Privacy-preserving Judgment of the Intersection for Convex Polygons

Yifei Yao
School of Computer & Communication Engineering,
University of Science & Technology Beijing, Beijing, China
Email: yaoyifei@mail.ustc.edu.cn

Shurong Ning
School of Computer & Communication Engineering,
University of Science & Technology Beijing, Beijing, China
Email: fancyning@163.com

Miaomiao Tian and Wei Yang
National High Performance Computing Center at Hefei, Hefei, China
Email: {qubit, miaotian}@mail.ustc.edu.cn

***Abstract*—As the basic issues of computational geometry, intersection and union of convex polygons can solve lots of problems, such as economy and military affairs. And privacy-preserving judgment of the intersection and union for convex polygons are most popular issues for information security. Traditional method of making the polygons public does not satisfy the requirements of personal privacy. In this paper, a method to compute intersection and union of convex polygons in secure two-party computation (STC) model has been considered, both proportionate partition and unproportionate partition cases are studied. Scan line algorithm is used to figure out the geometry matter, while secret comparison protocol is used for saving the privacy, a series of protocols for this matter is proposed, which combines computational geometry and secure multi-party computation (SMC) technique to achieve the functionality of cooperation calculation without leaking so much privacy. At last, the security, complexity and applicability analysis of the protocols are also discussed.**

***Index Terms*—STC, secret comparison protocol, privacy-preserving geometric computation, polygonal intersection, polygonal union**

I. Introduction

Along with the importance of privacy turning more and more attractive, the secure computation of basic algorithm in each field becomes popular questions. Privacy-preserving techniques provide methods to find important messages correctly in shared data collection. It turns out to be attractively because it can seek more benefit for participants [1]. Meanwhile, secure multi-party computation makes cooperative calculation privately, and prevents participants' data from leaking [2]. Polygonal intersection and union is base of computational geometry and computer graphics, they are of significance both in theory and practice [3]. Many issues need polygonal intersection such as removing hide line, pattern recognition, component position, linearity programming and so on. Meanwhile, polygonal union can help one decide architecturally plane area of ichnography. Methods to compute two objects intersection or union privately and effectively will settle these problems.

In former applications, people always collect the polygons information together and solve it by a trust third party (TTP). But the demand of privacy makes it hard to find such an agency trusted by both partners. Each party wants the result correctly avoiding leaking his information to the other. In this paper, we study how to calculate polygonal intersection and union in STC model. This solution does help in economy and military affairs. For example, a new company hopes to build a shopping mall, it must review if there is another company working at the same area. Both of them want to know weather their orbits meets or not without leaking their own border information. Meanwhile, military affairs also refer to the intersection question often. Fortunately, Reference [4] and [5] indicate that SMC technique can help to achieve the goal.

In this paper, we devise protocols to compute intersection and union of convex polygons approximately, and then analyze their security, complexity and applicability. The paper is organized as follows. In section 2 we describe preliminaries. We introduce the basic comparison protocol detailed in section 3 and present the STC protocol in section 4. Then in section 5 we discuss the protocols complexity and security. At last we conclude the paper in section 6.

II. Preliminaries

A. Secure Multi-party Computation

In a multi-agents network, SMC helps two or more parties complete the synergic calculation without leaking private information. Generally speaking, SMC is a

doi:10.4304/jcp.7.9.2224-2231

distributed cooperation. In this work, each party hold a secret as input, and they want to implement the cooperative computation while knowing nothing about others data except the final result.

Secure two-party computation (STC) was first investigated by A. C. Yao in reference [6]. Then, a general solution for SMC was proposed [4, 5]. From then on, the technology of SMC has already come into more and more domains and many scholars dive into this field, and lots of articles for special use of SMC come into being, such as data mining [7], statistical analysis [8], scientific computation [9, 10], electronic commerce, private information retrieval (PIR), privacy-preserving computation geometry (PPCG) [11, 12], quantum oblivious transfer and so on. Secure multi-party computation for set union and join makes SMC useful in data mining [13, 14]. PIR uses the SMC conception for reference to retrieve answer without leaking other information. Privacy-preserving location determinant of two geometry graphics imports SMC into military affairs [15, 16]. With the rapid development of economy, scientific computation and statistical analysis will use SMC technique as one of the basic security tools.

Reference [5] introduced several applications of SMC by W. Du, and it brought forward correlation and regression analysis problem of privacy-preserving statistical analysis firstly. He gives a solution in two-party instance with his matrix product protocol. Then L. Yehuda studied STC problems in a malicious condition [17].

Computational geometry is a subject studying plane and solid issues, which is important for settling matters in abstract problems. Geometry measurements, which include inner product, convex hulls, location judgment, and so on, are basal in production and living life of society. To achieve the purpose of preserving personal privacy in computational geometry, D. Li designed an approximate convex hulls protocol [18] and Q. Wang proposed a convex hull algorithm for planar point set in [19]. Reference [20] tells us how to determine the meeting points of two intersected circles. Meanwhile, reference [21] gives a method to share unified location with end user privacy control and reference [22] gives a security analysis of the Louis protocol for location privacy. All the research are balancing the privacy and efficiency, a method without any leaking is not really existed, but more efficient ways for limit disclosure of information are more useful and practical.

Previous methods work on a third-party who is trusted by all parties. A TTP can get enough information to complete the calculation and broadcasts the result. But the hypothesis itself is insecure and unpractical. Therefore, an executable protocol which can preserve participants' privacy becomes more and more dramatically. It is known that any secure computation problem can be solved by a circuit protocol, but the size of the corresponding circuit is always too large to realize. So investigators choose to design special protocol for special use instead of praying for a third party's keeping secret.

B. Secret Comparison Protocol

In 1982, A.C. Yao brought forward the famous millionaires problem: two millionaires, say Alice and Bob, want to know which is richer, without revealing their respective wealth. To begin with, Alice and Bob need a public-key cryptographic system which is strong, for example RSA, and assume that both of their belongings are in a certain integral range. Alice is worth millions, and Bob millions, they execute the millionaire protocol through public-key system and mode computation [6]. If the result is that Bob receives a data inosculating his J th digital, then $I \geq J$, otherwise $I < J$. At last, Bob sends Alice the result. In order to avoid cheating, each party initiates the protocol once for peace.

After that, another method for two parties comparing is brought forward. There are three parties taking part in the protocol: A, B, and an oblivious third party C who helps A and B to check if their private value a and b are equivalent or not. The method validates its security by computational undistinguished through homomorphism encryption and Φ -hiding assumption. It returns which one is greater or equal to the other, while Yao's method couldn't returns the equality message. This protocol is complex in computation and safe to resist decoding, it reduces the communication of random data perturbation techniques in Yao's method. Recently, reference [23] put forward an efficient protocol for private comparison problem, and this issue must be a popular problem for several years [24].

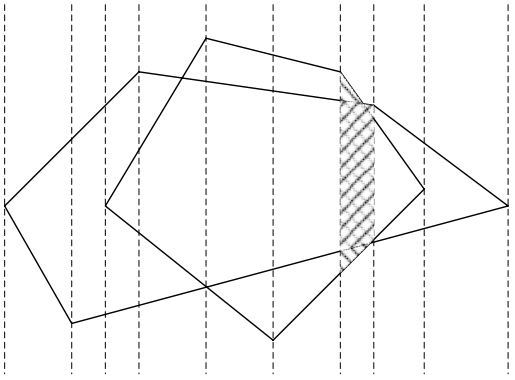


Figure 1. Areas decided by the peaks of convex polygons.

C. Models

Computational model: Generally speaking, there exist potential malicious attacks against any multi-party protocol [17]. In this paper, we study the problem under a semi-honest model, in which each semi-honest party follows the protocol with exception that he keeps a record of all its intermediate computations, and he will never try to intermit or disturb with dummy data [5]. The model is practical and useful, because everybody in the cooperation expects the right result rather than others private information.

Security model: We name I_A and I_B as the input instance of Alice and Bob, and name O_A and O_B as the corresponding output. C represents the computation

executed by the two partners, then (1) establishes. A protocol for executing C is secure when it satisfies two conditions as follow:

1. There is an infinity set (2) for (3) and (4).

$$(O_A, O_B) = C(I_A, I_B). \quad (1)$$

$$D_A = \{(IA_i, OA_i) \mid i = 1, 2, \cdots\}. \quad (2)$$

$$(O_A', O_B) = C(I_A', I_B). \quad (3)$$

$$\forall (I_A', O_A') \in D_A. \quad (4)$$

2. There is an infinity set (5) for (6) and (7).

$$D_B = \{(IB_i, OB_i) \mid i = 1, 2, \cdots\}. \quad (5)$$

$$(O_A, O_B') = C(I_A, I_B'). \quad (6)$$

$$\forall (I_B', O_B') \in D_B. \quad (7)$$

Apparently, the more closely that I_A' has association with I_A, the more information Bob will get about Alice, and vice versa. In the execution, it's inevitable to leak some message. But our protocol is robust to some extent. The adversary can obtain something through security analysis, but it's not enough for him to get the certain value. Although the protocol is not zero-knowledge, it is a desirable way to achieve high efficiency.

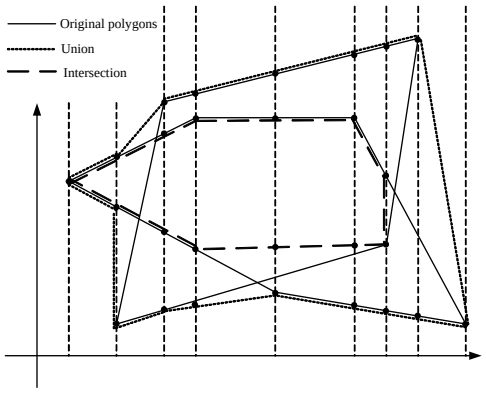


Figure 2. Example of unproportional instance - subset.

D. Related Algorithms in Computational Geometry

Irreciprocal Protocol in Unproportionate Partition: Firstly, we find out the maximal and minimal x-coordinate values, noted as a and b, then divide into k equidistant bands perpendicularly between a and b. The k bands form a memory serial, and we distribute the n points of set S into the memory serial. At last, we pick the maximal and minimal y-coordinate values of each band and save them as set S^*. S^* has $2k+4$ points at most, we construct its convex to form the approximate outline.

Each sector and a convex polygon intersect into a quadrangle. It is to say that P and Q intersect into a quadrangle in each sector. We will find it in $O(1)$. Then, a scanning work in linear time can set up these fragments. At last, we pick them up and move out the void peaks at the margin (Fig.1).

Theorem 1: The intersection of convex polygons L and M will be find in (8).

$$\theta(L+M). \quad (8)$$

The correctness of the theorem 1 can be found in reference [1]. An example is shown in Fig.2.

Irreciprocal Protocol in Proportionate Partition: In order to avoid the irregular workload of unproportionate partition, we can use proportionate partition instead. It likes the one above but equidistant and adjustable as demand. An example is shown in Fig.3.

Basic Generating Set Protocol: The method in irreciprocal protocol in unproportionate partition forms a proper subset of approximate convex as Fig.2. We modify it to get its generating convex. To achieve the goal, reference [1] uses the two outermost points in the same ordinate slip instead of maximum point P. Obviously, it generates a convex including proper subset. Similar to the error analysis, the point in our approximate convex but not proper subset does be in (9) from the proper convex.

$$(x_{\max} - x_{\min})/k. \quad (9)$$

We modify it to form a protocol computing the generating set of intersection and union of polygons as below.

Firstly, we divide the area into equidistant intervals. Secondly, the maximum and minimum ordinate of the two polygons in each strip will be compared. At last, the outermost point with the same ordinate is outputted instead of primary maximum point P. An example is showed as Fig.3.

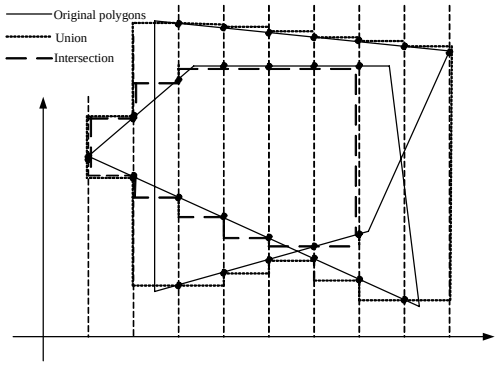


Figure 3. Example of proportional instance - generating set.

III. BUILDING BLOCKS

In this section, we introduce the secure building blocks. It performs comparison on one scanning bean. It's a basic tool for the latter protocols.

We assume that P_{high} and P_{low} belong to Alice, Q_{high} and Q_{low} belong to Bob. They are on the same scanning bean, and are ranked by their y-coordinate.

There will be four instances appearing on each bean as follows.

Result 1: $P_{\text{high}} > Q_{\text{high}}$ and $P_{\text{low}} < Q_{\text{low}}$: Then Q_{high} and Q_{low} belong to polygonal intersection, P_{high} and P_{low} belong to polygonal union (Fig.4).

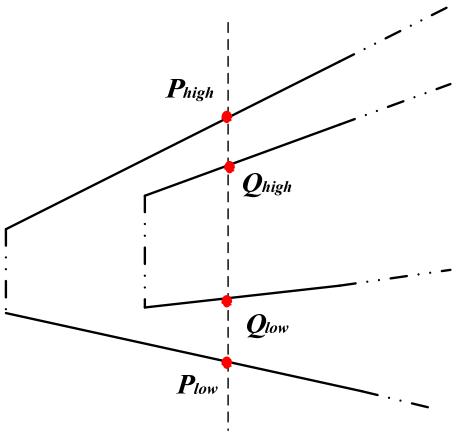


Figure 4. Basic comparison protocol: Result 1.

Result 2: $P_{\text{high}} < Q_{\text{high}}$ and $P_{low} < Q_{low}$:

Result 2.1: if $P_{\text{high}} > Q_{\text{low}}$ then P_{high} and Q_{low} belong to polygonal intersection, Q_{high} and P_{low} belong to polygonal union (Fig.5).

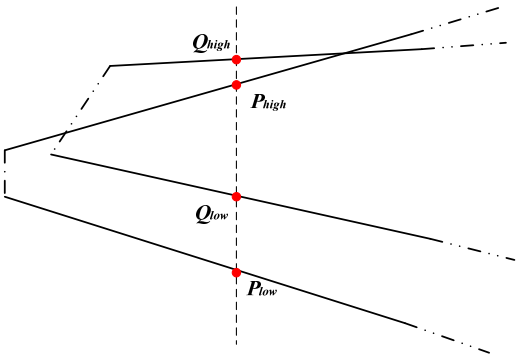


Figure 5. Basic comparison protocol: Result 2.1.

Result 2.2: if $P_{\text{high}} < Q_{\text{low}}$ then no one on this scanning bean belongs to the intersection, the four points all belong to the union (Fig.6).

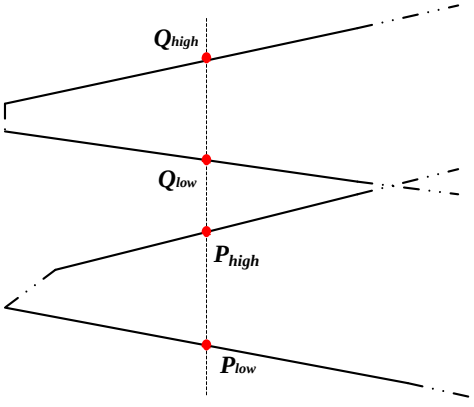


Figure 6. Basic comparison protocol: Result 2.2.

Result 3: $P_{\text{high}} > Q_{\text{high}}$ and $P_{\text{low}} > Q_{\text{low}}$:

Result 3.1: if $Q_{\text{high}} > P_{\text{low}}$ then Q_{high} and P_{low} belong to the polygonal intersection, P_{high} and Q_{low} belong to the polygonal union (Fig.7).

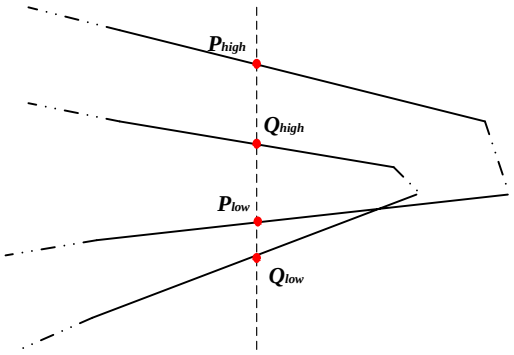


Figure 7. Basic comparison protocol: Result 3.1.

Result 3.2: if $Q_{\text{high}} < P_{\text{low}}$ then no one on this scanning bean belongs to the intersection, the four points all belong to the union (Fig.8).

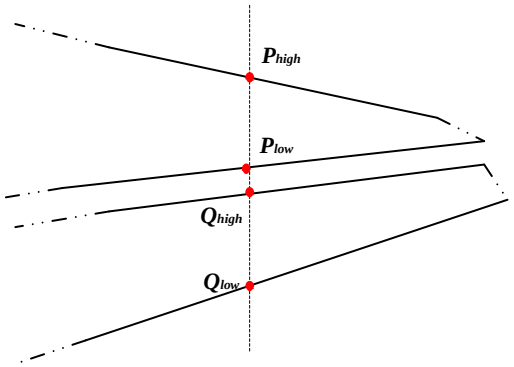


Figure 8. Basic comparison protocol: Result 3.2.

Result 4: $P_{\text{high}} < Q_{\text{high}}$ and $P_{\text{low}} > Q_{\text{low}}$: Then P_{high} and P_{low} belong to polygonal intersection, Q_{high} and Q_{low} belong to polygonal union (Fig.9).

We summarize the basic comparison protocol as below:

Protocol 1: Basic Comparison Protocol

Input: Alice has P_{high} and P_{low} , while Bob has Q_{high} and Q_{low} at each bargained scanning bean.

Output: Both Alice and Bob know which of his point is on the polygonal borderline with no information leaking to the other.

Alice cooperates with Bob to compare $\left(P_{\text{high}}, Q_{\text{high}}\right)$ and $\left(P_{\text{low}}, Q_{\text{low}}\right)$ using the secret comparison protocol in 2.2.

Case 1: if $P_{\text{high}} > Q_{\text{high}}$ and $P_{\text{low}} < Q_{\text{low}}$ then we get result 1 and terminate.

Case 2: if $P_{\text{high}} < Q_{\text{high}}$ and $P_{low} < Q_{low}$ then we continue to compare P_{high} and Q_{low} : if $P_{\text{high}} > Q_{\text{low}}$ then we get result 2.1 , else we get result 2.2, and terminate.

Case 3: if $P_{\text{high}} > Q_{\text{high}}$ and $P_{low} > Q_{low}$ then we continue to compare Q_{high} and P_{low} : if $Q_{\text{high}} > P_{\text{low}}$ then we get result 3.1 , else we get result 3.2, and terminate.

Case 4: if $P_{\text{high}} < Q_{\text{high}}$ and $P_{\text{low}} > Q_{\text{low}}$ then we get result 4 and terminate.

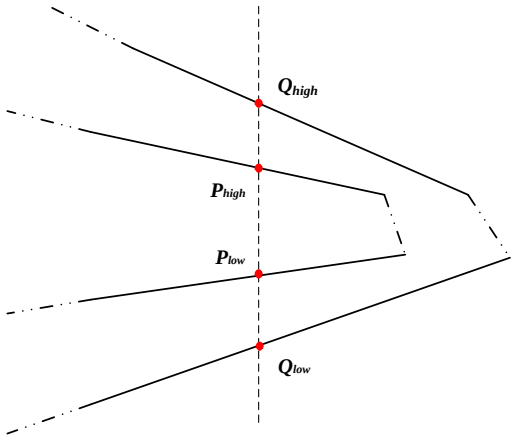


Figure 9. Basic comparison protocol: Result 4.

Theorem 2: Protocol 1 could complete the comparison on one scanning bean without compromising privacy.

Proof: We get the correctness from the figures above (Fig.4 - Fig.9).

For security, we study what is leaked through the process. On each scanning bean, both the parities get the result which point belongs to the convex borderline, but the value is secret to the other side. Because they will compare two times at most, neither can nose out the others information by repetitious execution. And what they got can be analyzed from the view though the execution without any other assistant. It does satisfy the demand of security model.

Theorem 3: Protocol 1 has complexity $O(1)$ times of secret comparison protocol.

Proof: On each scanning bean, they will compare two times at most. So, they can finish the progress in $O(1)$ times of secret compare protocol.

IV. Protocol to Compute Intersection and Union of Convex Polygons Approximately in STC

A. Secure Protocol for Approximate Intersection of Convex Polygons in STC

In this section, we discuss the secure protocols for approximate intersection of two polygons in unproportionate and proportionate partition.

Protocol in unproportionate partition: a protocol for unproportinate partition is proposed below.

Protocol 2: Secure Two-Party Protocol for Approximate Intersection of Two Polygons in Unproportionate Partition.

Input: Alices and Bobs private convex polygons

Output: the approximate intersection of the two polygons

Step1: Alice and Bob announce to each other the x-coordinate of each peak or selected x-coordinates to form the unproportionate partition scanning beans.

Step2: On each scanning bean, Alice has P_{high} and P_{low} Bob has Q_{high} and Q_{low} . They invoke the Basic Comparison Protocol to know which point is on their approximate intersection without leaking any other information.

Step3: They repeat step 2 until all the scanning beans are finished.

The benefit of this protocol is to avoid computing the actual coordinate of the point of intersection. The point of intersection will leak apex message more or less, and it is dangerous to leak the outline of polygons when scanning frequently. This protocol reduces leaking information by the way of avoiding calculating the apex. Alice registers her point if it is on the outline, otherwise, she only knows that the point on this scanning bean belongs to Bob without getting other message. If there is only one apex, we look it on as the maximum and minimum value simultaneously.

Protocol in proportionate partition: Protocol under proportionate partition is similar to that of unproportionate partition, the only difference is in step 1, and we modify it as below.

Step1: Alice and Bob choose their own greatest and least x-coordinate to compare securely for announcing the maximum and minimum value, and negotiate about the number of regions. Then, they carve up $n+1$ scanning beans proportionately between the two values.

Thus it can be seen that the complexity of this protocol is correlative to the partition number n . Although reducing partition number will preserve parties' privacy better, it descends precision meanwhile.

Protocol of generating set: The protocol of generating set is similar to that of subset, the difference is in step2. We get it when change Basic Comparison Protocol into Basic Generating Set protocol.

B. Secure Protocol for Approximate Union of Polygons

In this section, we discuss the secure protocols for approximate union of two polygons in unproportionate and proportionate partition.

Protocol in unproportionate partition: a protocol for unproportinate partition is proposed below.

Protocol 3: Secure Two-Party Protocol for Approximate Union of Two Polygons in Unproportionate Partition

Input: Alices and Bobs private convex polygons

Output: the approximate union of the two polygons

Step1: Alice and Bob announce their x-coordinate of each peak or selected x-coordinates to form the unproportionate partition scanning bean.

Step2: On each scanning bean, Alice has P_{high} and P_{low} , and Bob has Q_{high} and Q_{low} . They invoke the Basic Comparison Protocol to know which point is on their approximate union without leaking any other information.

Alice or Bob only knows if her/his point is the maximum or minimum value but nothing else.

Step3: They carry out step 2 repeatedly until all the scanning beans are finished.

Protocol in proportionate partition: Protocol under proportionate partition is similar to that of unproportionate partition, as protocol in protocol in proportionate partition is similar to that in protocol in unproportionate partition.

Protocol of generating set: The protocol of generating set is similar to that of subset, as protocol in protocol of generating set is similar to that in protocol in unproportionate partition.

V. Analysis

In this section, we analyze the complexity and security of the protocols.

A. Complexity Analysis

Conclusion 4: Secure two-party protocol to compute intersection or union of convex polygons in unproportionate partition has time and communication complexity $O(m+n)$ times of Basic Comparison Protocol. The corresponding protocols in proportionate partition has time and communication complexity $O(l)$ times of Basic Comparison Protocol, where l is the number of regions the both bargained on. Protocol for generating set likes the fore type.

For Basic Comparison Protocol, we get its security in section 3 and it can be finished in $O(1)$ times secret comparison protocol. Because of the comparability of the protocols, we take protocol 2 as example. In Step 1, Alice and Bob decide the partition of the scanning area, they use $O(1)$ times exchanging message to announce their x-coordinate. In Step 2, it cost them $O(1)$ times of secret comparison problem. In Step 3, they need $O(m+n)$ times of Basic Compare Protocol to scan all the beans. Meanwhile, the protocol in unproportionate partition has complexity $O(l)$ times of comparison, where l is the number of regions the both negotiate about. For they compare once at each scanning bean.

B. Security Analysis

Conclusion 5: Protocol 1 (Protocol 2, Protocol 3) can execute securely without leaking privacy.

Now, we analyze the message leaked at each case in Basic Comparison Protocol. On each scanning bean, Alice gets to know if her P_{high} and P_{low} are on the outline. In case 1, Alice sees Q_{high} and Q_{low} of Bobs are between P_{high} and P_{low}, Bob gets that his Q_{high} and Q_{low} are not on the outline and $P_{\text{high}} > Q_{\text{high}}$, $P_{\text{low}} < Q_{\text{low}}$ thereby. In case 2, if 2.1 happens, they see P_{high} is seated between Q_{high} and Q_{low}, and Q_{low} is greater than P_{high}. The rest may be deduced by analogy. So, Alice or Bob only knows the relative position of her/his point and the others but not the value.

Considering this problem, some message is predetermined to leak out. If anyone knows his point is on the outline, he immediately sees the others corresponding point is not on the outline. The acceptance or rejection indeed discloses some information about big or small on the same scanning bean, but it is inescapable. Our method can not guarantee this kind of message but only prevent from leaking any needless information.

Because the four points is independence and there is no rule between them, neither can analyze to know the others information through the secure intersection or union protocol. This does preserve the parties' privacy.

C. Applicability Analysis

The protocol of this paper is an approximate algorithm to calculate an outline. Although approximation avoids leaking apex message when computing intersection and union, it is at the cost of precision in calculation. Especially in proportional partition, our result is anamorphic if their figures change suddenly in some area as Fig.10. Therefore, the scheme in this paper doesn't adapt to work of high precision.

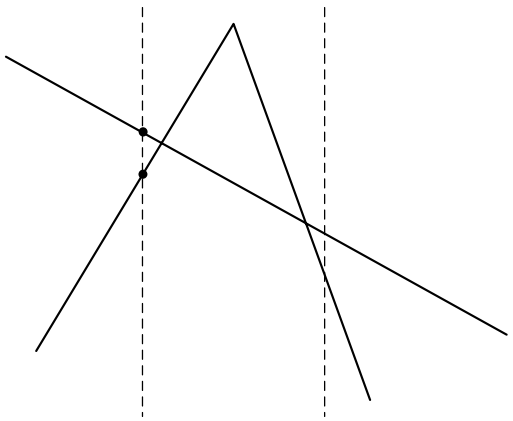


Figure 10. Mutant Example.

Meanwhile, the point on the outline belongs to one party but not shared by the both, so it can not used for computing acreage of intersection or union.

VI. Summary

Privacy-preserving computational geometry is important for secure multi-party computation, it offers basic tool to calculate conveniently. It is useful in science research and engineering technology. The intersection and union of convex polygons are basic issues in computational geometry, and the demand of privacy-preserving calls on secure protocols for special fields. We have proposed protocols to compute approximate intersection and union of convex polygons in STC model.

Detailed analyses about security and complexity are also presented. We tie in computational geometry and SMC technique rationally to solve the problem. By the help of secret comparison, the protocols use Basic Compare Protocol as sub-protocol and gain in privacy and efficiency at the price of precision appropriately. Along with the development of SMC, our future work would like to settle the problem in more complex settings, such as multi-party model, malicious behavior and so on.

ACKNOWLEDGMENT

The authors wish to thank Professor L. Huang, for providing excellent notes of the discussion. We would like to thank the participants in the National High Performance Computing Center for their helpful comments. And we are very grateful to Professor Y. Luo at Department of Computer Science and Technology at Anhui Normal University for useful comments and suggesting some corrections. This work is jointly supported by NSFC under Grant No. 60903067 and Funding Project for Beijing Excellent Talents Training under Grant No. 2011D009006000004.

REFERENCES

[1] V. A. Oleshchuk, V. Zadorozhny, Secure Multi-party Computations and Privacy Preservation: Results and Open Problems, *Telektronikk. Norway*, pp. 20–26, February 2007.

[2] Goldreich, *Secure Multi-party Computation (working draft)*. Available from www.wisdom.weizmann.ac.il/home/oded/public html/foc.html, 1998.

[3] M. Berg, O. Cheong, and M. Kreveld, *Computational Geometry: Algorithms and Applications*, 3rd ed. Springer-Verlag Berlin, Heidelberg, 2008, pp.45–62. doi: 10.1007/978-3-540-77974-2.

[4] S. Li, and Y. Dai, Secure Two-Party Computational Geometry, *Journal of Computer Science and Technology. Beijing, China*, vol. 20(2), pp. 259–263, 2005.

[5] W. Du, and Z. Zhan, A Practical Approach to Solve Secure Multi-party Computation Problems, *New Security Paradigms Workshop. Beach Virginia, USA*, pp. 127–135, September 2002.

[6] A. C. Yao, Protocol for Secure Computations (extended abstract), *21st Annual IEEE Symposium on the Foundations of Computer Science. IEEE Press, New York, USA*, pp. 160–164, 1982.

[7] C. Clifton, M. Kantarcioglou, Xiadong. Lin, and M. Y. Zhu, Tools for Privacy Preserving Distributed Data Mining, *SIGKDD Explorations Newsletter. New York, USA*, vol. 4, Issue 2, December 2002.

[8] Y. Yao, L. Huang, W. Yang, Y. Luo, W. Jing, and W. Xu, "Privacy-preserving Technology and Its Applications in Statistics Measurements," *The Second International Conference on Scalable Information Systems. Suzhou. China*, article 74, June 2007.

[9] W. Du, and J. A. Mikhail, Privacy-preserving Cooperative Scientific Computation, *14th IEEE Computer Security Foundations Workshop. Nova Scotia, Canada*, pp. 273–282, 2001.

[10] Y. Yao, L. Huang, and Y. Luo, Privacy-preserving matrix rank computation and its applications, *Chinese Journal of Electronics. Beijing, China,*, vol. 17(3), pp. 481–486, 2008.

[11] R. Dowsley, J. Graaf, D. Marques, and C. A. Nascimento, A Two-Party Protocol with Trusted Initializer for Computing the Inner Product, *Information Security Applications. Brazil*, vol. 6513, pp. 337–350, 2011. doi: 10.1007/978-3-642-17955-6_25.

[12] D. Eppstein, M. T. Goodrich, and R. Tamassia, Privacy-preserving data-oblivious geometric algorithms for geographic data, *GIS '10 Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems. New York. USA*, pp. 13–22, 2010. doi: 10.1145/1869790.1869796.

[13] B. Hawashin, F. Fotouhi, and T. M. Truta, A privacy preserving efficient protocol for semantic similarity join using long string attributes, *PAIS '11 Proceedings of the 4th International Workshop on Privacy and Anonymity in the Information Society. New York, USA*, article 6, 2011. doi:10.1145/1971690.1971696.

[14] J. Camenisch, and G. M. Zaverucha, Private Intersection of Certified Sets, *Financial Cryptography and Data Security. Lecture Notes in Computer Science*. Berlin, Heidelberg, vol. 5628, pp. 108–127, 2009. doi: 10.1007/978-3-642-03549-4_7.

[15] M. Hardt, and K. Talwar, On the geometry of differential privacy, *STOC '10 Proceedings of the 42nd ACM symposium on Theory of computing. New York, USA*, pp. 705–714, 2010.

[16] Y. Sun, H. Sun, H. Zhang, and Q. Wen, A Secure Protocol for Point-Segment Position Problem, *Web Information Systems and Mining. Lecture Notes in Computer Science*, vol. 6318, pp. 212–219, October 2010. doi: 10.1007/978-3-642-16515-3_27.

[17] L. Yehuda, and P. Benny, An Efficient Protocol for Secure Two-Party Computation in the Presence of Malicious Adversaries, *26th annual international conference on Advances in Cryptology. Barcelona, Spain*, pp. 52–78, 2007.

[18] D. Li, L. Huang, W. Yang, Y. Zhu, Y. Luo, and L. Li, A Practical Solution for Privacy-Preserving Approximate Convex Hulls Problem, WRI *International Conference on Communications and Mobile Computing. Kunming, USA*, vol. 3, pp. 539–544, January 2009.

[19] Q. Wang, and Y. Zhang, A Convex Hull Algorithm for Planar Point Set Based on Privacy Protecting, *First International Workshop on Education Technology and Computer Science. Wuhan. China*, vol. 3, pp. 434–437, March 2009.

[20] Y. Ye, L. Huang, W. Yang, and Z. Zhou, A Secure Protocol for Determining the Meeting Points of Two Intersected Circles, *International Conference on Information Technology and Computer Science. Kiev, Ukraine*, vol. 2, pp. 40–44, July 2009.

[21] J. Xie. and S. Wang, *A unified location sharing service with end user privacy control*, 1st ed., vol. 16. Issue 2. Bell Labs Technical Journal Special Issue: Application, 2011, pp.5–20. doi: 10.1002/bltj.20499.

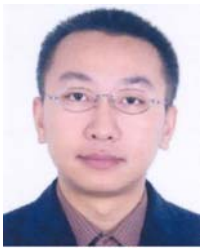
[22] A. Gupta, M. Saini, and A. Mathuria, Security analysis of the Louis protocol for location privacy, *Communication Systems and Networks and Workshops. Bangalore. India*, pp. 1–8, January 2009. doi: 10.1109/2009.4808858.

[23] Y. Luo, L. Huang, W. Yang, and W. Xu, An Efficient Protocol for Private Comparison Problem, *Chinese Journal of Electronics. Beijing. China*, vol. 18(2), pp. 205–209, April 2009.

[24] J. Qin, Z. Zhang, D. Feng, and B. Li, A Protocol of Comparing Information without Leaking, *Journal of Software. Beijing, China*, vol. 15(3), pp. 421–427, 2004.

Yifei Yao was born in Jilin Province, China, in 1981. She received the Ph.D. degree from the Department of Computer Science and Technology, University of Science and Technology of China in 2008. She is currently a lecturer of the School of Computer and Communication Engineering at University of Science and Technology of Beijing. Her major research interests are information security and distributed computing. (Email:yaoyifei@mail.ustc.edu.cn).

Shurong Ning was born in Shanxi Province, China, in 1976. She received the Ph.D. degree from Beijing Institute of Technology in 2006. She is currently a professor of the School of Computer and Communication Engineering at University of Science and Technology of Beijing. Her major research interests are artificial life, intelligent control and computer animation. (Email: fancyning@163.com).



Wei Yang was born in Anhui Province, China, in 1978. He received the Ph.D. degree from the Department of Computer Science and Technology, University of Science and Technology of China in 2007. He is currently a lecturer of the Department of Computer Science and Technology at University of Science and Technology of China. His major research interests are infromation security and quantum information. (Email: qubit@ustc.edu.cn).

Miaomiao Tian was born in Anhui Province, China, in 1987. He received the Master degree from the Department of Computer Science and Technology, University of Science and Technology of China in 2010. He is currently a Ph. D. candidate of the Department of Computer Science and Technology at University of Science and Technology of China. His major research interests are information security. (Email: miaotian@mail.ustc.edu.cn).