TABLE II

THE GRAPH DATASETS USED IN OUR EXPERIMENTS

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# *Methods to Compare:* Since this is the first work to address the CSD querying problem over encrypted graphs, we compare our method with the one over unencrypted graphs. We implement such a method following the state-of-the-art method over plaintext graphs introduced in [2]. The only difference is that we construct 2HCLI over the original graph, instead of an overlay graph. As a result, our implementation has a higher query efficiency but leads to a higher complexity of the index construction.

1. *Query Sets:* We randomly generate 200 queries over each dataset. The origin *s* and destination *t* in each query are also randomly selected. The cost constraint *θ* for each *(s, t)* pair is set as follows. We denote the *lower* bound *cmin* as the minimum cost of all paths from *s* to *t*, and the *upper* bound *cmax* as the minimum cost of the paths with the shortest distance from *s* to *t*. If the cost constraint *θ < cmin*, there will be no feasible answer to the query; and if the cost constraint *θ > cmax*, the shortest distance is always a valid answer to the query. To mitigate the impact of *θ* on the performance, we randomly choose 50 values of *θ* for each query, which falls in the interval *cmin, cmax* .

[ ]

Another important parameter is *α*, which determines the approximation guarantees of *α*-CSD queries. Since *α* is a constant value for all queries, we view it as a system parameter rather than part of specific queries. In order to achieve a balance between query accuracy and system efficiency, we set

TABLE III

SUMMARY OF INDEX CONSTRUCTION COST



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | |  | |
|  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

TABLE IV

THE QUERY TOKEN GENERATION TIME FOR DIFFERENT *dθ*



|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

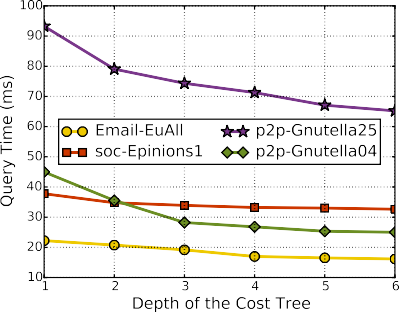


Fig. 7. The query time over encrypted 2HCLI with varying *dθ* .

# encrypted graphs is slightly higher than the one of unen- crypted graphs. Thus, the key point of improving the index construction efficiency over an encrypted graph is accelerating the process of constructing the plain 2HCLI of that graph. We leave this attempt as the future work.

*2) Query Token Generation:* The construction of query tokens is independent of specific graphs, we now analyze

# the size and generation time of a query token. The query

the approximation ratio *α* = 1*.*5 for all queries.

token mainly consists of 5 elements, namely *Sout,s*

, *Tout,s*,

1. *Evaluation of Secure 2HCLI and Query Token*
   1. *Index Size and Construction Time:* The index construc- tion of the graph is a one-time and offline computation. This process consists of two steps: one is constructing the plain 2HCLI, which is the same as the index construction process of the original plain CSD querying, and the other is encrypting the plain 2HCLI, which is the focus of this paper. Therefore, we consider the outputs of the first step as the index of unencrypted graph.

The index size and construction time are depicted in Table III. Note that the index size and construction time of different datasets have a great difference, which is mainly caused by the difference in graph topologies. Different from the original shortest distance query, where there is only one shortest path between any two vertices, in the CSD querying problem, there usually exist multiple constrained shortest paths between any two vertices. Intuitively, a dense graph may bear a higher index construction cost than a sparse one.

In general, the size of each encrypted index is roughly

6 larger than that of the corresponding plain index. The most important observation is that the index construction time of

×

*Sin,t* , *Tin,t* , and *Tθ* . Each of the first 4 elements has a length

of 16 bytes. Since the size of each ORE ciphertext is 16 bytes, a cost tree *Tθ* whose depth is *dθ* has a size of 16 *(*2*dθ* 1*)* bytes. Therefore, the total size of a query token is 16 *(*2*dθ* 3*)* bytes. Since *dθ* is a relatively small value, the size of a query token is usually less than 1 KB. The query token generation time with varying *dθ* is depicted in Table IV. Although the query token generation time increases significantly with *dθ* , the time cost is moderate for general cases (e.g., when *dθ* ≤ 6).

× +

× −

1. *Evaluation of Query Efficiency and Accuracy*
   1. *Query Efficiency:* To evaluate the query efficiency, for each *θ* , we generate the cost constraint tree with a different depth *dθ* . The query time is defined as the time interval from the submission of a query token to the receival of its query results. We compute the average query time of 200 queries.

The average query time with varying *dθ* over the encrypted 2HCLI is depicted in Fig. 7, where *dθ* increases from 1 to 6. We can see that the query time varies a lot for different graph datasets. For each dataset, increasing *dθ* can result in a decrease in the query time. This is because a larger *dθ* can filter out more distance pairs exceeding the cost constraint

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | |  |  |  |
|  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Fig. 8. The query time over the plain 2HCLI and the encrypted 2HCLI (*dθ* = 6).



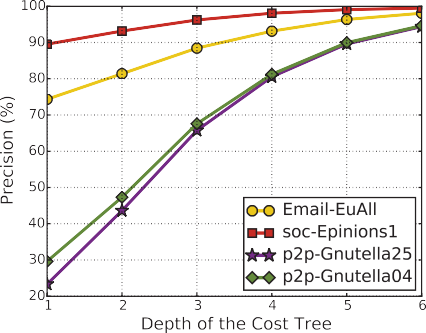


Fig. 9. The query precision for different depth *dθ* of the cost tree.

# and thereby reduce the number of candidates for distance computation using SWHE, which is the dominant operation in time consumption.

Fig. 8 presents the query time in the plain and encrypted scenarios for different datasets. The query time over the encrypted 2HCLI is higher than that over the plain 2HCLI because of the time-consuming operations on ciphertexts (e.g., the cost filtering and distance computation). Also, the time complexity of these operations is closely related to the size of a graph index listed in Table III, which leads to the difference among four datasets in Fig. 8.

* 1. *Query Accuracy:* In Connor, there are two components that affect the query accuracy, namely the tree-based cipher- texts comparison and the distance computation. The former may keep some distance pairs that do not satisfy the cost constraint in the candidate set *Y* , while the latter leverages the property of SWHE to obtain an approximate, but not exact, shortest distance based on all candidates in *Y* .

# We use the well-known metric *Precision* (*P*) to evaluate the accuracy of the cost constraint filtering process. *P* = *Tp*+*Fp* , where *Tp* and *Fp* represent the numbers of distance pairs

*Tp*

in *Y* whose costs truly satisfy or exceed the cost constraint, respectively. We use the same query as introduced above, and

compute the *P* for each query. Finally, we can obtain the average precision ¯ for all the queries.

*P*

Fig. 9 presents the relationship between the query precision *P*¯ and the depth of the cost constraint tree *dθ* over different datasets. We can see that for all the datasets, ¯ increases with *dθ* , because the cost constraint tree with a larger depth *dθ* helps us to detect constraint violations with a higher probability, as discussed in Section VI. In particular, *P*¯ is more than 94% for all datasets when *dθ* = 6.

*P*

Fig. 10. The CDF of deviation rate for different query (*dθ* = 6).

To evaluate the accuracy of the final query results, we propose a metric named the *deviation rate*. Let *re* and *rp* be the query results returned by Connor and the algo- rithm over the corresponding plain graphs, respectively. Then, we define the *deviation rate ξ re/rp*, which indicates how far *re* deviates from *rp*. Obviously, a *deviation rate* closer to 1 depicts more accurate query results.

=

Fig. 10 presents the cumulative distribution func- tions (CDFs) of the *deviation rate* over the dataset p2p-Gnutella04. We can see that *ξ* is larger than 0*.*90 for over 80% of the query results, and larger than 0*.*73 in the worst cases. Therefore, Connor is capable of achieving a relatively high accuracy with moderate computation complexity.

IX. CONCLUSION

# In this paper, we have presented Connor, the first graph encryption scheme that enables the cloud-based approximate CSD queries. In particular, we proposed a tree-based cipher- texts comparison protocol for cost constraint filtering with con- trolled disclosure. The security analysis showed that Connor could achieve the CQA2-security. We implemented a prototype and evaluated the performance using the real-world graph datasets. The evaluation results demonstrated the effectiveness of Connor. In the future work, we plan to design techniques to support dynamic index updates.

REFERENCES

1. X. Meng, S. Kamara, K. Nissim, and G. Kollios, “GRECS: Graph encryption for approximate shortest distance queries,” in *Proc. ACM CCS*, New York, NY, USA, 2015, pp. 504–517.
2. S. Wang, X. Xiao, Y. Yang, and W. Lin, “Effective indexing for approximate constrained shortest path queries on large road networks,” *Proc. VLDB Endowment*, vol. 10, no. 2, pp. 61–72, 2016.
3. M. Shen, M. Wei, L. Zhu, and M. Wang, “Classification of encrypted traffic with second-order Markov chains and application attribute bigrams,” *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 8, pp. 1830–1843, Aug. 2017.
4. M. Shen, K. Xu, K. Yang, and H.-H. Chen, “Towards efficient virtual network embedding across multiple network domains,” in *Proc. IEEE 22nd Int. Symp. Quality Service*, May 2014, pp. 61–70.
5. K. Xu, M. Shen, H. Liu, J. Liu, F. Li, and T. Li, “Achieving optimal traffic engineering using a generalized routing framework,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 27, no. 1, pp. 51–65, Jan. 2016.
6. Y. Low, J. Gonzalez, A. Kyrola, D. Bickson, E. C. Guestrin, and J.

M. Hellerstein, “GraphLab: A new framework for parallel machine learning,” *Proc. 26th Conf. Uncertainty Artif. Intell. (UAI)*, Jun. 2010, pp. 340–349.

1. G. Malewicz *et al.*, “Pregel: A system for large-scale graph processing,” in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2010, pp. 135–146, 2010.
2. W.-S. Han *et al.*, “TurboGraph: A fast parallel graph engine handling billion-scale graphs in a single PC,” in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2013, pp. 77–85.
3. P. Hansen, “Bicriterion path problems,” in *Multiple Criteria Decision Making Theory and Application*. Berlin, Germany: Springer, 1980, pp. 109–127.
4. R. Hassin, “Approximation schemes for the restricted shortest path problem,” *Math. Oper. Res.*, vol. 17, no. 1, pp. 36–42, 1992.
5. G. Tsaggouris and C. Zaroliagis, “Multiobjective optimization: Improved FPTAS for shortest paths and non-linear objectives with applications,” *Theory Comput. Syst.*, vol. 45, no. 1, pp. 162–186, 2009.
6. S. Storandt, “Route planning for bicycles-exact constrained shortest paths made practical via contraction hierarchy,” in *Proc. ICAPS*, vol. 4. 2012, p. 46.
7. A. Sealfon, “Shortest paths and distances with differential privacy,” in

*Proc. ACM SIGMOD*, New York, NY, USA, 2016, pp. 29–41.

1. K. Lewi and D. J. Wu, “Order-revealing encryption: New constructions, applications, and lower bounds,” in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS)*, New York, NY, USA, 2016, pp. 1167–1178.
2. N. Chenette, K. Lewi, S. A. Weis, and D. J. Wu, “Practical order- revealing encryption with limited leakage,” in *Proc. IACR-FSE*, 2016, pp. 474–493.
3. D. Boneh, E.-J. Goh, and K. Nissim, “Evaluating 2-DNF formulas on ciphertexts,” in *Proc. TCC*, 2005, pp. 325–341.
4. M. Chase and S. Kamara, “Structured encryption and controlled disclo- sure,” in *Proc. ASIACRYPT*, 2010, pp. 577–594.
5. C. Chen *et al.*, “An efficient privacy-preserving ranked keyword search method,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 27, no. 4, pp. 951–963, Apr. 2016.
6. X. Huang and X. Du, “Achieving big data privacy via hybrid cloud,” in

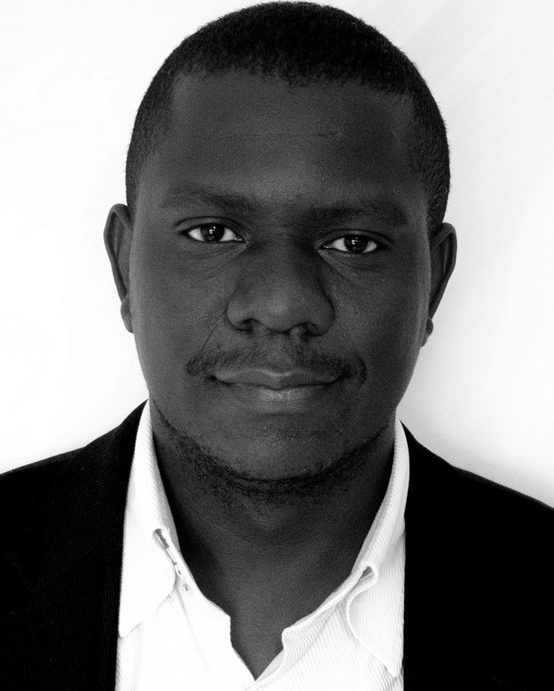
*Proc. Int. Conf. INFOCOM*, Apr. 2014, pp. 512–517.

1. Y. Cheng, X. Fu, X. Du, B. Luo, and M. Guizani, “A lightweight live memory forensic approach based on hardware virtualization,” *Inf. Sci.*, vol. 379, pp. 23–41, Feb. 2017.
2. L. Wu, X. Du, and J. Wu, “MobiFish: A lightweight anti-phishing scheme for mobile phones,” in *Proc. Int. Conf. Comput. Commun. Netw.*, 2014, pp. 1–8.
3. L. Wu, X. Du, and X. Fu, “Security threats to mobile multimedia applications: Camera-based attacks on mobile phones,” *IEEE Commun. Mag.*, vol. 52, no. 3, pp. 80–87, Mar. 2014.
4. X. Du, Y. Xiao, M. Guizani, and H.-H. Chen, “An effective key management scheme for heterogeneous sensor networks,” *Ad Hoc Netw.*, vol. 5, no. 1, pp. 24–34, 2007.
5. N. Cao, Z. Yang, C. Wang, K. Ren, and W. Lou, “Privacy-preserving query over encrypted graph-structured data in cloud computing,” in *Proc. 31st Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jun. 2011, pp. 393–402.
6. S. P. Kasiviswanathan, K. Nissim, S. Raskhodnikova, and A. Smith, “Analyzing graphs with node differential privacy,” in *Theory of Cryp- tography*, A. Sahai, Ed. Berlin, Germany: Springer, 2013, pp. 457–476, doi: [10.1007/978-3-642-36594-2\_26.](http://dx.doi.org/10.1007/978-3-642-36594-2_26)
7. E. Shen and T. Yu, “Mining frequent graph patterns with differential privacy,” in *Proc. SIGKDD*, New York, NY, USA, 2013, pp. 545–553.
8. M. Blanton, A. Steele, and M. Alisagari, “Data-oblivious graph algo- rithms for secure computation and outsourcing,” in *Proc. 8th ACM SIGSAC Symp. Inf., Comput. Commun. Secur. (ASIA CCS)*, New York, NY, USA, 2013, pp. 207–218.
9. A. Aly, E. Cuvelier, S. Mawet, O. Pereira, and M. V. Vyve, *Securely Solving Simple Combinatorial Graph Problems*. Berlin, Germany: Springer, 2013.
10. M. Keller and P. Scholl, *Efficient, Oblivious Data Structures for MPC*. Berlin, Germany: Springer, 2014.
11. D. Gupta *et al.*, “A new approach to interdomain routing based on secure multi-party computation,” in *Proc. ACM Workshop Hot Topics Netw.*, 2012, pp. 37–42.
12. F. Bayatbabolghani, M. Blanton, M. Aliasgari, and M. Goodrich. (Feb. 2017). “Secure fingerprint alignment and matching protocols.” [Online]. Available: https://arxiv.org/abs/1702.03379
13. E. Cohen, E. Halperin, H. Kaplan, and U. Zwick, “Reachability and distance queries via 2-hop labels,” *SIAM J. Comput.*, vol. 32, no. 5, pp. 937–946, 2002.
14. T. Akiba, Y. Iwata, and Y. Yoshida, “Fast exact shortest-path distance queries on large networks by pruned landmark labeling,” in *Proc. SIGMOD*, 2013, pp. 349–360.
15. D. X. Song, D. Wagner, and A. Perrig, “Practical techniques for searches on encrypted data,” in *Proc. IEEE Symp. Secur. Privacy*, May 2000, pp. 44–55.
16. R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, “Searchable symmetric encryption: Improved definitions and efficient constructions,” in *Proc. ACM CCS*, New York, NY, USA, 2006, pp. 79–88.
17. S. Kamara, C. Papamanthou, and T. Roeder, “Dynamic searchable symmetric encryption,” in *Proc. ACM Conf. Comput. Commun. Secur.*, 2012, pp. 965–976.
18. D. Cash *et al.*, “Dynamic searchable encryption in very-large databases: Data structures and implementation,” in *Proc. NDSS*, 2014, pp. 1–32.
19. E. Stefanov, C. Papamanthou, and E. Shi, “Practical dynamic searchable encryption with small leakage,” in *Proc. NDSS*, 2014, pp. 23–26.
20. R. Curtmola, J. A. Garay, S. Kamara, and R. Ostrovsky, “Searchable symmetric encryption: Improved definitions and efficient constructions,” *J. Comput. Secur.*, vol. 19, no. 5, pp. 895–934, 2011.
21. M. Naveed, S. Kamara, and V. Charles Wright, “Inference attacks on property-preserving encrypted databases,” in *Proc. 22nd ACM SIGSAC Conf. Comput. Commun. Secur. (CCS)*, New York, NY, USA, 2015, pp. 644–655.
22. A. Ben-Efraim, Y. Lindell, and E. Omri, “Optimizing semi-honest secure multiparty computation for the Internet,” in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2016, pp. 578–590.
23. A. Ben-David, N. Nisan, and B. Pinkas, “FairplayMP: A system for secure multi-party computation,” in *Proc. ACM CCS*, Alexandria, VA, USA, Oct. 2008, pp. 257–266.

**Meng Shen** (M’14) received the B.Eng. degree in computer science from Shandong University, Jinan, China, in 2009, and the Ph.D. degree in computer science from Tsinghua University, Beijing, China, in 2014. He is currently an Assistant Professor with the Beijing Institute of Technology, Beijing, China. His research interests include privacy protection of cloud-based services, network virtualization, and traffic engineering. He was a recipient of the Best Paper Runner-Up Award at IEEE IPCCC 2014.

**Baoli Ma** received the B.Eng. degree in computer science from Beijing Institute of Technology, Bei- jing, China, in 2015, where he is currently pursuing the master’s degree with the School of Computer Science. His research interest is secure searchable encryption.

**Liehuang Zhu** is a Professor with the School of Computer Science, Beijing Institute of Technology. He is selected into the Program for New Century Excellent Talents in University from the Ministry of Education, China. His research interests include Internet of Things, cloud computing security, and internet and mobile security.

**Rashid Mijumbi** (M’17) received the Ph.D. degree in telecommunications engineering from the Uni- versitat Politecnica de Catalunya (UPC), Barcelona, Spain. He was a Post-Doctoral Researcher with UPC and the Telecommunications Software and Systems Group, Waterford, Ireland, where he participated in several Spanish national, European, and Irish National Research Projects. He is currently a Soft- ware Systems Reliability Engineer with Bell Labs CTO, Nokia, Dublin, Ireland. His current research focus is on various aspects of 5G, Network Func-

tions Virtualization, and Software Defined Networking systems. He was a recipient of the 2016 IEEE Transactions Outstanding Reviewer Award recog- nizing outstanding contributions to the IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT.

**Xiaojiang Du** (SM’09) received the B.S. and M.S. degrees from Tsinghua University, Beijing, China, in 1996 and 1998, respectively, and the M.S. and Ph.D. degrees from the University of Maryland at College Park, in 2002 and 2003, respectively, all in electrical engineering. He is currently a tenured Professor with the Department of Com- puter and Information Sciences, Temple University, Philadelphia, USA. His research interests are wire- less communications, wireless networks, security, and systems. He has authored over 200 journal and

conference papers in these areas, as well as a book published by Springer. He is a Life Member of the ACM. He received over $5 million U.S. dollars research grants from the U.S. National Science Foundation, Army Research Office, Air Force, NASA, the State of Pennsylvania, and Amazon. He was a recipient of the Best Paper Award at IEEE GLOBECOM 2014 and the Best Poster Runner-Up Award at ACM MobiHoc 2014. He serves on the editorial boards of three international journals.

**Jiankun Hu** is a Professor with the School of Engineering and IT, University of New South Wales, Canberra, Australia. He is an invited expert of Aus- tralia Attorney-Generals Office assisting the draft of Australia National Identity Management Policy. He has served at the Panel on Mathematics, Information and Computing Sciences, Australian Research Council ERA (The Excellence in Research for Australia) Evaluation Committee 2012. His research interest is in the field of cyber security covering intrusion detection, sensor key manage-

ment, and biometrics authentication. He has many publications in top venues including IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, *Pattern Recognition*, and IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS.

He is an Associate Editor of IEEE TRANSACTIONS ON INFORMATION

FORENSICS AND SECURITY.