CMSC 858F: Algorithmic Lower Bounds: Fun with Hardness Proofs
Fall 2014
Quadratic Hardness and the 3-SUM Problem

Instructor: Mohammad T. Hajiaghayi
Scribe: Karthik Abinav S

September 23, 2014

1 Overview

In the previous lecture, we looked at the APSP problem and some of the other closely related problems. We studied the cubic hardness of these problems. In this lecture, we will go about doing something similar, but in the domain of quadratic hardness. With regard to this, we will choose 3-SUM problem as the representative problem. We will look at some related problems, that can be reduced to the 3-SUM problem in sub-quadratic time. And finally, make some comments about the quadratic hardness of these problems.

2 Problem Definition

Definition 1 (3-SUM problem) *Given a multi-set S of integers, such that* $S \subseteq \{-n^3, \ldots, n^3\}$*, the problem is to determine, if there exists three integers* $a, b, c \in S$ *such that* $a + b + c = 0$*.*

The simplest algorithm is to first sort the multi-set S. Then fix values of a,b and do a binary search on the value of c. This will require a running time of

$\tilde{O}(n^2)$. Formally, the algorithm looks as follows:

```
Data: A multi-set S ⊆ {−n^3, ..., n^3}
Result: A boolean value True if there exists three numbers a,b,c such
        that a+b+c =0 and false otherwise
Sort the integers in multi-set S ;
for a ∈ S do
    for b ∈ S do
        found = binarySearch(-(a+b),S) ;
        if found then
            Output Yes ;
            Halt ;
        end
    end
end
Output No ;
```

Algorithm 1: An $\tilde{O}(n^2)$ algorithm for the 3-SUM problem

In fact, we can get an algorithm that runs in $O(n^2)$. In other words, the extra poly logarithmic factor can be removed. The algorithm is as follows:

```
Data: A multi-set S ⊆ {−n^3, ..., n^3}
Result: A boolean value True if there exists three numbers a,b,c such
        that a+b+c =0 and false otherwise
Sort the integers in multi-set S ;
for i =1 to n do
    a = S[i] ;
    bPointer = i+1 ;
    cPointer = n;
    while bPointer < cPointer do
        b = S[bPointer] ;
        c = S[cPointer] ;
        if a + b + c = 0 then
            Output Yes ;
            Halt ;
        end
        if a +b + c > 0 then
            cPointer = cPointer - 1 ;
        end
        if a+b+c < 0 then
            bPointer = bPointer + 1 ;
        end
    end
end
Output No ;
```

Algorithm 2: An $O(n^2)$ algorithm for the 3-SUM problem

Scribe: Karthik Abinav S
Lecture 5 **Date:** 09/23/2014

3 3-SUM Conjecture

Definition 2 *It is conjectured that the problem 3-SUM is truly quadratic, i.e. given a constant* $\epsilon > 0$*, there is no algorithm that can decide 3-SUM in time* $O(n^{2-\epsilon})$.

A lot of problems are proved to be 3-SUM hard. In other words, assuming the 3-SUM conjecture is true, those problems are proved to be truly quadratic. The paper by Abboud and Williams [1], gives many hardness results based on this conjecture, for dynamic problems. In these kinds of problems, we are given a class of input objects. The task is to design efficient algorithms and data structures to answer multiple queries on this set of input objects. Also, the input set is modified between two queries, i.e. an element is removed or inserted or value is modified.

As a remark, one should note that unlike cubic hardness and quadratic hardness, it is not so relevant to study linear time hardness. This is because an exact algorithm needs at least linear time to read the input bits.

4 Sub-Quadratic Reductions

Definition 3 *We say a problem A reduces to problem B under sub-quadratic reductions if, given a sub-quadratic algorithm for B, we can use it to obtain a sub-quadratic algorithm to problem A.*

We will now look at some of the problems which have a sub-quadratic reduction to 3-SUM or to one of the other problems which further have a sub-quadratic reduction to 3-SUM.

4.1 3-SUM'

This a closely related problem to the 3-SUM problem.

Definition 4 (3-SUM') *Given three set of integers A,B,C such that the* $|A| = |B| = |C| = n$*, the problem is to decide if there exists three integers* $a \in A$*,* $b \in B$*,* $c \in C$*, such that* $a + b = c$*.*

4.1.1 Reduction from 3-SUM to 3-SUM'

This direction of reduction is trivial. We just set A=S, B=S, C=S.

4.1.2 Reduction from the 3-SUM' to 3-SUM

Without loss of generality, assume that all elements in set A,B,C are positive (We could add a large number M to every element in A and B, and 2M to every element in C).

Let m= $2 * \max(A, B, C)$, i.e. maximum element in set $A \cup B \cup C$. We construct the set S as follows:

- For each element $a \in A$, add $a' = a + m$ to the set S.
- For each element $b \in B$, add $b' = b$ to the set S.
- For each element $c \in C$, add $c' = -c - m$ to the set S.

Claim 1 *a+b =c if and only if a'+b'+c' = 0.*

Proof: The only if part of this claim is trivially true, i.e. if a+b=c, then a'+b'+c' = 0. Let us now look at the if part of the claim.

From the above construction, observe that $m < a' \leq 1.5m$, $0 < b' \leq 0.5m$ and $-1.5m \leq c' < -m$. Let $x, y, z \in S$, x+y+z = 0.

We can now make the following observations :

- At most one of the three elements can come from the set A. Suppose on the contrary, if two elements were from set A, then the minimum possible sum of the three numbers would be strictly greater than $m + m - 1.5m = 0.5m > 0$. Hence, the sum of 0 can never be achieved.
- At most one element can come from the set C. Suppose, on the contrary there were two elements from the set C, then the maximum achievable sum is strictly less than $-m - m + 1.5m = -0.5m < 0$. Hence, again the sum of 0 can never be achieved.
- At least one element should come from the set C. This is because the elements in set A and B are all positive. Hence, the only way to get a sum of 0, is to have at least one negative number and that should come from the set C.
- It cannot be the case that two elements from set B and one element from set C happens. The sum of positive number coming from B can be at most m. The minimum possible negative number is strictly lesser than $-m$. Hence, the sum is strictly greater than 0.

From the above observations, it is clear that the only possible case is when x was obtained from an element in A, y was obtained from an element in B and z was obtained from an element in C. This completes the proof of the claim.

It is important to note that, the construction of the new set S from the original sets takes only linear time. ■

Infact, this can be generalized to prove that k-SUM and k'-SUM are equivalent under sub-quadratic reductions.

Note that a simple $O(n^2)$ algorithm exists for the 3-SUM' problem. First sort the sets B and C. Now, for each $a \in A$, check if the sets $a + B$ and C has a non-empty intersection, where $a + B = \{a + b : b \in B\}$.

4.2 GeomBase

Definition 5 (GeomBase Problem) *Given n points each having integer coordinates and lying on one of three horizontal lines (lines parallel to x-axis) with y coordinates 0,1,2 , the problem is to decide if there exists a non-horizontal line that contains any three points.*

We will now show a simple reduction from the 3-SUM' problem to the GeomBase problem.

Theorem 1 *There exists a sub-quadratic reduction from 3-SUM' to GeomBase.*

Proof:
For every element $a \in A$, create a point (a,0). Similarly, for every element $b \in B$, create a point (b,2). And for every element $c \in C$ create a point ($\frac{c}{2}$,1).

It can be immediately seen that, points (a,0), (b,2) and ($\frac{c}{2}$,1) are collinear if and only if a+b = c (This is because $\frac{c}{2}$ - a = b - $\frac{c}{2}$ for the points to be collinear).
■

In fact, the problems 3-SUM' and GeomBase are equivalent under sub-quadratic reductions. The following theorem shows the reduction in other direction.

Theorem 2 *There is a sub-quadratic reduction from GeomBase to 3-SUM'.*

Proof: The reduction is somewhat complementary to the reduction in other direction. For each point (a,0) in the GeomBase problem, add an element a to the set A. For each point (b,2), add a point b to the set B. And for every point (c,1), add the element $2c$ to the set C.

It is easy to observe the correctness of this reduction.

■

4.3 Three-Points-on-Line

We will now see another problem in geometry known as the Three-Points-on-Line problem and show a sub-quadratic reduction from the 3-SUM problem to this problem.

Definition 6 (Three-Points-on-Line problem) *Given a set R of points on the plane, the problem is to decide if there exists a line passing through at least three points from the set R.*

Theorem 3 *There is a sub-quadratic reduction from the 3-SUM problem to the Three-Points-on-Line problem.*

Proof: We are given a set S of n integers. For every element $x \in S$, we will add a point (x, x^3) to the set R of points. The following claim will prove the correctness of this reduction.

Claim 2 *For every $a, b, c \in S$, a+b+c = 0 if and only if the points (a, a^3), (b, b^3) and (c, c^3) are colinear.1*

Proof: This was part of a question on Homework 1. Hence, we will avoid the proof in this scribe. ■

The claim essentially completes the reduction. Note, that construction of the set R takes only linear time. ■

4.4 Point-on-Three-Lines

This problem is the geometric dual[1] of the Three-Points-on-Line problem.

Definition 7 (Point-on-Three-Lines problem) *Given a set of lines in the plane, the problem is to decide if there exists three lines in that set, which intersect at a common point.*

Since, this is the geometric dual of the Three-Points-on-Line problem, there exist a sub-quadratic reduction from 3-SUM to this problem. In other words, this problem is also 3-SUM hard.

4.5 Visibility Between Segments

Definition 8 (Visibility Between Segments Problem) *Given a set S of 'n' horizontal line segments on a plane, and two fixed horizontal line segement S_1 and S_2, the problem is to decide, whether there exists a point s_1 on S_1 and a point s_2 on S_2, such that a line segment can been drawn between s_1 and s_2 without intersecting any of the given n horizontal lines.*

We can show that this problem is 3-SUM hard by showing a reduction from GeomBase problem to this problem.

Theorem 4 *There exists a sub-quadratic reduction from GeomBase problem to Visibility Between Segments problem.*

[1] For more details about Geometric Duality, please refer to the lecture notes here - `https://www.cs.duke.edu/~harish/papers/geoduality.pdf`

Proof: Let the points on the three lines A,B,C be sorted by their x-coordinates. Let the ordered set of points be $a_1, a_2, \ldots, a_i$, $b_1, b_2, \ldots, b_j$ and $c_1, c_2, \ldots, c_k$ on the lines A,B,C respectively. Let ϵ be equal to $\frac{1}{5}$. Transform the points on line A to horizontal line segments, with x-coordinates in the interval $[a_1 + \epsilon,$ $a_2 - \epsilon]$, ..., $[a_{i-1} + \epsilon,\ a_i - \epsilon]$ and having the same y-coordinate. Similarly, transform the points on line B and line C to horizontal line segments. Note, that the y-coordinate for the points from line A, the points from line B and the points from line C are distinct. Additionally, construct two distinct horizontal line segments S_1 and S_2, such that all these line segments are enclosed within S_1 and S_2. The diagram below gives a pictorial representation of this construction.

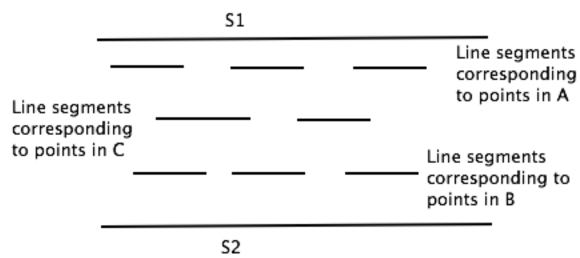

Figure 1: Pictorial representation of the reduction

■

Claim 3 *There exists a line through points* $a \in A$, $b \in B$ *and* $c \in C$ *if and only if there is a line of sight between line segments* S_1 *and* S_2.

Proof: The only if part of the above claim is straightforward to observe. In other words, if there exists a line through points a, b, c, then that line forms the line of sight between the line segments S_1 and S_2.

To prove the if part, consider a line segment L between a point on S_1 and S_2, which does not intersect any of the line segments except S_1 and S_2. This implies that L should pass through the holes $(a-\epsilon,\ a+\epsilon)$, $(b-\epsilon,\ b+\epsilon)$ and $(c-\epsilon,\ c+\epsilon)$. Let the point of intersection on the holes be $a+\delta_1$, $b+\delta_2$ and $c+\delta_3$ respectively. From the condition on the slope of the line, we have $(a+\delta_1)+(b+\delta_2) = 2*(c+\delta_3)$. Also, note that $-\epsilon < \delta_1, \delta_2, \delta_3 < \epsilon$. In the construction, we set the value of ϵ to be $\frac{1}{5}$. Hence, value of $|\delta_1+\delta_2-2*\delta_3| \leq \frac{4}{5} < 1$. This implies that $a+b = 2*c$, since a,b,c were integers. In other words, the line L passes through the points a,b,c. This completes the proof of the claim.

■

4.6 Other Geometric Problems

There are several other geometric problems which can be shown to be 3-SUM hard under sub-quadratic reductions. One of them is, to decide if given a set of triangles, is there a "hole" in the union of triangles(i.e. point in the plane surrounded by some part of some triangle but the point doesn't belong to any triangle). The details of the reduction can be found in the work by Gajentaan and Overmars [2]. Another such problem is a problem from robot motion planning. The problem is to decide, given a set of line segments whether there exists a path from start to finish, where each line can be either translated or rotated. For more details, the reader can refer to the work by Vegter[3] where they give a $O(n^2)$ solution to this problem.

5 Recent Improvements

In the past few years, there have been some improvements on the 3-SUM problem. Baran, Demaine and Patrascu [4] shave a polylog factor in the running time and give a $O\left(\frac{n^2}{\frac{\log^2 n}{\log\log^2 n}}\right)$ time randomized algorithm. In another work by Gronland and Pettie [5], they give a $\tilde{O}(n^{\frac{3}{2}})$ algorithm, but under a non-regular decision tree model. Under the standard RAM model[2], the 3-SUM conjecture still holds true.

Borassi, Crescenzi and Habib [6] also show that some problems are truly quadratic unless the well known SETH hypothesis(as defined in previous lecture) fails.

References

[1] Abboud, Amir, Vassilevska Williams, Virginia. *Popular conjectures imply strong lower bounds for dynamic problems.* 26 September 2014. arXiv:1402.0054 [cs.DS].

[2] In the standard RAM model on w-bit word, access to any cell in the memory location can be done in constant time. Also, the basic operations on words takes constant time.

[2] Gajentaan, Anka, Overmars, Mark. *On a class of $O(n^2)$ problems in computational geometry* . 1990. Computational Geometry Theory and Applications.

[3] Vegter, Gert. *The Visibility Diagram: A Data Structure for Visibility Problems and Motion Planning.* SWAT 1990.

[4] Baran, Ilya, Demaine, Erik, Patrascu, Mihai. *Subquadratic Algorithms for 3SUM.* Algorithmica 2008.

[5] Gronland, Allan, Pettie, Seth. *Threesomes, Degenerates, and Love Triangles.* FOCS 2014.

[6] Borassi, Michele, Crescenzi, Pierluigi, Habib, Michel. *Into the Square - On the Complexity of Quadratic-Time Solvable Problems.* 2014. CoRR abs/1407.4972.