Anonymous Neighbourhood Check

Piotr Kawa, Piotr Szyma

June 2019

Contents

1 Introduction

There are scenarios where two parties would like to find out some information without revealing too much data. Such problem might concern the users' location and in particular checking if the other user is in the neighbourhood. The issue of anonymous localization has already been addressed (1).

In the following paper we propose the method to find out if two users are located in each others neighbourhood without revealing their exact location.

2 Problem analysis

The following section contains information about the concepts used throughout the document.

2.1 Private Set Intersection

Private Set Intersection is a system allowing two parties (denoted in document as Alice and Bob) to find out the intersection of their sets without revealing anything except the intersection itself.

The example of PSI could be a case when two patients would like to find out what illnesses do they have in common. They are only able to find out them — the information about the remaining ones is safe. Figure 1 represents the algorithm.

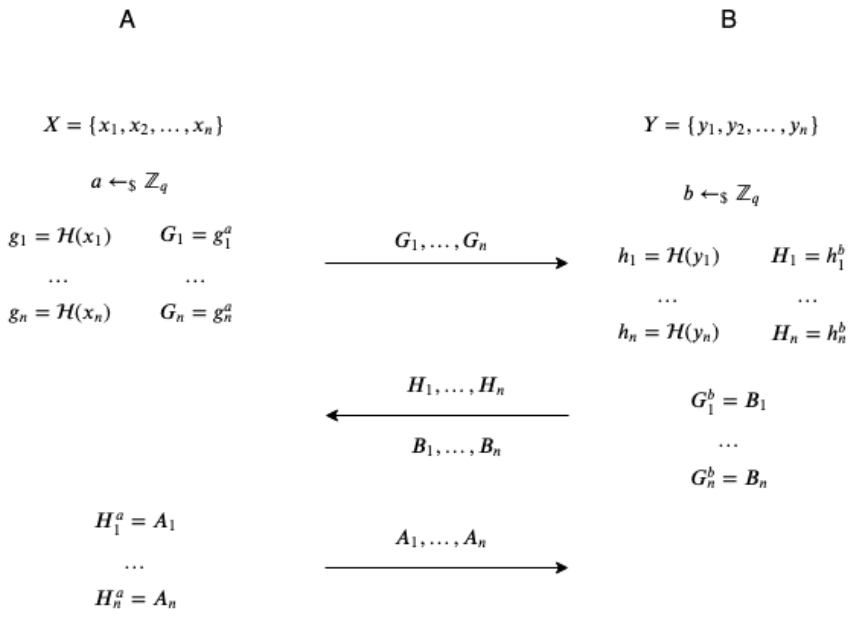

Figure 1: Privacy Set Intersection

Both Alice and Bob have their sets. Each of them picks uniformly at random an ephemeral value from the given group $\mathbb{Z}_q$, where q is public — values are denoted as a and b, respectively for Alice and Bob.

Alice then computes g values. They are created by using some hashing function on each of the elements of the set — the exact hashing method is not specified.

Once the $g_1, \ldots, g_n$ are created, they are used in order to compute another set. It consists of $G_i = g_i^a$ values (a is previously described ephemeral value).

Once the G values are computed, Alice sends them to Bob. Bob performs same operation in order to create set H i.e. hashes y values and then performs exponentiation using ephemeral b. Using the G values received from Alice, Bob computes $B_i = G_i^b$ values and sends them with H values. Once Alice receives H values, she computes set A (in the same way as B set and sends it to Bob).

In this moment, both Bob and Alice got all his and other side sets A and B containing elements raised both to power a and b. Then they create set $A \cap B$ that is intersection of two previous sets. For each values from the intersection that are also in set of their own values raised to a and b they can restore the set (1) of common values.

$$X_{common} = \left\{ x : x^{ab} \in A \cap B \right\} \tag{1}$$

3 Proposed solution

The following section contains information about the way the users' location in represented on the map, how the neighbourhood is defined and the way to anonymously check if both parties are closely.

3.1 Representation of users' location on the map

Map, that is used to denote users' location, is split using a grid — it consists of vertical and horizontal lines. There is no requirement regarding the number of both types lines — the more there are, the more precise the solution. Section 3.1.1 contains example approach.

Intersection points are denoted as the consecutive numbers i.e. moving from top left point that is 1, the number increases. Once the points on a given horizontal level are depleted, the operation starts again for the level below.

The numbers are used in order to represent users' location — each of the rectangles is represented by four numbers denoting the corners.

Assigning users to the adequate rectangle is performed by comparing latitude and longitude coordinates with the coordinates of the areas (denoted later also as panels). Since there are N intersection points, a location consists of 4 points from group $\mathbb{Z}_N$.

	(1, 1)	(1, 2)	(1, 3)	
	(2, 1)	(2, 2)	(2, 3)	
	(3, 1)	(3, 2)	(3, 3)	

	123	124	125	
	223	224	225	
	323	324	325	

Figure 2: Coordinates grid without (left) and with applied mapping (right).

3.1.1 Example

The example approach is for instance territory of Poland. The area of the country is $322,575$ km^2. Since the the maximum length and width are consecutively 649 and 689 km, the area can be split into $649 \times 689 = 447,161$ chunks of 1 km^2.

3.2 Mapping function

In order to compute the chunk's points latitude and longitude of the user's location is required.

The following equations are used in order to compute upper left:

$$P_{ul} = x + y \tag{2}$$

$$x = \lfloor longitude \rfloor + 1 \tag{3}$$

$$y = (90 - \lceil latitude \rceil) * 360 \tag{4}$$

and lower left points of the area:

$$P_{ll} = x + y \tag{5}$$

$$x = \lfloor longitude \rfloor + 1 \tag{6}$$

$$y = (90 - (\lceil latitude \rceil - 1)) * 360 \tag{7}$$

The upper right and lower right points are computed by adding 1 to respectively upper left and lower left points.

3.3 Neighbourhood

Two users (u_1, u_2) are in each others neighbourhood in situation where their locations (rectangles defined by four points) share at least one common corner. That is, having a 3×3 field of 9 rectangles with the center one as user u_1 location, if user u_2 location is in any of rectangles in this 3×3 field of rectangles, then user u_2 is in the neighbourhood of user u_1. In other words — if at least one of points is mutual, then two users are in each others neighbourhood. Having two sets of points U_1 (8) and U_2 (9), the condition for two locations U_1, U_2 to ,,cross" is stated on equation 10. Figure 3 shows examples of relations between locations.

$$U_1 = \left\{ P_{i,1} : 1 \leq i \leq 4 \right\} \tag{8}$$

$$U_2 = \left\{ P_{i,2} : 1 \leq i \leq 4 \right\} \tag{9}$$

$$\mid U_1 \cap U_2 \mid \geq 1 \tag{10}$$

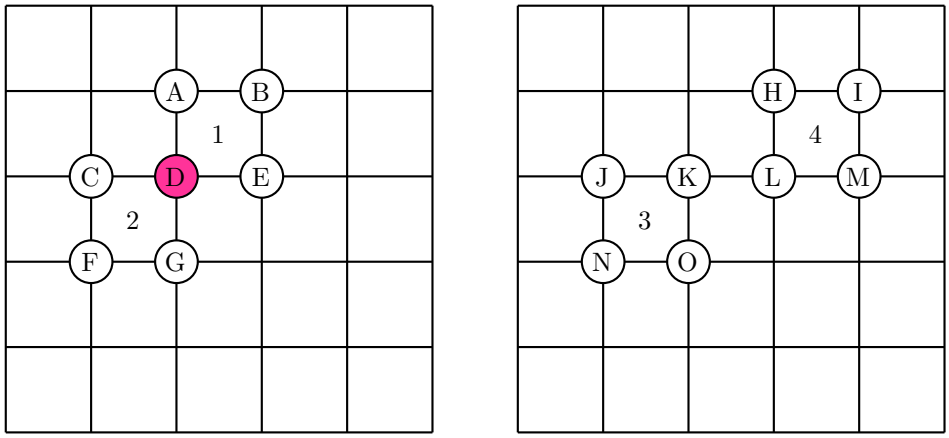

Figure 3: Coordinates grid with possible relations. Locations 1 and 2 are in the neighbourhood (thanks to point D), whereas locations 3 and 4 have no mutual points.

From the geometrical perspective, in case of the same area all four points are the same. Surrounding areas have either two (in case of horizontal and vertical ones) or one (diagonal) mutual points. Figure 4 shows examples of such mutual points.

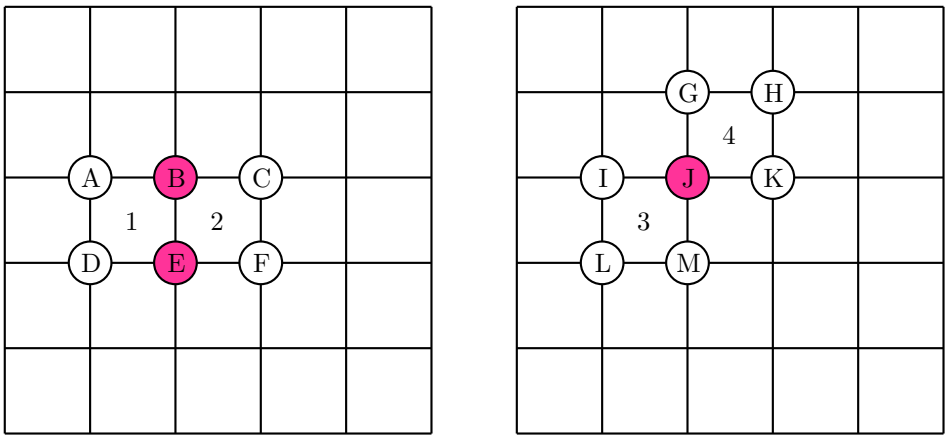

Figure 4: Coordinates grid with example of vertical neighbourhood (left) and diagonal neighbourhood (right).

3.4 Anonymous Neighbourhood Establishment

To anonymously establish the neighbourhood with another party we have to combine the concepts presented in this paper.

1. At first we map a location into set of four points of numbers from Z_q group. The mapping function is described in the section 3.2.

2. Next step is to perform the ,,Privacy Set Intersection" algorithm with the other party that we wanted to share our location with. The ,,PSI" algorithm is described in section 2.1.

3. Based on the number of points that are in common with the other party — in other words based on the way the neighbourhood relation is defined in the section 3.3 — a check if the parties in the neighbourhood or not can be performed.

4 Summary

Presented solution is able to provide an anonymous way of determining if two parties are in each others neighbourhood without revealing their exact location. It is also highly flexible — there are no restrictions regarding the representation of the map. This means that the specificity of the information can be easily changed — the only requirement is altering the number of vertical and horizontal lines.

In addition, proposed solution is not the only possible approach to the problem. In exchange for the presented Private Set Intersection another might be used — for instance Socialist Millionaire Problem (2).

References

[1] A. Franchi, G. Oriolo, and P. Stegagno, *On the solvability of the mutual localization problem with anonymous position measures*, 06 2010, pp. 3193 – 3199.

[2] A. C. Yao, *Protocols for secure computations*, in Proceedings of the 23rd Annual Symposium on Foundations of Computer Science, SFCS '82, Washington, DC, USA, 1982, IEEE Computer Society, pp. 160–164.