Covering maximal cliques in real-world graphs with dense subgraphs

Sabyasachi Basu* Shweta Jain† Haim Kaplan‡ Jakub Łącki§ Blair D. Sullivan†

Abstract

Listing dense subgraphs of a given graph is a widely studied problem with practical applications in social network analysis, bioinformatics, topological graph analysis, fraud detection and others. A fundamental challenge is the fact that the number of dense subgraphs – and even the number of maximal cliques – of a graph can be exponential in the number of vertices. In order to address this difficulty we reformulate the problem as follows: can a small set of dense subgraphs capture the maximal cliques of a graph? We call a collection $\mathcal{C} = \{C_i\}$ of subgraphs a ρ-dense *clique container* of the graph if (a) every maximal clique is contained in some C_i and (b) the edge density of each C_i is $\geq \rho$. We prove that for a constant ρ, given a graph with n vertices and polylogarithmic degeneracy, there exists a ρ-dense clique container of size $n^{1+o(1)}$. Further, we consider the class of c-closed graphs, which model the real-world phenomenon of triadic closure, and show they also have ρ-dense clique container whose size is exponentially smaller than the number of maximal cliques (which itself is polynomial in n but exponential in c). We complement these results with simple algorithms that construct such clique containers and experimentally demonstrate that their sizes are surprisingly small in the case of real-world graphs.

1 Introduction

Finding dense subgraphs is an important problem with many applications in social network analysis, bioinformatics, fraud detection, brain network analysis, transportation and logistics, cybersecurity among others. It has been used for tasks such as community detection, identifying functional modules or protein complexes in protein-protein interaction networks, finding abnormal dense subgraphs in brain connectivity networks that may be biomarkers for neurological disorders, finding dense clusters of malicious IP addresses or devices and many others. There have been many different notions of "dense" that have been used in the literature but the core idea is the same: to find sets of vertices that have many edges among them. We point the reader to several recent surveys ([13, 12, 8, 1, 14]) that explore various versions of the problem and their respective applications in detail. In this paper, we restrict our attention to dense subgraphs defined using *edge density*.

The large body of prior work on mining dense subgraphs employs two main approaches. The first treats it as a clustering problem, leveraging various algorithms that perform surprisingly well in practice, such as Louvain. However, this approach inherently partitions the vertex set – excluding solutions with overlapping dense subgraphs. As a result, we may miss important dense regions of the graph. While some extensions address this limitation, they typically fail to provide guarantees on the quality of the extracted subgraphs. The second approach aims to capture these overlaps by enumerating *all* dense subgraphs. However, this method inherently has exponential runtime when the number of such subgraphs is large, making it prohibitively slow for large datasets [15, 16].

To address this challenge, we pose a slightly different question: Can we identify a small set of dense subgraphs such that every clique is fully contained within at least one of the subgraphs? Our goal is then to fix a minimum required subgraph density ρ and construct a minimum-size clique container of a given graph. We observe that in the case of general graphs, there exist graphs for which small (polynomial in the size of the input) ρ-dense clique containers with constant ρ do not exist. In this work, we consider this problem in two classes of graphs that model real-world networks: bounded degeneracy graphs and c-closed graphs [9, 3].

2 Finding small clique containers

Let $G = (V, E)$ be an undirected graph with $n = |V|$ vertices and $m = |E|$ edges. For $C \subseteq V$, we denote by $G[C]$ the subgraph of G induced on C. We define the *density* of a graph G to be $m/\binom{n}{2}$, or 1, if the graph has only 1 vertex. We denote the degeneracy of G by α, and let $0 < \rho \leq 1$ represent a density threshold.

Definition 1. *Let $G = (V, E)$ be an undirected graph. A ρ-dense clique container of G is a collection of clusters $C_1, \dots, C_k$, such that: (a) for any clique H in G, there exists a cluster C_i, such that $H \subseteq C_i$, and (b) the density of any cluster C_i is at least ρ.*

*University of California, Santa Cruz, sbasu3@ucsc.edu
†University of Utah, {shweta.jain, blair.sullivan}@utah.edu
‡Tel Aviv University, Google Research, haimk@tauex.tau.ac.il
§Google Research, jlacki@google.com

Dataset	$\|V\|$	$\|E\|$	$\#T$	$\#\ MC$	$\frac{\#T}{\#MC}$	α	CC	
							$\rho = 0.1$	$\rho = 0.5$
Wiki-Vote	7k	104k	608k	459k	1.35	9	0.008	0.051
Enron	37k	184k	727k	227k	3.42	43	0.084	0.106
Epinions	76k	509k	1624k	1775k	0.97	67	0.020	0.066
Wiki-Talk	2494k	5021k	9204k	86333k	0.18	131	0.007	0.011

Table 1: Container sizes for four real-world networks, along with descriptive statistics including n, m, the number of triangles $\#T$, number of maximal cliques $\#MC$ and degeneracy α. At right, we give the size of the output ρ-dense clique containers (CC), measured relative to the number of maximal cliques.

We first give a clique container for bounded degeneracy graphs. Our algorithm, CliqueContainer, is an adaptation of Chiba-Nishizeki [7] that uses backtracking with degeneracy ordering to recursively enumerate clusters in certain subgraphs of the graph. The main idea is as follows. Consider a recursive call for a subgraph H. If the density of H is at least ρ, we can simply return a container of size 1 containing $V(H)$. Otherwise, there exists a vertex v of degree at most $\rho \cdot |V(H)|$. In this case, we recursively find (a) ρ-dense clusters covering v, by recursing on the neighborhood of v, and (b) ρ-dense clusters not covering v by recursing on $H \setminus v$. Since one of the two recursive problems is smaller by a factor of ρ, we can derive an improved bound. We note that our algorithm draws inspiration from the graph containers of Kleitman and Winston [11] studied in the context of independent sets, which were recently generalized to hypergraphs [2] and used algorithmically in several lines of work [4, 5, 17, 10].

Lemma 2.1. *Algorithm* CliqueContainer *outputs a ρ-dense clique container of size $O(n \cdot \alpha^{(1+\log_{1/\rho} \alpha)/2})$ in time $O(n \cdot \alpha^{(5+\log_{1/\rho} \alpha)/2})$ for $\rho \in [1/n, 1)$.*

Whenever $\rho \leq 1/n$, there exists a trivial ρ-dense clique container of size $O(n)$ consisting of the connected components of G. We also note that since $\alpha \leq n-1$, Lemma 2.1 implies that a ρ-dense clique container of size quasi-polynomial in n always exists. This is remarkable because the number of maximal cliques can be exponential ($O(3^{n/3})$) [6] in the worst case.

Corollary 2.1. *For $\alpha = 2^{o(\sqrt{\log n})}$ and $\rho = 1 - \Omega(1)$ algorithm* CliqueContainer *outputs a ρ-dense clique container of size $O(n^{1+o(1)})$ in time $O(n^{1+o(1)})$.*

Next, we consider *c-closed* graphs: a class of graphs inspired by the triadic closure property of real-world graphs [9, 3]. A graph is said to be c-closed if every pair of vertices that share c or more common neighbors are themselves connected, and real-world graphs tend to have low c-values ($c \ll \sqrt{n}$) [9, 3]. Significantly, the enumeration of maximal cliques in c-closed graphs[1] [9, 3] can be done in time $O(n^2 3^{(c-1)/3)})$ [9, 3]. We show the following:

Lemma 2.2. *Every c-closed graph G has a ρ-dense clique container of size $O(n^2 3^{(c-\lfloor 1/\rho \rfloor)/3})$ which can be found in time $O(p(n,c) + n^2 3^{(c-\lfloor 1/\rho \rfloor)/3})$ where $p(n,c)$ is the time required to enumerate the wedges of G.*

Essentially, we show that a key step in the algorithm of Fox et. al [9] that takes time $O(3^{(c-1)/3})$ can be replaced with another that takes time $3^{(c-\lfloor 1/\rho \rfloor)/3}$.

Corollary 2.2. *For $\rho \leq 1/c$, a c-closed graph has a ρ-dense clique container of size $O(n^2)$ that can be found in time $O(p(n,c) + n^2)$.*

Since real-world graphs typically have low α and c, this means we can cover their maximal cliques using a polynomial number of relatively dense subgraphs.

3 Experimental results

We implemented algorithm CliqueContainer in C++ and evaluated it on four real-world graphs on a commodity machine. Table 1 shows the information about these graphs and the sizes of the ρ-dense clique containers obtained for $\rho = 0.1$ and $\rho = 0.5$. We note that the number of maximal cliques is often close to the number of triangles in the graph; in our datasets, there are typically several more triangles. In contrast, the size of the clique container drops drastically even at 0.5, and typically drops a further order of magnitude at 0.1. We color code these ratios to aid the reader (a darker shade of green is a lower ratio, and thus better).

4 Discussion and future work

We introduced *ρ-dense clique containers* as a useful proxy for the set of dense subgraphs and showed that several graph classes admit small clique containers which we can find efficiently. We leave open whether the achieved sizes are tight and whether similarly small containers exist in broader classes of real-world graphs.

[1] The problem is *fixed parameter tractable* with respect to c

References

[1] Alessia Amelio and Clara Pizzuti. Overlapping community discovery methods: a survey. In *Social networks: Analysis and case studies*, pages 105–125. Springer, 2014.

[2] József Balogh, Robert Morris, and Wojciech Samotij. The method of hypergraph containers. In *Proceedings of the International Congress of Mathematicians: Rio de Janeiro 2018*, pages 3059–3092. World Scientific, 2018.

[3] Balaram Behera, Edin Husic, Shweta Jain, Tim Roughgarden, and C Seshadhri. Fpt algorithms for finding near-cliques in c-closed graphs. *Innovations in Theoretical Computer Science (ITCS)*, 2022.

[4] Eric Blais and Cameron Seth. Testing graph properties with the container method. In *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1787–1795, 2023.

[5] Eric Blais and Cameron Seth. New graph and hypergraph container lemmas with applications in property testing. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, STOC 2024, page 1793–1804, 2024.

[6] Coen Bron and Joep Kerbosch. Algorithm 457: finding all cliques of an undirected graph. *Communications of the ACM*, 16(9):575–577, 1973.

[7] Norishige Chiba and Takao Nishizeki. Arboricity and subgraph listing algorithms. *SIAM Journal on computing*, 14(1):210–223, 1985.

[8] Yixiang Fang, Xin Huang, Lu Qin, Ying Zhang, Wenjie Zhang, Reynold Cheng, and Xuemin Lin. A survey of community search over big graphs. *The VLDB Journal*, 29(1):353–392, 2020.

[9] Jacob Fox, Tim Roughgarden, C Seshadhri, Fan Wei, and Nicole Wein. Finding cliques in social networks: A new distribution-free model. *SIAM journal on computing*, 49(2):448–464, 2020.

[10] Matthew Jenssen, Will Perkins, and Aditya Potukuchi. Approximately counting independent sets in bipartite graphs via graph containers. *Random Structures & Algorithms*, 63(1):215–241, 2023.

[11] Daniel J. Kleitman and Kenneth J. Winston. On the number of graphs without 4-cycles. *Discrete Mathematics*, 41(2):167–172, 1982.

[12] Tommaso Lanciano, Atsushi Miyauchi, Adriano Fazzone, and Francesco Bonchi. A survey on the densest subgraph problem and its variants. *ACM Computing Surveys*, 56(8):1–40, 2024.

[13] Victor E. Lee, Ning Ruan, Ruoming Jin, and Charu Aggarwal. *A Survey of Algorithms for Dense Subgraph Discovery*, pages 303–336. Springer US, 2010.

[14] Victor E Lee, Ning Ruan, Ruoming Jin, and Charu Aggarwal. A survey of algorithms for dense subgraph discovery. *Managing and mining graph data*, pages 303–336, 2010.

[15] Ahsanur Rahman, Kalyan Roy, Ramiza Maliha, and Townim Faisal Chowdhury. A fast exact algorithm to enumerate maximal pseudo-cliques in large sparse graphs. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2479–2490, 2024.

[16] Takeaki Uno. An efficient algorithm for solving pseudo clique enumeration problem. *Algorithmica*, 56:3–16, 2010.

[17] Or Zamir. Algorithmic applications of hypergraph and partition containers. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, STOC 2023, page 985–998, 2023.