

**Open Problems Column**  
**Edited by William Gasarch**  
**This Issue's Column!**

This issue's Open Problem Column is by William Gasarch and Erik Metz.  
It is on *Generalizing the 3SUM Problem*.

**Request for Columns!**

I invite any reader who has knowledge of some area to contact me and arrange to write a column about open problems in that area. That area can be (1) broad or narrow or anywhere inbetween, and (2) really important or really unimportant or anywhere inbetween.

**Generalizing the 3SUM Problem**  
**By William Gasarch<sup>1</sup> and Erik Metz<sup>2</sup>**

## 1 3SUM-Hardness and Completeness

**Def 1.1** 3SUM is the following problem:

1. Input: A set  $A$  of  $n$  integers.
2. Output: YES if there is  $x, y, z \in A$  such that  $x + y + z = 0$ , NO otherwise.
3. Caveat: The complexity of an algorithm is the number of operations. Hence we count one multiplication, even if the numbers involved are huge, to be one step.

**Def 1.2** An algorithm is *subquadratic* if there exists an  $\epsilon > 0$  such that the algorithm runs in time  $O(n^{2-\epsilon})$ .

There is an  $O(n^2)$  algorithm for 3SUM. Is there a subquadratic algorithm? The consensus is that there is not.

Imagine if we did not have the Cook-Levin theorem, but the consensus was that SAT was hard. We could still define notions of hardness and even completeness. This is what Gajentaan and Overmars [GO12] did in the context of 3SUM where there is no analog to the Cook-Levin Theorem.

---

<sup>1</sup>gasarch@umd.edu

<sup>2</sup>emetz1618@gmail.com

**Def 1.3** Let  $A$  and  $B$  be problems.

1.  $A \leq B$  if, given an oracle for  $B$ , one can solve  $A$  in subquadratic time.
2. A problem  $B$  is 3SUM-*hard* if  $3\text{SUM} \leq B$ .
3. A problem  $B$  is 3SUM-*complete* if  $3\text{SUM} \leq B$  and  $B \leq 3\text{SUM}$ .

Gajentaan and Overmars [GO12] proved many problems 3SUM-complete. See Soss et al. [SEO03] and King [Kin20] for additional 3SUM-complete problems.

## 2 Generalizing 3SUM

**Def 2.1** Let  $a, b, c \in \mathbb{Z}$ . The *abcSUM problem* is as follows: Given a set of  $n$  integers,  $A$ , is there  $x, y, z \in A$  such that  $ax + by + cz = 0$ .

If  $a = 0$  or  $b = 0$  or  $c = 0$  then abcSUM is 2SUM which is in  $O(n)$  time. If  $a, b, c$  share a factor than you can just divide by it. What about the other cases? Henceforth we will assume  $a \neq 0$ ,  $b \neq 0$ , and  $c \neq 0$ . We will also assume that  $a, b, c$  have no common factors.

## 3 abcSUM is $\leq$ 3SUM

**Theorem 3.1** Let  $a, b, c \in \mathbb{Z}$ . Then  $\text{abcSUM} \leq 3\text{SUM}$ .

**Proof:** Assume there is an  $O(n^{2-\epsilon})$  algorithm for 3SUM. Here is an  $O(n^{2-\epsilon})$  algorithm for abcSUM.

1. Input  $A$ .
2. Let

$$A' = \{7ar + 1 : r \in A\} \cup \{7br + 2 : r \in A\} \cup \{7cr - 3 : r \in A\}.$$

3. Run the 3SUM algorithm on  $A'$ . If it says YES, output YES. If it says NO, then output NO.

This algorithm is clearly in  $O(n^{2-\epsilon})$  time. We show that it is correct.

**If alg says YES then there is  $x, y, z \in A$  with  $ax + by + cz = 0$ .**

Assume the algorithm says YES. Then there is an  $x, y, z \in A'$  such that  $x + y + z = 0$ . Let  $x = 7ar_1 + d_1$ ,  $y = 7br_2 + d_2$ ,  $z = 7cr_3 + d_3$  where  $r_1, r_2, r_3 \in A$  and  $d_1, d_2, d_3 \in \{1, 2, -3\}$ .

$$(7ar_1 + d_1) + (7br_2 + d_2) + (7cr_3 + d_3) = 0$$

$$d_1 + d_2 + d_3 \equiv 0 \pmod{7}.$$

By cases one can see that you must have  $\{d_1, d_2, d_3\} = \{1, 2, -3\}$ .

Since  $x + y + z = 0$  we have  $ar_1 + br_2 + cr_3 = 0$ .

**If there is  $x, y, z \in A$  with  $ax + by + cz = 0$  then alg says YES.**

Assume that there is an  $x, y, z \in A$  such that  $ax + by + cz = 0$ . Then

$$(7ax + 1) + (7by + 2) + (7cz - 3) = 7(ax + by + cz) = 0$$

Hence the algorithm will find this triple and say YES. ■

## 4 For Many $a, b, c$ : $3SUM \leq abcSUM$

**Def 4.1** Let  $a, b, c \in \mathbb{Z}$  with  $a, b, c \neq 0$  and  $a, b, c$  have no common factor.  $(a, b, c)$  are *cool* if there exists  $D \in \mathbb{N}$  and  $k_1, k_2, k_3 \in \mathbb{Z}$  (all distinct) such that the following hold:

- $ak_1 + bk_2 + ck_3 = 0$ .
- The only solution to

$$ak' + bk'' + ck''' \equiv 0 \pmod{D}$$

with  $k', k'', k''' \in \{k_1, k_2, k_3\}$  (repeats allowed) is  $k_1, k_2, k_3$ .

**Theorem 4.2** Let  $a, b, c \in \mathbb{Z}$  be cool. Then  $3SUM \leq abcSUM$ .

**Proof:** Assume there is an  $O(n^{2-\epsilon})$  algorithm for abcSUM Here is an  $O(n^{2-\epsilon})$  algorithm for 3SUM.

Let  $D, k_1, k_2, k_3$  be from  $(a, b, c)$  being cool.

1. Input  $A$ .
2. Let

$$A' = \{Dbcr + k_1 : r \in A\} \cup \{Dacr + k_2 : r \in A\} \cup \{Dabr + k_3 : r \in A\}.$$

3. Run the abcSUM algorithm on  $A'$ . If it says YES, output YES. If it says NO, then output NO.

This algorithm is clearly in  $O(n^{2-\epsilon})$  time. We show that it is correct.

**If alg says YES then there is a triple in  $A$  that sums to 0.**

Assume that there is an  $x, y, z \in A'$  such that  $ax + by + cz = 0$ . Let

$$x = DXr_1 + k'.$$

$$y = DYr_2 + k''.$$

$$z = DZr_3 + k'''.$$

where  $X, Y, Z \in \{bc, ac, ab\}$  and  $k', k'', k''' \in \{k_1, k_2, k_3\}$ . In both cases repeats are allowed.

Take the equation  $ax + by + cz = 0 \pmod D$  to get

$$ak' + bk'' + ck''' \equiv 0 \pmod D.$$

Since  $D, k_1, k_2, k_3$  are cool we have that  $k' = k_1, k'' = k_2$ , and  $k''' = k_3$ . Hence we may assume that  $X = bc, Y = ac$ , and  $Z = ab$ . So

$$x = Dbcr_1 + k_1$$

$$y = Dacr_2 + k_2$$

$$z = Dabr_3 + k_3.$$

Since  $ax + by + cz = 0$  we have

$$(Dabcr_1 + ak_1) + (Dabcr_2 + bk_2) + (Dabcr_3 + ck_3) = 0$$

$$Dabc(r_1 + r_2 + r_3) + (ak_1 + bk_2 + ck_3) = 0$$

Since  $D, k_1, k_2, k_3$  is cool,  $ak_1 + bk_2 + ck_3 = 0$ . Hence

$$r_1 + r_2 + r_3 = 0.$$

So we have a triple in  $A$  that sums to 0.

**If there is triple in  $A$  that sums to 0 then alg says YES.**

If  $r_1, r_2, r_3 \in A$  and  $r_1 + r_2 + r_3 = 0$  then

$$x = Dbcr_1 + k_1$$

$$y = Dacr_2 + k_2$$

$$z = Dabr_3 + k_3.$$

are all in  $A'$  and

$$ax + by + cz = Dabc(r_1 + r_2 + r_3) + ak_1 + bk_2 + ck_3 = 0.$$

Hence the algorithm will output YES.

■

## 5 Open Questions

If  $a + b + c = 0$  then  $(a, b, c)$  is not cool (we leave this proof to the reader). Hence Theorem 4.2 will not cover all  $(a, b, c)$ .

1. Show that for all  $(a, b, c)$  with  $a + b + c = 0$ ,  $3\text{SUM} \leq \text{abcSUM}$ .
2. Show that if  $abc \neq 0$  and  $(a, b, c)$  is not cool then  $a + b + c = 0$ ?
3. Disproof either of the above.

## References

- [GO12] Anka Gajentaan and Mark H. Overmars. On a class of  $O(n^2)$  problems in computational geometry. *Computational Geometry*, 45(4):140–152, 2012.  
<https://doi.org/10.1016/j.comgeo.2011.11.006>.
- [Kin20] James King. A survey of 3SUM-hard problems, 2020.  
<https://www.cs.mcgill.ca/~jking/papers/3sumhard.pdf>.
- [SEO03] Michael A. Soss, Jeff Erickson, and Mark H. Overmars. Preprocessing chains for fast dihedral rotations is hard or even impossible. *Computational Geometry*, 26(3):235–246, 2003.  
[https://doi.org/10.1016/S0925-7721\(02\)00156-6](https://doi.org/10.1016/S0925-7721(02)00156-6).