December 5, 2017

Matrix multiplication in softly quadratic time

4. Assorted real (floating-point) semi-ring flavors for matrix multiplication

The key idea is that we use tower approach to find largest or smallest pair sum.

What is our tower approach? We use a creative point arrangement that uses closest pairs at least as far as critical distance. We have two towers stacked in plane (i.e. one for rows at bottom and one for columns at top) and specify a min. distance to force certain row-column pair to be considered. We have a slice for each row and each column. Orthogonal to the stacking direction is the direction along which elements for the row or column are dispersed. Distances between intra-row or intra-column elements are s.t. we ensure cross-pair interaction does not occur. The distances are also s.t. we jut down from rows and we jut up from columns in a way reminiscent of a Fresnel lens shape. For more details, see Sharathkumar, section 4.2. Time required for construction is $O(n^2 \cdot \log^2(n))$ and time for query is $O(\log^2(n))$. We introduced a factor of $O(\log(n))$ to these two times because we assume that $O(m)$ bits are enough to describe largest exponent that can be associated with our m-bit primitive, so that n^2 times this largest exponent requires $O(m \cdot \log(n))$ bits. We need this large value for adequate isolation of values in tower; we need around n^2 times the largest exponent describable by a primitive; and the extra factor for construction and query times follows. Also, we avoid introducing another $O(\log(n))$-factor though we wish to specify particular row-column pair by having certain min. distance; we have fixed regular spacing for column rungs and different fixed regular spacing for row rungs.

We can use two tricks - logarithms turn multiplication into addition and exponentials turn addition into multiplication.

We support min-plus, max-plus, min-max, max-min, min-leq, max-leq, min-times, max-times semi-rings.

Many of these flavors are reducible to each other.

We can solve min-plus, max-plus, min-max, max-min using double tower as mentioned before (see Sharathkumar), but with given scalars, not powers.

We can solve min-leq, max-leq using double tower (as mentioned before, but with given scalars, not powers) and infinity-related logic (i.e. for min-leq, get min. distance, which may possibly be infinity; and for max-leq, get max. distance, which may possibly be infinity).

We can solve min-times and max-times by reducing to min-plus and max-plus by splitting into $+/+, +/-, -/+, -/-$ subproblems (because the product for each may or may not be modulated by a negative sign) and via pre-processing using magnitude logarithms and post-processing by selecting the most suitable pair of elements (out of four) responsible for a certain sum.

This implies ω is softly two for real MM for these semi-rings.

Running time

$O(n^2 \cdot \log^2(n))$

References

- Sharathkumar et al. - Range-aggregate proximity queries (2007)