

# An Efficient Algorithm for All-Pairs Bounded Edge Connectivity

Friday, July 14, 2023 12:10 PM (20 minutes)

Shyan Akmal and Ce Jin

**Abstract:** Our work concerns algorithms for a variant of  $\text{Maximum Flow}$  in unweighted graphs. In the All-Pairs Connectivity (APC) problem, we are given a graph  $G$  on  $n$  vertices and  $m$  edges, and are tasked with computing the size of a minimum  $st$ -cut in  $G$  for all pairs of vertices  $(s, t)$ . Significant algorithmic breakthroughs have recently shown that over undirected graphs, APC can be solved in  $n^{2+o(1)}$  time, which is essentially optimal. In contrast, the true time complexity of APC over directed graphs remains open: this problem can be solved in  $O(m^\omega)$  time, where  $\omega \in (2, 2.373)$  is the exponent of matrix multiplication, but no matching conditional lower bound is known.

Following [Abboud et al., ICALP 2019], we study a bounded version of APC called the  $k$ -Bounded All Pairs Connectivity ( $k$ -APC) problem. In this variant of APC, we are given an integer  $k$  in addition to the graph  $G$ , and are now tasked with reporting the size of a minimum  $st$ -cut only for pairs  $(s, t)$  of vertices with min-cut value less than  $k$  (if the min  $st$ -cut has size at least  $k$ , we just can report it is “large” instead of computing the exact value).

Our main result is an  $O((kn)^\omega)$  time algorithm for  $k$ -APC in directed graphs.

This is the first algorithm which solves  $k$ -APC faster than the more general APC for all  $k \geq 3$ . This runtime is essentially  $O(n^\omega)$  for all  $k \leq \text{poly}(\log n)$ , which essentially matches the optimal runtime for the  $k=1$  case of  $k$ -APC, under popular conjectures from fine-grained complexity. Previously, this runtime was only known for  $k \leq 2$  in general [Georgiadis et al., ICALP 2017], and for  $k \leq o((\log n)^{1/2})$  in the special case of directed acyclic graphs [Abboud et al., ICALP 2019]. Our result employs the same algebraic framework used in previous work, introduced by [Cheung, Lau, and Leung, FOCS 2011]. A direct implementation of this framework involves inverting a large random matrix. Our faster algorithm for  $k$ -APC is based off the insight that it suffices to invert a low-rank random matrix instead of a generic random matrix, which yields a speed-up.

We also obtain a new algorithm for a variant of  $k$ -APC, the  $k$ -Bounded All-Pairs Vertex Connectivity ( $k$ -APVC) problem, where we now report the sizes of minimum vertex-cuts instead of edge-cuts. We show how to solve  $k$ -APVC in  $O(k^2 n^\omega)$  time. Previous work gave an  $O((kn)^\omega)$  algorithm for this problem [Abboud et al., ICALP 2019], so our algorithm is faster if  $\omega > 2$ .

**Presenter:** AKMAL, Shyan

**Session Classification:** Track A-1