This week we will see two problems related to both APSP and 3SUM. One of them, $(\min,+)$-convolution is no harder than APSP and 3SUM, the other, Zero Triangle is no easier than APSP and 3SUM.

1 Reductions to Zero Triangle

Definition 1 (Zero Triangle)**.** Given an n-node graph $G = (V, E)$ with integer edge weights $w : E \to \mathbb{Z}$, determine if there exist three vertices $a, b, c \in V$ such that $w(a, b) + w(b, c) + w(c, a) = 0$.

For our first reduction we will need the following observation.

Lemma 1. *For three non-negative integers x, y, z, we have $x + y > z$ if and only if there exists k such that either*

$$\lfloor x/2^k \rfloor + \lfloor y/2^k \rfloor = \lfloor z/2^k \rfloor + 1,$$

or the $(k-1)$-st least significant bit[1] *of x and y is 1, of z is 0, and*

$$\lfloor x/2^k \rfloor + \lfloor y/2^k \rfloor = \lfloor z/2^k \rfloor.$$

Exercise 1. Prove the above lemma.

Theorem 1. *If Zero Triangle is in $T(n)$ time, then Negative Triangle is in $\mathcal{O}(T(n)\log n)$ time. In particular, if Zero Triangle is in truly subcubic time, then APSP is in truly subcubic time.*

Proof. We will reduce the tripartite version of the Negative Triangle problem to Zero Triangle. In the Negative Triangle problem we are looking for a triangle with edge weights x, y, z such that $x + y + z < 0$, which is equivalent to $(-x + n^c) + (-y + n^c) > (z + 2 \cdot n^c)$. We apply Lemma 1, iterate over $\mathcal{O}(\log n^c)$ possible choices of k and two possible cases from the lemma, and in each iteration we check the relevant condition by solving the Zero Triangle problem. □

Theorem 2. *If Zero Triangle is in $T(n)$ time, then Convolution 3SUM is in $\mathcal{O}(\sqrt{n} \cdot T(\sqrt{n}))$ time, and hence 3SUM is in $\widetilde{\mathcal{O}}(\sqrt{n} \cdot T(\sqrt{n}))$. In particular, if Zero Triangle is in $\mathcal{O}(n^{3-\varepsilon})$ time, then 3SUM is in $\widetilde{\mathcal{O}}(n^{2-\varepsilon/2})$ time.*

Proof. We create a tripartite graph with vertex set X, Y, Z, where $|X| = |Y| = \sqrt{n}$ and $|Z| = \mathcal{O}(n)$. We will index vertices of X and Y by integers from $\{0, \ldots, \sqrt{n} - 1\}$, and vertices of Z with integers from $\{-2n, \ldots, 2n\}$.

For $x \in X$ and $z \in Z$ we put weight $w(x, z) = a_{x+z}$.

For $y \in Y$ and $z \in Z$ we put weight $w(y, z) = a_{y\sqrt{n}-z}$.

For $x \in X$ and $y \in Y$ we put weight $w(x, y) = -a_{x+y\sqrt{n}}$.

Now we have that for any $i, j \in [n]$ (such that $i + j \in [n]$) we can always find x, y such that $i + j = x + y\sqrt{n}$, then set z so that $i = x + z$, and then we will have $j = y\sqrt{n} - z$. Then, $a_i + a_j = a_{i+j}$ if and only if $w(x, y) + w(y, z) + w(x, z) = 0$, that is x, y, z is a zero

[1] The k-th least significant bit of a is $\lfloor a/2^k \rfloor \bmod 2$.

triangle. Hence, all we need to do in order to solve Convolution 3SUM is to solve the Zero Triangle problem on our graph. The graph has $\mathcal{O}(n)$ nodes, which would be too many to get the desired running. Fortunately, it has many nodes only in one part, so we can do the following. We split Z arbitrarily into $Z_1, Z_2, \ldots, Z_{\sqrt{n}}$, each Z_i of size $\mathcal{O}(\sqrt{n})$. Now, for each $i \in [\sqrt{n}]$ we solve Zero Triangle on the graph induced by $X \cup Y \cup Z_i$, which takes time $\mathcal{O}(T(\sqrt{n}))$ per iteration, or $\mathcal{O}(\sqrt{n} \cdot T(\sqrt{n}))$ time in total. We return "yes" if at least one instance of Zero Triangle returned "yes". □

2 Reductions from $(\min, +)$-convolution

Definition 2 ((min,+)-convolution)**.** Given two n-element vectors $(a_1, \ldots, a_n), (b_1, \ldots, b_n)$, $a_i, b_i \in \{-n^{\text{const}}, \ldots, n^{\text{const}}\}$, compute the n-element vector c such that

$$c_k = \min_{i+j=k} a_i + b_j = \min\big\{(a_0 + b_k), (a_1 + b_{k-1}), \ldots, (a_k + b_0)\big\}.$$

Theorem 3. *If APSP is in $T(n)$ time, then $(\min, +)$-convolution is in $\mathcal{O}(\sqrt{n} \cdot T(\sqrt{n}))$ time. In particular, if APSP is in $\mathcal{O}(n^{3-\varepsilon})$ time, then $(\min, +)$-convolution is in $\mathcal{O}(n^{2-\varepsilon/2})$ time.*

Proof. We will use essentially the same proof strategy as in the reduction from Zero Triangle to Convolution 3SUM, but let us present it in a different way. Note that each entry c_k of the output of $(\min, +)$-convolution corresponds to the $(\min, +)$-inner product of vector a with vector b reversed and shifted by k. We will create two matrices A and B, each with $\mathcal{O}(n^{3/2})$ entries. Matrix A will contain all "baby-step" shifts of a, from 0 to $\sqrt{n}$, matrix B will contain all "giant-step" shifts of (reversed) b, from $0 \cdot \sqrt{n}$ to $\sqrt{n} \cdot \sqrt{n}$. Each entry of the $(\min, +)$-product $A \star B$ will correspond to an entry of the $(\min, +)$-convolution output.

Let's make it formal. Matrix A is the $\sqrt{n} \times \mathcal{O}(n)$ matrix, and i-th row of A equals to a shifted by i, that is

$$A[i][k] = a_{k-i}.$$

We put ∞ where the index overflows. Similarly, matrix B is the $\mathcal{O}(n) \times \sqrt{n}$ matrix, and j-th column of B equals to b reversed and shifted by $j \cdot \sqrt{n}$, that is

$$B[k][j] = b_{n-k-j\cdot\sqrt{n}}.$$

Now, note that

$$(A \star B)[i][j] = \min_k A[i][k] + B[k][j] = \min_k a_{k-i} + b_{n-k-j\cdot\sqrt{n}} = \\ = \min_{k'} a_{k'} + b_{n-(k'+i)-j\cdot\sqrt{n}} = c_{n-(i+j\sqrt{n})}.$$

In order to compute $A \star B$, we first split each of them into $\mathcal{O}(\sqrt{(n)})$ square $\sqrt{n} \times \sqrt{n}$ matrices; then, we compute $\mathcal{O}(\sqrt{n})$ products, each in time $T(\sqrt{n})$; finally, we take the element-wise minimum. □

Exercise 2. Note that in the above reduction we have not used any specific properties of operations min and $+$. In particular, the reduction should work for any matrix product and the corresponding convolution problem. Does it work for the Hamming distance variants (convolution in the April 15th's notes, matrix product in the Problem Set 8)? Why, or why not?

Theorem 4. *If 3SUM is in truly subquadratic time, then* $(\min, +)$*-convolution is in truly subquadratic time.*

Proof. We will go through an intermediate problem called Subadditivity. In this problem we are given an integer vector $a_1, \ldots, a_n$, and we need to decide if there is a pair $i, j \in [n]$ such that $a_i + a_j > a_{i+j}$. (If there is no such pair, the vector is called *subadditive*.) Note that Subadditivity is to $(\min, +)$-convolution what Negative Triangle is to $(\min, +)$-product. In particular, we can prove that Subadditivity and $(\min, +)$-convolution are subquadratic equivalent. Now, we just proceed as in Theorem 1, that is, we use Lemma 1 to reduce Subadditivity to $\mathcal{O}(\log n)$ instances of Convolution 3SUM. For more detail, see Appendix A in `https://arxiv.org/pdf/1702.07669.pdf`. □

3 Further reductions

If time permits, we will prove that $(\min, +)$-convolution is subquadratic equivalent to the Knapsack problem with $t = \Theta(n)$, and that Zero Triangle can be solved in truly subcubic time if it is possible to determine for each edge in an m-edge graph whether the edge belongs to a triangle in $\mathcal{O}(m^{4/3-\varepsilon})$ time . The latter problem can be solved in $\mathcal{O}(m^{2\omega/(\omega+1)})$ time, which is $\mathcal{O}(m^{1.41})$ with the current bounds for ω, and becomes $\mathcal{O}(m^{4/3})$ if $\omega = 2$.