2+\tau)\log n/\log\log n)$ time to locate τ runs of side-sharing tandems in n x n array.

Iteratively identify h-runs of each height $1 \le k \le n$.

1. Identify all h-runs of height 1, by locating 1d runs on each row, in linear time.
2. Find all h-runs of height 2, by linking runs on adjacent rows.
3. Go through each height 3 ≤ k ≤ n (in increasing order), and for each start row 1 ≤ i ≤ n-k+1, identify h-runs of height k by linking h-runs of smaller heights on adjacent rows.

Interval *x*-Intersection Query

Preprocess a set of ψ intervals V

Query: Given an integer x > 0 and interval u = [p, q] with integer endpoints such that 1 ≤ p < n, 1 < q ≤ n, p < q, list all intervals in *V* that intersect u by at least *x* units.

For ω results:

O(ω log ψ / log log ψ) query time

O(ψ log ψ / log log ψ) preprocessing time