Computing Node Clustering Coefficients Securely

Katchaguy Areekijseree, Yuzhe Tang and Sucheta Soundarajan
Department of Electrical Engineering and Computer Science, Syracuse University, NY USA.
{kareekij, ytang100, susounda}@syr.edu

Introduction

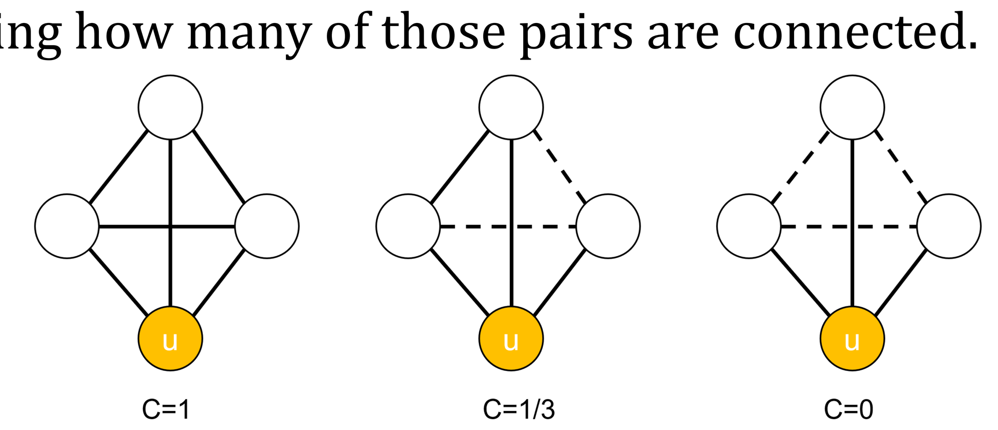
- Complex networks (e.g. social networks) gain a lot of interest from the researchers.
- To gain more insights, we can perform an analysis on this network.
- Some information **may be leaked** or **shared with others** when perform any analysis task.
  - E.g. number of friends and who they are.
- However, some users are not willing to share their information.
- Secure multi-party computation (MPC) [1] allows individuals to jointly perform any computation without reveal individual's input (convert any computation into a circuit of AND gates).
- **Long-term goal**: To develop a 'library' for performing graph operations securely.

Computing Clustering Coefficient

Suppose we want to compute node *u*'s clustering coefficient

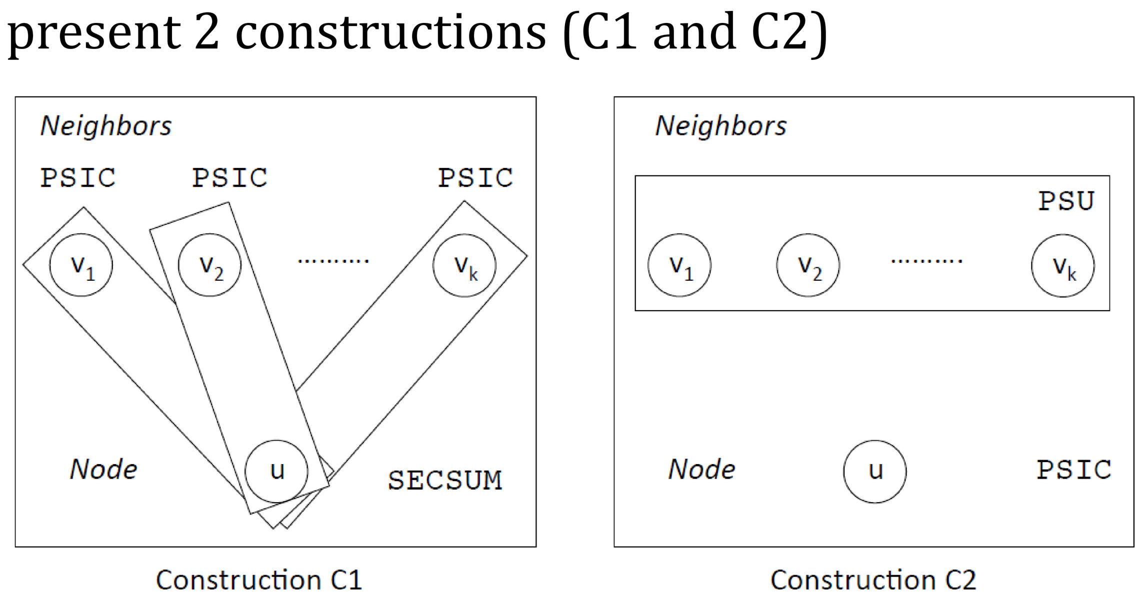
A. Non-secure setting

- Iterating over all pairs of node *u*'s neighbors.
- Counting how many of those pairs are connected.



B. Secure setting (*u* does not know its neighbors' connections):

- We present 3 secured operations
  1) Private Set Intersection Cardinality (PSIC) – compute size of the sets intersection.
  2) Private Set Union (PSU) – union of multiple sets.
  3) Secure Sum (SECSUM) – sum of numbers
- We present 2 constructions (C1 and C2)



Algorithm 1 C1: Secure Clustering Coefficient Computation

function C1(u) ▷ u is a node with degree d_u
  for v in $nbs(u)$ **do**
    s_i = PSIC(u, v)
  end for
  sum = SECSUM$(u, s_{i:i\in[1,2,..,d_u]}, nbs(u))$
  $cc = sum/(d_u \cdot (d_u - 1))$
  return cc
end function

- There are *trade-off* between **security** and **efficiency**.

Cost of MPC

Cost of MPC can be evaluated by the number of AND gates.

- C[PSIC] = $(M_1 + M_2)log(M_1 + M_2)(n^2 - n)/2$
- C[PSU] = $(\sum_{i=1}^{n} M_i)\log(\sum_{i=1}^{n} M_i)\,(n^2 - n)\,/2$
- C[SECSUM] = $(n^3 - n^2)/2$

where n is the number of parties and M_i is the set cardinality.

Experimental Setup

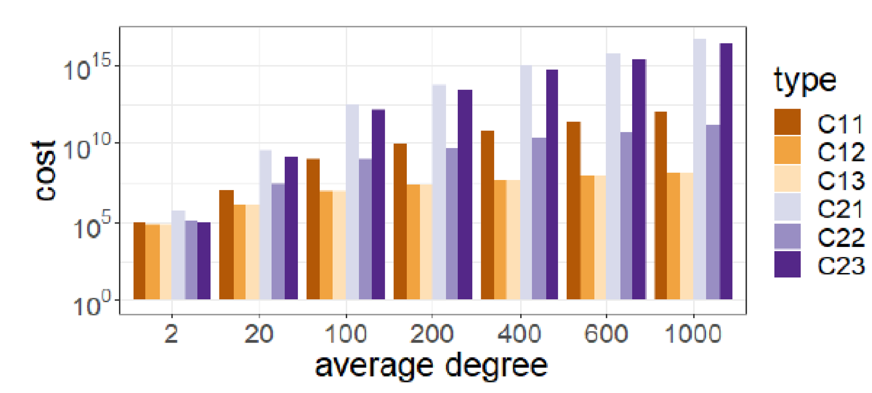
We consider 3 network properties that could computational cost

- Average degree
- Degree distribution
- Clustering coefficient

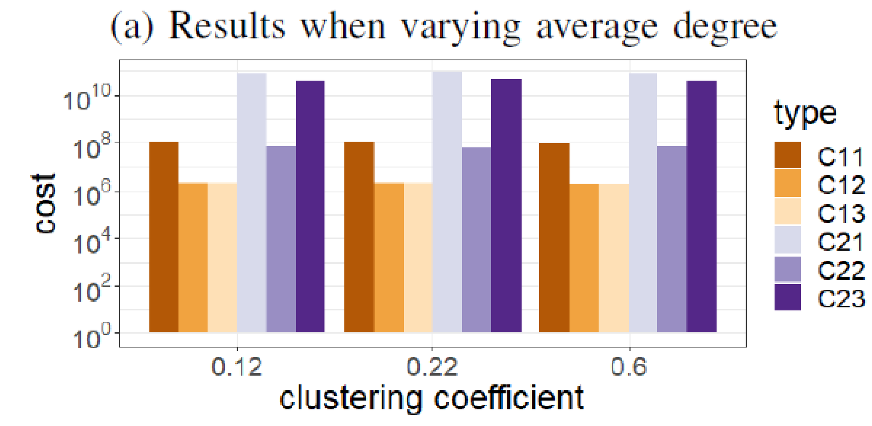
We generate 2000-node networks

- *ER* (binomial) and *LFR* models[2] (power-law).

Results

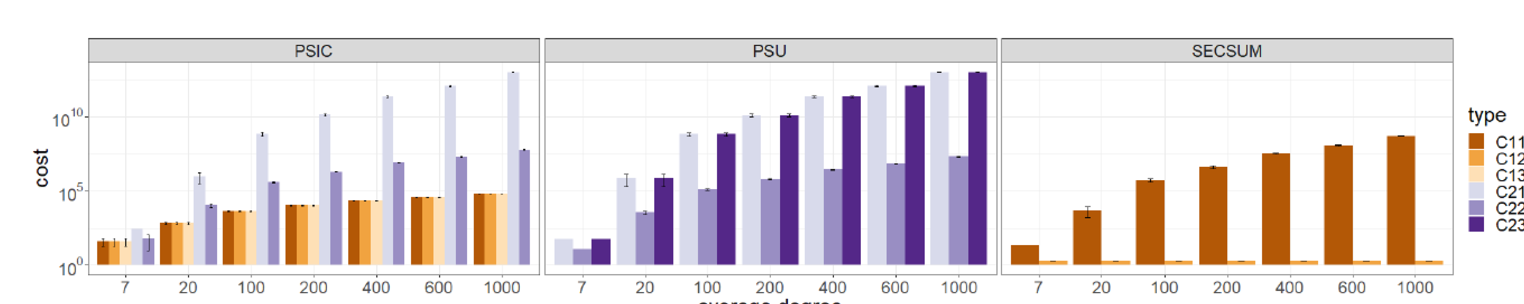


(a) Results when varying average degree



(b) Results when varying clustering coefficient

Figure 1: Average total computation cost of each node on networks with different average degrees and clustering coefficients. Costs increase with degree.



(a) Random networks

Figure 2: Average computation cost of each primitive operation (PSIC, PSU and SECSUM) on Erdos-Renyi networks with varying average degrees.

Key Observations

1. The computation cost increases as average degree increases, it highly depends on number of parties (u's neighbors).
2. Clustering coefficient has little effect on the computation cost.
3. Computation costs do not seem to be dependent on degree distribution.

Conclusion

- We demonstrate how to design secure MPC algorithm.
- We proposed two high-level constructions (C1 and C2)
- Our *long-term* goal is to develop a library of secure graph primitives operations.
- This primitives operations will be building blocks for more sophisticated techniques.
  - e.g. community detection or link prediction.

References

1. O. Goldreich, S. Micali, and A. Wigderson, "How to play any mental game or A completeness theorem for protocols with honest majority," in Proceedings of the 19th Annual ACM Symposium on Theory of Computing, 1987, pp. 218–229.
2. A. Lancichinetti, S. Fortunato, and F. Radicchi, "Benchmark graphs for testing community detection algorithms," Physical review E, vol. 78, no. 4, p. 046110, 2008.