Pairwise Reachability Oracles and Preservers Under Failures

Diptarka Chakraborty
National University of Singapore, Singapore

Kushagra Chatterjee
National University of Singapore, Singapore

Keerti Choudhary
Indian Institute of Technology Delhi, India

Abstract

In this paper, we consider reachability oracles and reachability preservers for directed graphs/networks prone to edge/node failures. Let $G = (V, E)$ be a directed graph on n-nodes, and $\mathcal{P} \subseteq V \times V$ be a set of vertex pairs in G. We present the first non-trivial constructions of single and dual fault-tolerant pairwise reachability oracle with constant query time. Furthermore, we provide extremal bounds for sparse fault-tolerant reachability preservers, resilient to two or more failures. Prior to this work, such oracles and reachability preservers were widely studied for the special scenario of single-source and all-pairs settings. However, for the scenario of arbitrary pairs, no prior (non-trivial) results were known for dual (or more) failures, except those implied from the single-source setting. One of the main questions is whether it is possible to beat the $O(n|\mathcal{P}|)$ size bound (derived from the single-source setting) for reachability oracle and preserver for dual failures (or $O(2^k n|\mathcal{P}|)$ bound for k failures). We answer this question affirmatively. Below we summarize our contributions.

- For an n-vertex directed graph $G = (V, E)$ and $\mathcal{P} \subseteq V \times V$, we present a construction of $O(n\sqrt{|\mathcal{P}|})$ sized dual fault-tolerant pairwise reachability oracle with constant query time. We further provide a matching (up to the word size) lower bound of $\Omega(n\sqrt{|\mathcal{P}|})$ on the size (in bits) of the oracle for the dual fault setting, thereby proving that our oracle is (near-)optimal.
- Next, we provide a construction of $O(n + \min\{|\mathcal{P}|\sqrt{n},\ n\sqrt{|\mathcal{P}|}\})$ sized oracle with $O(1)$ query time, resilient to single node/edge failure. In particular, for $|\mathcal{P}|$ bounded by $O(\sqrt{n})$ this yields an oracle of just $O(n)$ size. We complement the upper bound with a lower bound of $\Omega(n^{2/3}|\mathcal{P}|^{1/2})$ (in bits), refuting the possibility of a linear-sized oracle for $\mathcal{P}$ of size $\omega(n^{2/3})$.
- We also present a construction of $O(n^{4/3}|\mathcal{P}|^{1/3})$ sized pairwise reachability preservers resilient to dual edge/vertex failures. Previously, such preservers were known to exist only under single failure and had $O(n + \min\{|\mathcal{P}|\sqrt{n},\ n\sqrt{|\mathcal{P}|}\})$ size [Chakraborty and Choudhary, ICALP'20]. We also show a lower bound of $\Omega(n\sqrt{|\mathcal{P}|})$ edges on the size of dual fault-tolerant reachability preservers, thereby providing a sharp gap between single and dual fault-tolerant reachability preservers for $|\mathcal{P}| = o(n)$.
- Finally, we provide a generic pairwise reachability preserver construction that provides a $o(2^k n|\mathcal{P}|)$ sized subgraph resilient to k failures, for any $k \geq 1$. Before this work, we only knew of an $O(2^k n|\mathcal{P}|)$ bound implied from the single-source setting [Baswana, Choudhary, and Roditty, STOC'16].

2012 ACM Subject Classification Mathematics of computing → Graph algorithms; Theory of computation → Sparsification and spanners

Keywords and phrases Fault-tolerant, Reachability Oracle, Reachability Preservers, Graph sparsification, Lower bounds

Digital Object Identifier 10.4230/LIPIcs.ICALP.2022.35

Category Track A: Algorithms, Complexity and Games

Related Version *Full Version*: `https://arxiv.org/abs/2110.11613`

Funding *Diptarka Chakraborty*: The author is supported in part by NUS ODPRT Grant, WBS No. A-0008078-00-00.
Kushagra Chatterjee: The author is supported in part by NUS ODPRT Grant, WBS No. A-0008078-00-00.
Keerti Choudhary: The author is supported in part by Google India Algorithms Research grant 2021.

Acknowledgements The authors would like to thank anonymous reviewers for many helpful comments on a preliminary version of this work, especially for pointing out [9].

1 Introduction

Networks in most real-life applications are prone to failures. These failures, though unpredictable, are transient due to some simultaneous repair process that is undertaken in the application. This motivates the research on designing fault-tolerant structures for various graph problems. In the past few years, a lot of work has been done in designing fault-tolerant structures for various graph problems like connectivity [34, 32, 4, 26, 11], finding shortest paths [20], graph-structures preserving approximate distances [30, 19, 15, 22, 5, 6, 10, 3] etc. Reachability is one of the fundamental graph properties which is as ubiquitous as graphs themselves. In this paper, we study pairwise reachability structures under edge/node failures. In particular, given any set $\mathcal{P}$ of node-pairs, we provide design of graph sparsification structures, and sensitivity oracles for the reachability problem. We present our results in terms of edge failures. However, all our upper bound results also hold for node failures.[1]

1.1 Sensitivity Oracle

In the Sensitivity oracle, the goal is to design a data structure for a network prone to edge/vertex failures to efficiently answer queries pertaining to the graph structure (e.g., connectivity, reachability, distance, etc.). We first formally define the notion of Fault-Tolerant Reachability Oracle (FTRO).

▶ **Definition 1** (FTRO). *Let $\mathcal{P} \in V \times V$ be any set of pairs of vertices. For a graph G, a data structure $DS(G)$ is said to be a k-*Fault-Tolerant Reachability Oracle *of G for $\mathcal{P}$, denoted as k-FTRO$(G, \mathcal{P})$, if given a query with any pair $(s, t) \in \mathcal{P}$ and any subset $F \subseteq E$ of at most k edges, $DS(G)$ efficiently decides whether or not t is reachable from s in $G \setminus F$.*

To date, no non-trivial bounds were known for FT-pairwise reachability oracle. The only known results are for single-source setting (i.e., $\mathcal{P} = \{s\} \times V$) and all-pairs setting $\mathcal{P} = V \times V$.

For single-source setting, i.e., when $\mathcal{P} = \{s\} \times V$ for some source vertex $s \in V$, under (single and) dual failure, we have an $O(n)$ size oracle with $O(1)$ query time due to [29, 17]. As an immediate corollary, for arbitrary $\mathcal{P}$ pairs, we get an $O(n|\mathcal{P}|)$-sized single/dual failure pairwise FTRO with constant query time. The bound is extremely bad for a large-sized set $\mathcal{P}$. By storing a subgraph that preserves pairwise reachability (to be discussed in detail in Section 1.2) under single failure due to [11], we get an 1-FTRO of size $O(n + \min\{\sqrt{n}|\mathcal{P}|, n\sqrt{|\mathcal{P}|}\})$ but with $O(n)$ query time. The $O(n)$ query time is due to the fresh reachability computation over

[1] In the input graph, each vertex v can be replaced by an edge (v_{in}, v_{out}), where all the incoming and outgoing edges of v are directed into v_{in} and directed out of v_{out} respectively. Thus the failure of vertex v is equivalent to failure of edge (v_{in}, v_{out}).

the stored subgraph on each query, which is entirely undesirable in terms of the efficiency of a data structure. For the special setting of all-pairs, i.e., $\mathcal{P} = V \times V$, Brand and Saranurak [37] provided a $O(n^2)$ sized k-FTRO that has $O(k^\omega)$ query time, where ω is the constant of matrix-multiplication.

One of the main questions is the following: Does there exist a pairwise reachability oracle of size $o(n|\mathcal{P}|)$ and query time $o(n)$ even for a single failure? In this paper, we answer this question affirmatively. We provide an efficient construction of a $O(n\sqrt{|\mathcal{P}|})$ sized FTRO with constant query time that is resilient to dual failure (not just single failure).

▶ **Theorem 2** (Upper Bound on 2-FTRO). *A directed graph $G = (V, E)$ with n vertices can be processed in randomized polynomial time for a given set $\mathcal{P} \subseteq V \times V$ of vertex-pairs, to build a data structure of size $O(n\sqrt{|\mathcal{P}|})$, such that for any pair $(s, t) \in \mathcal{P}$ and any set F of (at most) two edge failure, it decides whether there is an s to t path in $G \setminus F$ in time $O(1)$.*

We further show that the above size bound cannot be improved further by providing a matching (up to the word size) lower bound for two failures. To date, no non-trivial (better than linear) size lower bound is known for any pairwise FTRO.

▶ **Theorem 3** (Lower Bound on 2-FTRO). *For any positive integers n, r ($r \leq n^2$), there exists an n-vertex directed graph with a vertex-pair set $\mathcal{P}$ of size r, such that any 2-FTRO$(G, \mathcal{P})$ must be of size $\Omega(n\sqrt{|\mathcal{P}|})$ (in bits).*

In case of source-wise 2-FTRO for a source set S (i.e, when $\mathcal{P} = S \times V$), our lower bound construction provides a lower bound of $\Omega(n|S|)$ (in bits). It is again a matching (up to the word size) lower bound because we know of an $O(n|S|)$-sized 2-FTRO for any source set S due to [17]. It is also worth noting that our lower bound holds irrespective of the query time and also for directed acyclic graphs.

The above lower bound does not hold for a single failure. So it is natural to ask whether we can design a smaller data structure, more specifically, $O(n)$-sized oracle that is resilient to a single failure. We provide a construction of $O(n + \min\{|\mathcal{P}|\sqrt{n}, n\sqrt{|\mathcal{P}|}\})$ sized 1-FTRO with constant $O(1)$ query time. In particular, we show that as long as the number of pairs is bounded by $O(\sqrt{n})$, we can achieve an oracle with $O(n)$ size and $O(1)$ query time. This result provides us a sharp separation in optimum size of a FTRO between single and dual failure. To the best of our knowledge, this is the first separation result between single and dual failure reachability oracle.

▶ **Theorem 4** (Upper Bound on 1-FTRO). *A directed graph $G = (V, E)$ with n vertices can be processed in polynomial time for a given set $\mathcal{P} \subseteq V \times V$ of vertex-pairs, to build a data structure of size $O(n + \min\{|\mathcal{P}|\sqrt{n}, n\sqrt{|\mathcal{P}|}\})$, such that for any pair $(s, t) \in \mathcal{P}$ and a failure edge f, it decides whether there is an s to t path in $G \setminus \{f\}$ in time $O(1)$.*

Note, the size bound of the above theorem matches the current best known bound for the pairwise reachability preserving subgraph for single failure [11].

The above upper bound gives $O(n)$ sized oracle only when the number of pairs is $O(\sqrt{n})$. Is it always possible to get a linear-sized pairwise 1-FTRO? More specifically, does any n-node graph G and a set $\mathcal{P}$ of node-pairs always possess a 1-FTRO$(G, \mathcal{P})$ of size $O(n + |\mathcal{P}|)$?[2] In this paper, we refute this possibility by showing the following.

[2] The presence of $|\mathcal{P}|$ term in the bound is justifiable by the fact that for non-failure case (i.e., the standard static setting), we can get a trivial $O(|\mathcal{P}|)$ sized oracle.

▶ **Theorem 5** (Lower Bound on 1-FTRO). *For any positive integers $n, d \geq 2$, any $p = p(n)$, there exists an n-vertex directed graph and a node-pair set $\mathcal{P}$ of size p, such that any 1-FTRO$(G, \mathcal{P})$ must be of size $\Omega\big(n^{2/(d+1)} p^{(d-1)/d}\big)$ (in bits).*

By setting $d = 2$ in the above theorem, we get a lower bound of $\Omega(n^{2/3}p^{1/2})$. This shows us that for $p = \omega(n^{2/3})$, there is an n-node graph G and a pair set $\mathcal{P}$ of size p, for which linear size 1-FTRO is not possible. Again, our lower bound holds irrespective of the query time and also for DAGs. We show the lower bound by establishing a connection between the optimal sized pairwise 1-FTRO and pairwise reachability preserving subgraph without any failure. In general, we show that the optimal size of any pairwise k-FTRO must be at least that of the reachability preserving subgraph with $k - 1$ failures. Instead of just deciding the reachability between a pair of vertex, suppose the data structure is also asked to report a path between them (if exists). Then by a standard information-theoretic argument, the optimal size of any such data structure resilient to k failures must be of size at least that of reachability preserving subgraph with k failures. Unfortunately, such a direct argument does not work for a (Boolean) data structure that only decides the reachability. Ours is the first such connection. Readers may note that there is a gap between our upper and lower bound for pairwise 1-FTRO. We leave this as an interesting open question.

1.2 Reachability Preservers

In the context of graph sparsification, *reachability preserver* (or *reachability subgraph*) for a directed graph G and a set $\mathcal{P}$ of vertex-pairs is a sparse subgraph H with as few edges as possible so that for any pair $(s, t) \in \mathcal{P}$ there is a path from s to t in H if and only if there is such a path in G. In the standard static setting (with no failure), this object has been studied widely [18, 8, 1]. We study these objects in the presence of edge/node failures.

Let us formally define fault-tolerant reachability subgraph (FTRS) for a set of node-pairs.

▶ **Definition 6** (FTRS). *Let $\mathcal{P} \in V \times V$ be any set of pairs of vertices. A subgraph H of G is said to be a k-*Fault-Tolerant Reachability Subgraph *of G for $\mathcal{P}$, denoted as k-FTRS$(G, \mathcal{P})$, if for any pair $(s, t) \in \mathcal{P}$ and for any subset $F \subseteq E$ of at most k edges, t is reachable from s in $G \setminus F$ if and only if t is reachable from s in $H \setminus F$.*

For the particular case of single-source, i.e., $\mathcal{P} = \{s\} \times V$, Baswana, Choudhary, and Roditty [4] provided a polynomial-time algorithm that, given any n-node directed graph, constructs an $O(2^k n)$-sized k-FTRS. As a corollary, to preserve reachability between arbitrary $\mathcal{P}$ pairs, we get an $O(2^k n|\mathcal{P}|)$-sized k-fault-tolerant reachability preserver. For the general setting of arbitrary pairs, the only previously known non-trivial result was for single failure [11], wherein the authors gave an upper bound of $O(n + \min(|\mathcal{P}|\sqrt{n},\ n\sqrt{|\mathcal{P}|}))$ edges. It was left open whether for dual or more failures whether keeping fewer than $O(n|\mathcal{P}|)$ edges sufficient to preserve the pairwise reachability. In particular, does any n-node graph and a set $\mathcal{P}$ of node-pairs always admit a k-FTRS of size $o(2^k n|\mathcal{P}|)$?

In this work, we answer the above question affirmatively. For dual failures, we provide an upper bound of $O(n^{4/3}|\mathcal{P}|^{1/3})$ edges on the structure of 2-FTRS$(G, \mathcal{P})$.

▶ **Theorem 7** (Upper Bound on 2-FTRS). *For any directed graph $G = (V, E)$ with n vertices and a set $\mathcal{P} \subseteq V \times V$ of vertex-pairs, there exists a 2-FTRS$(G, \mathcal{P})$ having at most $O(n^{4/3}|\mathcal{P}|^{1/3})$ edges. Furthermore, we can find such a subgraph in polynomial time.*

Clearly, for $\mathcal{P}$ of size $\omega(\sqrt{n})$, the above result breaks below the $O(n|\mathcal{P}|)$ bound. We complement our upper bound result the following lower bound.

▶ **Theorem 8** (Lower Bound on 2-FTRS). *For every n, r $(r \le n^2)$, there exists an n-vertex directed graph G and a vertex-pair set $\mathcal{P}$ of size r such that any 2-FTRS$(G, \mathcal{P})$ requires $\Omega(n\sqrt{|\mathcal{P}|})$ edges.*

Again, we show a lower bound for source-wise 2-FTRS for a source set S (i.e, when $\mathcal{P} = S \times V$), of $\Omega(n|S|)$. This matches the $O(n|S|)$ upper bound [4] of 2-FTRS for any source set S. So for the source-wise preserver, we completely resolve the question regarding the size of an optimal preserver resilient to two or more failures. For general $k > 2$ failures, we provide a lower bound of $\Omega(2^{k/2} n\sqrt{|\mathcal{P}|})$ on the size of pairwise k-FTRS.

Previously, seemingly a much weaker lower bound was known [11], where the authors could only show a lower bound of $\Omega(n|\mathcal{P}|^{1/8})$, for $\mathcal{P}$ of size n^ϵ with $\epsilon \le 2/3$. Our result provides a sharp separation between single and dual fault-tolerant reachability preservers for any $\mathcal{P}$ satisfying $\omega(1) \le |\mathcal{P}| \le o(n)$.

We also consider the question of beating $O(2^k n|\mathcal{P}|)$ bound for general k-FTRS. We show that for a certain regime of the size of $\mathcal{P}$, it is indeed possible to attain $o(2^k n|\mathcal{P}|)$ bound.

▶ **Theorem 9** (Upper Bound on k-FTRS). *For any $k \ge 1$, a directed graph $G = (V, E)$ with n vertices and a set $\mathcal{P} \subseteq V \times V$ of vertex-pairs satisfying $|\mathcal{P}| = \omega(kn^{1-\frac{1}{k}} \log n)$, there exists a k-FTRS$(G, \mathcal{P})$ having only $o(2^k n|\mathcal{P}|)$ edges.*

We summarize our results on single and dual failures in Table 1. Readers may note that there is a gap between the size of 2-FTRO and 2-FTRS in our results. We pose closing this embarrassing gap as an interesting open question.

Table 1 A comparison of size of FTRO and FTRS for single and dual failures.

Problem	Single Failure	Dual Failure
Reachability Oracle	$O(n + \min(\lvert\mathcal{P}\rvert\sqrt{n},\ n\sqrt{\lvert\mathcal{P}\rvert}))$ $\Omega(n^{2/3}\lvert\mathcal{P}\rvert^{1/2})$ (in bits) (New)	$O(n\sqrt{\lvert\mathcal{P}\rvert})$ $\Omega(n\sqrt{\lvert\mathcal{P}\rvert})$ (in bits) (New)
Reachability Preserver	$O(n + \min(\lvert\mathcal{P}\rvert\sqrt{n},\ n\sqrt{\lvert\mathcal{P}\rvert}))$ [11]	$O(n^{4/3}\lvert\mathcal{P}\rvert^{1/3})$ $\Omega(n\sqrt{\lvert\mathcal{P}\rvert})$ (New)

1.3 Related Work

A simple version of reachability preserver is when there is a single source vertex s, and we would like to preserve reachability from s to all other vertices. Baswana *et al.* [4] provided an efficient construction of a k-fault-tolerant single-source reachability preserver of size $O(2^k n)$. Further, they showed that this upper bound on the size of a preserver is tight up to some constant factor. As an immediate corollary, we get a k-FTRS of size $O(2^k n|\mathcal{P}|)$ (by applying the algorithm of [4] to find subgraph for each source vertex in pairs of $\mathcal{P}$ and then taking the union of all these subgraphs). We do not know whether this bound is tight for general k. However, for the standard static setting (with no faulty edges) much better bound is known. We know that even to preserve all the pairwise distances, not just reachability, there is a subgraph of size $O\big(n + \min(n^{2/3}|\mathcal{P}|, n\sqrt{|\mathcal{P}|})\ \big)$ [18, 8]. Later Abboud and Bodwin [1] showed that for any directed graph $G = (V, E)$ given a set S of source vertices and a pair-set $\mathcal{P} \subseteq S \times V$ we can construct a pairwise reachability preserver of size $O\big(n + \min(\sqrt{n|\mathcal{P}||S|}, (n|\mathcal{P}|)^{2/3})\ \big)$. It is further shown that for any integer $d \geq 2$ there is an infinite family of n-node graphs and vertex-pair sets $\mathcal{P}$ for which any pairwise reachability preserver must be of size $\Omega\big(n^{2/(d+1)}|\mathcal{P}|^{(d-1)/d}\big)$. Note, for undirected graphs, storing spanning forests is sufficient to preserve pairwise reachability information, and thus we can always get a linear size reachability preserver for undirected graphs. We would like to emphasize that all our results in this paper hold for directed graphs.

By [4] we immediately get an oracle of size $O(2^k n)$ for k edge (or vertex) failures with query time $O(2^k n)$. For just dual failures, we have an $O(n)$ size oracle with $O(1)$ query time due to [17].

For undirected graphs, the optimal bound of $O(kn)$ edges for k-fault-tolerant connectivity preserver directly follows from k-edge (vertex) connectivity certificate constructions provided by Nagamochi and Ibaraki [31]. For connectivity oracle, Pătraşcu and Thorup [35] presented a data structure of $O(m)$ size that can handle any k edge failures in $O(k \log^2 n \log\log n)$ time to subsequently answer connectivity queries between any two vertices in $O(\log\log n)$ time. For small values of k, Duan and Pettie [24] improved the update time of [35] to $O(k^2 \log\log n)$ by presenting a data structure of $\widetilde{O}(m)$ size. For handling vertex failures, Duan and Pettie [25] provided a data structure of $O(mk \log n)$ size with $O(k^3 \log^3 n)$ update time and $O(k)$ query time.

Other closely related problems that have been studied in the fault-tolerant model include computing distance preservers [20, 33, 32, 12], depth-first-search tree [2], spanners [15, 22], approximate distance preservers [5, 34, 7], approximate distance oracles [23, 16], compact routing schemes [16, 14].

2 Preliminaries

For any integer n, we use $[n]$ to denote the set $\{1, 2, \cdots, n\}$. Given a directed graph $G = (V, E)$ on $n = |V|$ vertices and $m = |E|$ edges, the following notations will be used throughout the paper.

- $V(H)$: The set of vertices present in a graph H.
- $E(H)$: The set of edges present in a graph H.
- $H \setminus F$: For a set of edges F, the graph obtained by deleting the edges in F from graph H.
- $s - t$ path : A directed path from a vertex s to another vertex t.
- $P \circ Q$: The concatenation of two paths P and Q, i.e., a path that first follows P and then Q.
- $P[L]$: The subpath of the path P containing the first L vertices of P.
- $P[-L]$: The subpath of the path P containing the last L vertices of P.
- $P[u - v]$: The $u - v$ subpath of the path P.

Our algorithm for computing *pairwise-reachability* preservers (and oracles) in a fault tolerant environment employs the concept of a *single-source* FTRS which is a sparse subgraph that preserves reachability from a designated source vertex even after the failure of at most k edges in G. Baswana *et al.* [4] provide a construction of sparse k-FTRS for any general $k \geq 1$ when there is a designated source vertex.

▶ **Theorem 10** ([4]). *For any directed graph $G = (V, E)$, a designated source vertex $s \in V$, and an integer $k \geq 1$, there exists a (sparse) subgraph H of G which is a k-FTRS$(G, \{s\} \times V)$ and contains at most $2^k n$ edges. Moreover, such a subgraph is computable in $O(2^k mn)$ time, where n and m are respectively the number of vertices and edges in graph G.*

Note, in the FTRO definition we restrict ourselves to a data structure with constant query time. (The term *oracle* came from its ability to answer a query in constant time.) Unlike single-source k-FTRS, there is no non-trivial construction of single-source k-FTRO for $k \geq 3$. For $k = 2$, the following result by Choudhary [17] provides an $O(n)$ size oracle.

▶ **Theorem 11** ([17]). *There is a polynomial time algorithm that given any directed graph $G = (V, E)$, a designated source vertex $s \in V$, constructs a 2-FTRO$(G, \{s\} \times V)$ of size $O(n)$.*

Our constructions will require the knowledge of the vertices reachable from a vertex s as well as the vertices that can reach s. So we will use FTRS (and FTRO) defined with respect to a source vertex ($\{s\} \times V$ case), as well as FTRS (and FTRO) defined with respect to a destination vertex ($V \times \{s\}$ case).

In this paper, we consider fault-tolerant structures with respect to edge failures only. Vertex failures can be handled by simply splitting a vertex v into an edge (v_{in}, v_{out}), where the incoming and outgoing edges of v are respectively directed into v_{in} and directed out of v_{out}.

3 Technical Overview

Construction of a 2-FTRS for a single pair

Our starting point is a simple construction of a linear (in the number of vertices) sized 2-FTRS for a single pair. Recall, we already know of such a subgraph by [4]. However, this new alternate construction will shed more light on the specific structure of a 2-FTRS, which will play a pivotal role in our constructions of pairwise 2-FTRS and 2-FTRO. Given a directed graph G and a vertex-pair (s, t), we construct a subgraph $H_{(s,t)}$ as follows: First, consider two "maximally disjoint" $s-t$ paths $P^1_{(s,t)}$ and $P^2_{(s,t)}$ (that meet only at the cut-edges and cut-vertices). We refer to these two paths as *outer strands*. Next, we add several *coupling paths* between these two outer strands, which are edge-disjoint with the outer strands. For each vertex v on the outer strands, we check for the "earliest" vertex on the strand $P^1_{(s,t)}$ (and $P^2_{(s,t)}$), from which there is a path $Q^1_{(s,t),v}$ (and $Q^2_{(s,t),v}$) to v that is edge-disjoint with both the outer strands. We refer to these path $Q^i_{(s,t),v}$ as coupling paths. Roughly speaking, two outer strands together with the coupling paths constitute the subgraph $H_{(s,t)}$ (see Figure 1). The actual construction is slightly different. Let us first briefly discuss why the above subgraph is a 2-FTRS for the pair (s, t). Then we will comment on the issue with the above simple construction and how we overcome that.

Consider any two failure edges f_1, f_2. W.l.o.g. assume, they do not form an $s - t$ cut-set; otherwise, after the failure there won't be any $s - t$ path. Thus if both f_1, f_2 lie on one of the two outer strands (i.e., either on $P^1_{(s,t)}$ or $P^2_{(s,t)}$), then since these two strands are maximally disjoint, one of them will survive after the failures. So, let f_1, f_2 lie on the strand $P^1_{(s,t)}$, $P^2_{(s,t)}$ respectively. Then consider the subpaths of $P^1_{(s,t)}$, $P^2_{(s,t)}$ above f_1, f_2, and the subpaths of $P^1_{(s,t)}$, $P^2_{(s,t)}$ below f_1, f_2. Since by assumption f_1, f_2 does not form an $s - t$ cut-set, there must be a coupling path (edge-disjoint with $P^1_{(s,t)}$, $P^2_{(s,t)}$) from one of the top subpaths to one of the bottom subpaths in G. Since $H_{(s,t)}$ consists of all the coupling paths, we get a surviving path in $H_{(s,t)} \setminus \{f_1, f_2\}$. This shows that $H_{(s,t)}$ is a 2-FTRS$(G, (s, t))$. Moreover, one may observe from the above argument that, after failure of any two edges, one of the surviving paths in $H_{(s,t)}$ must be of the following form: It first follows one of the outer strand from s to some vertex u, then takes a coupling path till some vertex v on one of

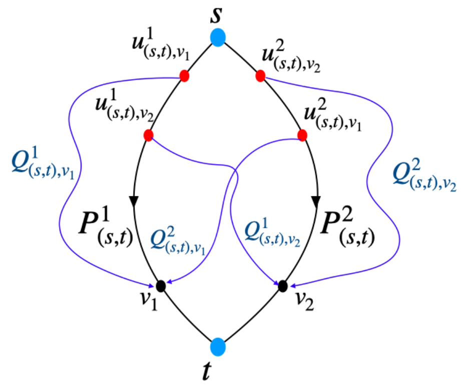

Figure 1 $H_{(s,t)}$=2-FTRS for a single pair (s,t). Two black paths are the outer strands and the purple paths are the coupling paths between them.

the outer strands, and finally follows the corresponding outer strand from v to t. We refer to such a path as *nice path*. The existence of such nice paths helps us in proving the correctness of our pairwise 2-FTRO and 2-FTRS construction in the subsequent sections.

As we mentioned earlier, our actual construction is slightly different. The main issue with the above simple construction is that the constituted subgraph could be of size $\omega(n)$ after adding all the coupling paths. To mitigate this issue, instead of adding all the coupling paths, we only add the "essential" coupling paths. (See the full version for the details.) It allows us to achieve $O(n)$ size bound without affecting the correctness of 2-FTRS. The guarantee of the existence of nice paths also remains unaffected. Of course, the correctness argument will become slightly more intricate.

Next, we use the above construction of a 2-FTRS of a single pair to study the pairwise dual fault-tolerant graph structures (reachability oracle and preserver). Our input is a directed graph $G = (V, E)$ with n nodes, and a node-pair set $\mathcal{P} \subseteq V \times V$.

Pairwise 2-FTRO: Upper bound

One of the main contributions of this work is a construction of a dual fault-tolerant pairwise reachability oracle (2-FTRO) of size $O(n\sqrt{|\mathcal{P}|})$. For simplicity, below, we briefly describe a construction that provides a slightly weaker bound, in particular, $O(n\sqrt{|\mathcal{P}|}\log n)$. Later we will comment on how to remove this extra $\log n$ factor.

We start with the 2-FTRS $H_{(s,t)}$, for each pair $(s,t) \in \mathcal{P}$. First, for all $(s,t) \in \mathcal{P}$, we consider the top and bottom $\Theta(n/\sqrt{|\mathcal{P}|})$ portion of the outer strands $(P^1_{(s,t)}, P^2_{(s,t)})$. We find a subset of vertices that intersects all these subpaths, i.e., acts as a "hitting set". Using a standard greedy algorithm we get a hitting set of size $O(\sqrt{|\mathcal{P}|}\log n)$. (Note, one may alternatively use random sampling to achieve the same bound for the hitting set with high probability.) Then we compute linear-sized single-source and single-destination 2-FTRO having query time $O(1)$, for each of the vertices in the hitting set using [17]. For each $(s,t) \in \mathcal{P}$, in a table $T_{(s,t)}$, we store one vertex from each of the top and bottom $\Theta(n/\sqrt{|\mathcal{P}|})$ length subpaths of $P^1_{(s,t)}$, $P^2_{(s,t)}$, that is also included in the hitting set. That constitutes the first part of our data structure.

Observe, for any two failure edges f_1, f_2 and a pair $(s,t) \in \mathcal{P}$, we know that there must be a surviving *nice* $s-t$ path in $H_{(s,t)} \setminus \{f_1, f_2\}$ (unless f_1, f_2 form an $s-t$ cut-set). Now, if that surviving path follows the bottom (or top) $\Theta(n/\sqrt{|\mathcal{P}|})$ length subpath of $P^j_{(s,t)}$ (for some $j \in \{1, 2\}$), it must also pass through one of the stored vertices in $T_{(s,t)}$, say v (due to the hitting set property). In this scenario, we can easily check for presence of an $s - v$ and $v - t$ path after the failure of f_1, f_2 in G by trying with all the (at most four) stored vertices in $T_{(s,t)}$. For that, we only need $O(1)$ time during query.

Now, it only remains to consider the case when the surviving nice path in $H_{(s,t)} \setminus \{f_1, f_2\}$ passes though a coupling path $Q^i_{(s,t),v}$, for some vertex v lying on the bottom $\Theta(n/\sqrt{|\mathcal{P}|})$ length subpath of a outer strand, that starts from some vertex u lying on the top $\Theta(n/\sqrt{|\mathcal{P}|})$ length subpath of a outer strand. Informally, we only need to consider the scenario when both the outer strands are of length $O(n/\sqrt{|\mathcal{P}|})$. For simplicity, from now on we continue the description with this assumption. Intuitively, this enables us to look into a smaller graph (which need not be a subgraph of the original graph). We build an *auxiliary graph* $A_{(s,t)}$ (see Figure 2) from $H_{(s,t)}$. We define the auxiliary graph entirely on the vertex set of the outer strands ($P^1_{(s,t)}$, $P^2_{(s,t)}$). First, we add both the outer strands (i.e. all their edges) to the auxiliary graph. Next, we add *auxiliary edges* between the vertices if and only if there is a path between them in $H_{(s,t)}$ that is edge-disjoint with the outer strands. Then, We construct a 2-FTRO (having query time $O(1)$) for $A_{(s,t)}$ using [17]. We do this for all pair $(s, t) \in \mathcal{P}$. Since each auxiliary graph is defined over a set of $O(n/\sqrt{|\mathcal{P}|})$ sized vertex set, we need total $O(n\sqrt{|\mathcal{P}|})$ space. This finishes the description of our data structure. So, the final data structure consists of 2-FTROs of the vertices in the hitting set and 2-FTRO computed over $A_{(s,t)}$'s. Hence, the size of the whole data structure is $O(n\sqrt{|P|}\log n)$.

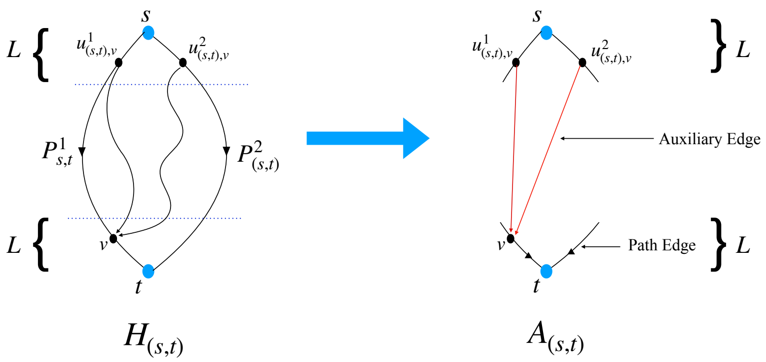

Figure 2 Auxiliary graph $A_{(s,t)}$ constructed from $H_{(s,t)}$.

It is not hard to see that any $s - t$ path of $H_{(s,t)}$ also leads to a valid $s - t$ path in $A_{(s,t)}$ and vice versa. However, it is not immediate that it will be the case even after the failure of f_1, f_2. The difficulty arises because many paths in $H_{(s,t)}$ now map to one path in $A_{(s,t)}$. We show that it is indeed the case that there is an $s - t$ path in $H_{(s,t)} \setminus \{f_1, f_2\}$ if and only if there is an $s - t$ path in the auxiliary graph $A_{(s,t)} \setminus \{f_1, f_2\}$ (given $P^1_{(s,t)}$, $P^2_{(s,t)}$ are of length at most $O(n/\sqrt{P})$). The actual description is slightly more involved because we cannot make any assumption on the length of $P^1_{(s,t)}$, $P^2_{(s,t)}$. Essentially, we need only to consider the top and bottom $O(n\sqrt{|\mathcal{P}|})$ portion of the outer strands and define the auxiliary graph over them. Then we prove the above claim without any assumption on the length of the outer strands. The guarantee on the existence of a nice $s - t$ path in $H_{(s,t)}$ after at most two failures comes handy in this case. Recall, in a nice path, there is at most one coupling sub-path. If both the endpoints of this coupling sub-path lie on the top and bottom $O(n\sqrt{|\mathcal{P}|})$ portion of the outer strands, we get an auxiliary edge. As a result, we get an $s-t$ path in the auxiliary graph after the failures. We refer the readers to the full version for the details. Note, without the guarantee of a nice path, there could be many coupling sub-paths in a surviving $s-t$ path after failures. As a result, we may not get an auxiliary edge in our auxiliary graph.

During the query for a pair $(s,t) \in \mathcal{P}$, it suffices to either place $O(1)$ many reachability queries on the first part of our data structure or check the presence of an $s-t$ path in the auxiliary graph $A_{(s,t)}$. Hence, we get the overall query time to be only $O(1)$.

To remove the $O(\log n)$ factor from the size bound, we use the concept of *sparsifier with slack* [13, 27, 21, 9]. The extra $O(\log n)$ factor was coming due to the construction of the greedy hitting set. We show that if we prematurely terminate the same greedy hitting set algorithm, we get a *fractional hitting set* that hits a constant fraction of the input sets. As a consequence, we get a pairwise 2-FTRO with slack of size $O(n\sqrt{|\mathcal{P}|})$. Then we use an argument similar to that in [9] to get the same size bound for the (standard) pairwise 2-FTRO.

Optimality of pairwise dual fault-tolerant oracle

In this paper, we show that for pairwise 2-FTRO, our $O(n\sqrt{|\mathcal{P}|})$ bound is essentially tight up to the word size (Theorem 3). Actually, we show that for any source-wise 2-FTRO for a designated source set S, the trivial $O(n|S|)$ upper bound followed from [17], is tight. To do that, we first provide a $\Omega(n|S|)$ lower bound for source-wise 2-FTRS (leading to Theorem 8) by constructing a hard instance. Then we extend that lower bound to source-wise 2-FTRO using communication complexity.

We construct the hard instance for 2-FTRS as follows. Take any integer N, r. We consider two r-sized sets of vertex-disjoint (directed) paths each of length N. Then include a (directed) complete bipartite graph between the left set of r vertices and the right set of r vertices, for each level $k \in [N]$ (See Figure 3). The set S contains the start vertices of the paths in the left set and the terminal vertices of the paths in the right set. So, $|S| = 2r$. The number of vertices and the edges in this graph are $n = 2Nr$ and $\Theta(Nr^2)$ respectively. It is not difficult to observe that each edge in this graph must be present in any 2-FTRS for this graph with the pair set $\mathcal{P} = S \times S$ (See Figure 3). Consequently, we get a $\Omega(n|S|) = \Omega(n\sqrt{|\mathcal{P}|})$ lower bound.

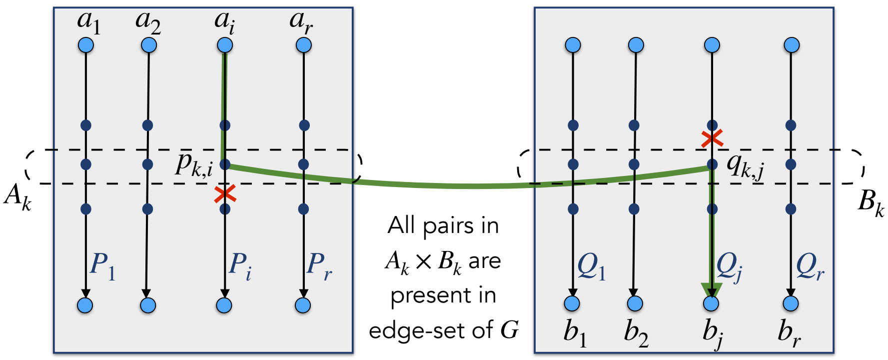

Figure 3 Dual fault-tolerant reachability preserver.

We then extend the above construction to 2-FTRO. Note, FTRO is a Boolean data structure that only decides reachability between a pair of vertices - does not report a path (if exists). If the data structure also reports a path (if exists), then using a standard information-theoretic argument, it is possible to show that the data structure must be of a size equal to that of a FTRS. Such an argument does not work in general for FTRO. This is a general hurdle that we need to overcome to extend the lower of FTRS to FTRO. Fortunately, the hard instance we constructed for FTRS allows us to provide a reduction from (a close variant of) the *Index* problem. Then we use the randomized one-way communication complexity lower bound for the Index problem [28] to get a lower bound of $\Omega(n|S|)$ on the size (in bits) of any 2-FTRO. It is worth mentioning that our lower bound holds irrespective of the query time of 2-FTRO.

Pairwise 1-FTRO: Upper and lower bound

The above lower bound only holds for dual failures. So it remains open whether for single failure we can attain better than $O(n\sqrt{|\mathcal{P}|})$ size bound with $o(n)$ (ideally, constant) query time. In this section, we provide a construction of 1-FTRO of size $O(n + |\mathcal{P}|\sqrt{n})$ and query time $O(1)$. We show the construction by first designing a reachability oracle of size $O(n + |\mathcal{P}|\sqrt{n})$ resilient to single vertex failure. Then we extend that vertex fault-tolerant data structure to an edge fault-tolerant data structure of the same size.

One of the key ingredients of our construction is a clever data structure that could answer all-pairs reachability query under single vertex failure as long as all three vertices involved in the query belong to a cut-set. Formally, for a pair (s, t), let C be any subset of $s - t$ cut-vertices in the graph G. We build an $O(|C|)$ size data structure that, for any three vertices $x, y, z \in C$, decides the reachability between y, z upon failure of vertex x, in constant time. This data structure is inspired by the loop-nesting-forest [36]. We consider the ordering σ among the cut-vertices in C. Next, we build a predecessor forest and a successor forest with respect to this ordering σ. In the predecessor forest, the parent of any node w is the immediate predecessor u such that u, w are strongly connected even after the removal of all the predecessors of u from G. We symmetrically build the successor forest. Using constantly many Lowest Common Ancestor (LCA) and Level Ancestor (LA) queries on these forests, we can now decide the reachability between y, z upon failure of the vertex x (for any $x, y, z \in C$). By deploying any standard linear space LCA/LA data structure, we attain $O(|C|)$ space-bound and $O(1)$ query time. We refer the readers to the full version for the details.

Next, we use the above data structure to construct a pairwise fault-tolerant reachability oracle for single vertex failure. Observe, to decide the reachability between a pair upon failure, we need to check whether the failure vertex is a cut-vertex for that pair or not. At the high level, we need to keep the information about the cut-vertex-set for each pair. However, we cannot store them explicitly using small space. We first consider the cut-vertex-set for all the pairs in $\mathcal{P}$. Then we identify a *core pair-set* as follows: Take any pair in $\mathcal{P}$ with at least $\sqrt{n}$ cut-vertices, and add it into the core pair-set. Then iteratively add pairs from $\mathcal{P}$ with at least $\sqrt{n}$ new/uncovered (not part of the cut-vertex-set of any previously added pair) cut-vertices. For each pair added in the core pair-set, it *owns* the corresponding cut-vertices that were uncovered till then. Once we get the core pair-set, we are left with pairs each having at most $\sqrt{n}$ uncovered cut-vertices. Further note, the size of the core pair-set is at most $O(\sqrt{n})$. Consider the union of all the cut-vertices of the core pair-set, and build the previously mentioned data structure on it. For the remaining pairs, we store the uncovered cut-vertices associated with them using any static dictionary data structure. Note, each pair might have cut-vertices that are owned by some core pair. To keep track of them, for each pair and each core pair, we store the first and the last cut-vertex shared by them. These all constitute the final data structure. It is not hard to see that the size is $O(n + |\mathcal{P}|\sqrt{n})$. We show that it suffices to either perform a query on the previously mentioned data structure or check whether the failure vertex is a cut-vertex in the dictionary structure during the query. (See the full version.)

Now, we extend the vertex fault-tolerant oracle to edge fault-tolerant oracle. We consider the set $\mathcal{C}$ of all the cut-edges for all the pairs in $\mathcal{P}$. Then we find the subset $\mathcal{C}_0$ that only contains the edges whose endpoints are strongly connected in G, but not after the failure of that edge. Observe, all the edges in $\mathcal{C}$ must be part of any strong-connectivity certificate, and all the edges in $\mathcal{C}_0$ must be part of any reachability preserver without any failure. Thus $|\mathcal{C}| = O(n)$ and $|\mathcal{C}_0| = O(n + \min\{|\mathcal{P}|\sqrt{n}, (n|\mathcal{P}|)^{2/3}\})$ by [1]. Next, we construct a new graph with $n + |\mathcal{C}_0|$ vertices such that checking the reachability upon an edge failure in the original graph reduces to checking the reachability (perhaps between a different pair) upon vertex failure in the new graph. As a consequence, we get an $O(n + |\mathcal{P}|\sqrt{n})$ sized 1-FTRO with constant query time.

We complement our upper bound result with a lower bound of $\Omega(n^{2/d+1}|\mathcal{P}|^{(d-1)/d})$ size (in bits) for any $d \geq 2$. We prove our lower bound by establishing a connection between the optimal size of a pairwise k-FTRO and that of a pairwise $(k-1)$-FTRS. In particular, we show that the optimal size of FTRO for any n-node graph and p-sized pair-set is at least that of $(k-1)$-FTRS for any $n/2$-node graph and p-sized pair-set. Such a connection between FTRO and FTRS is entirely new. To prove this relation, we use information-theoretic encoding-decoding argument. (See the full version.) Then the lower bound of 1-FTRO follows from the lower bound of reachability preserver without any failure by [1].

Pairwise 2-FTRS: Upper bound

We have already shown a lower bound of $\Omega(n\sqrt{|\mathcal{P}|})$ on the size of a pairwise 2-FTRS. However, so far we only known of $O(n|\mathcal{P}|)$ size 2-FTRS for any n-node graph G and pair set $\mathcal{P}$. In this work, we provide a deterministic polynomial time construction of a pairwise 2-FTRS of size $O(n^{4/3}|\mathcal{P}|^{1/3})$. For simplicity, below, we briefly describe a construction that provides a slightly weaker bound, in particular, $O(n\sqrt{|\mathcal{P}|}\log n)$. In the actual construction, we get rid of the $\log n$ factor by constructing a preserver with slack and then using the result of [9] - an idea similar to that used in pairwise 2-FTRO described earlier. Before proceeding further, let us emphasize that the main underlying idea behind our construction could be generalized to k-FTRS, for any $k \geq 1$, albeit with the help of randomization. We describe that generic construction later.

To get a sparse 2-FTRS for a node-pair set $\mathcal{P}$, we perform two-step sparsification. First, we apply our alternate construction of 2-FTRS for each of the pairs of $\mathcal{P}$. Then take a union of all of these subgraphs to get a $O(n|\mathcal{P}|)$ size intermediate subgraph H_{inter}, which is clearly a 2-FTRS for $\mathcal{P}$. Next, we further sparsify this intermediate subgraph. Similar to the technique used in oracle construction, we consider the top and bottom $\Theta(n^{2/3}|P|^{-1/3})$ portions of the outer strands $(P^1_{(s,t)}, P^2_{(s,t)})$ of $H_{s,t}$, for each $(s,t) \in \mathcal{P}$. Next, we construct a greedy hitting set containing $O(n^{1/3}|\mathcal{P}|^{1/3}\log n)$ vertices that intersects all these subpaths. We compute linear-sized single-source and single-destination 2-FTRS for each of these vertices in the hitting set using [4]. Let H_1 be the union of all these single-source and single-destination 2-FTRS. So, H_1 is of size $O(n^{4/3}|\mathcal{P}|^{1/3}\log n)$.

Consider any two failure edges f_1, f_2 and a pair $(s,t) \in \mathcal{P}$. If there is a surviving path in $G \setminus \{f_1, f_2\}$, we know that there is a *nice* $s - t$ path in $H_{(s,t)} \setminus \{f_1, f_2\}$. Using an argument similar to that in the oracle construction, if that nice path follows the top or bottom $\Theta(n^{2/3}|P|^{-1/3})$ portion of a outer strand, then there is also an $s - t$ path in $H_1 \setminus \{f_1, f_2\}$. So now on, it suffices to look into the case when the surviving nice path in

$H_{(s,t)} \setminus \{f_1, f_2\}$ passes though a coupling path $Q^i_{(s,t),v}$, for some vertex v lying on the bottom $\Theta(n^{2/3}|P|^{-1/3})$ length subpath of a outer strand, that starts from some vertex u lying on the top $\Theta(n^{2/3}|P|^{-1/3})$ length subpath of a outer strand.

If we could include the top and bottom $\Theta(n^{2/3}|P|^{-1/3})$ portion of the outer strands, and all the "essential" coupling path $Q^i_{(s,t),v}$'s with endpoints lying on the top and bottom $\Theta(n^{2/3}|P|^{-1/3})$ portions of the outer strands, we will be done. We indeed consider a union of the top and bottom $\Theta(n^{2/3}|P|^{-1/3})$ portion of the outer strands (for all $(s, t) \in \mathcal{P}$), and let us denote that by H_2. So, H_2 is of size $O(n^{2/3}|P|^{2/3})$. Unfortunately, we do not have a guarantee on the length of the coupling paths. Thus, if we include all the required coupling paths, we cannot argue about the sparsity of the final subgraph. (This portion of the construction differs significantly from that of our oracle construction.) We consider the subgraph obtained by taking a union of all the "essential" coupling paths with endpoints lying on the top and bottom $\Theta(n^{2/3}|P|^{-1/3})$ portions of the outer strands. (Let us denote this union by B.) Then we sparsify this subgraph further. For that purpose, we first isolate all the "high frequency" vertices (iteratively) and remove all the coupling paths containing them. Since total number of coupling paths in this subgraph is only $\Theta(n^{2/3}|P|^{2/3})$, we end up with "a few" ($O(n^{1/3}|P|^{1/3})$) high frequency vertices. Now, observe, in the remaining subgraph (denoted as H_4), degree of each vertex is "small" (at most $O(n^{1/3}|P|^{1/3})$). Next, for each of the high-frequency vertices, compute linear-sized single-source and single-destination 2-FTRS, and take a union of them to form a subgraph H_3. The union of H_1, H_2, H_3 and H_4 constitute the final subgraph.

It is not difficult to see that a surviving nice $s - t$ path in $H_{(s,t)} \setminus \{f_1, f_2\}$ either passes through one of the high frequency vertices, in which case we get an $s - t$ path in H_3; or is included in $H_2 \cup H_4$. Thus So, the union of all H_1, H_2, H_3 and H_4 will be a 2-FTRS for $\mathcal{P}$. It is worth mentioning that the correctness proof works only because of a guarantee of the existence of the nice path. Note, a nice path follows at most one coupling path as a subpath. Thus either that nice path follows the top or bottom $\Theta(n^{2/3}|P|^{-1/3})$ portion of the outer strands, or the coupling sub-path is part of B, which we further sparsify to get H_3 and H_4. Without the guarantee of a nice path, there could be many coupling sub-paths in a surviving $s - t$ path after failures. Endpoints of these coupling sub-paths may not lie on the top or bottom $\Theta(n^{2/3}|P|^{-1/3})$ portion of the outer strands. As a result, we miss them in B. As a consequence, $H_3 \cup H_4$ could not capture those coupling sub-paths.

Observe, each of the H_i's is of size at most $O(n^{4/3}|\mathcal{P}|^{1/3} \log n)$ (the $\log n$ factor is only there for H_1). So the total size is also $O(n^{4/3}|P|^{1/3} \log n)$.

Pairwise k-FTRS: Beating the $O(2^k n|\mathcal{P}|)$ bound

Currently, for any $k \geq 1$, we only know there always exists a pairwise k-FTRS of size $O(2^k n|\mathcal{P}|)$ for any pair set $\mathcal{P}$ [4]. Previously, we beat the above bound for $k = 2$. We cannot directly extend that bound for k-FTRS. The main obstacle is that for pairwise 2-FTRS, we have used a particular structure of a 2-FTRS for a single pair (i.e., the subgraph $H_{(s,t)}$). It is pretty difficult get such kind of structure for k-FTRS for any $k > 2$. However, the main underlying idea behind our pairwise 2-FTRS construction is to handle the "long" and "short" surviving paths separately. Informally, in the case of 2-FTRS, the hitting set helps us to look into only the short paths in $H_{(s,t)}$. We do a similar thing for k-FTRS using randomization.

Take a parameter ℓ, whose value we will fix later. We sample a uniformly random subset W of vertices of size $\tilde{O}(kn/\ell)$. Then we build a single-source and single-destination k-FTRS from those vertices. Let us denote the union of all these k-FTRSs as H_1. The size of H_1 is $O(k2^k n^2/\ell)$ by [4]. Suppose, for any failure edge-set F of size at most k, the length of any shortest $s-t$ surviving path in $G \setminus F$ is at least ℓ. Consider an (arbitrarily chosen) shortest surviving $s-t$ path in $G \setminus F$ (which is of length at least ℓ). Then it is not hard to argue that W intersects that path with high probability. Thus we get an $s-t$ path in $H_1 \setminus F$. This part is similar to what we get in 2-FTRS using greedy hitting set.

So now on, we only need to handle the case when a shortest surviving path is of length at least ℓ. To do this, roughly speaking, we enumerate over all possible failure-set of size at most k and add a shortest surviving path in the subgraph. Trivially, we get a bound of only $O(n^k)$ on the number of possible failure edge-sets of size at most k. However, observe, we only need to handle the case when a surviving shortest path is of length at most ℓ. So before the failure also the length of a shortest path was at most ℓ. The initial shortest path (before any failure) will get destroyed only if one or more failure edges lie on that shortest path. This observation reduces the number of possible failure-set to $O(\ell^k)$. So, for each pair, we add $O(\ell^k)$ paths, each of length at most ℓ. For all the pairs, the total number of added edges is at most $O(\ell^{k+1}|\mathcal{P}|)$. We can further improve this bound to $O(\ell^k|\mathcal{P}|)$ by enumerating the failure-set of size up to $k-1$ and then adding two maximally edge-disjoint shortest paths. (See the full version for the details.) At the end, we get a subgraph of size $O(k2^k n^2/\ell + \ell^k|\mathcal{P}|)$. By optimizing the parameter ℓ, we get a size bound of $\widetilde{O}(k\ 2^k\ n^{\frac{2k}{k+1}}\ |\mathcal{P}|^{\frac{1}{k+1}})$. Note, this size bound beats the $O(2^k n|\mathcal{P}|)$ bound whenever $|\mathcal{P}| = \omega(kn^{\frac{k-1}{k}}\log n)$.

4 Conclusion

In this paper, we study compact oracle and sparse preservers for the problem of pairwise reachability under failures. For dual failures, we provide a construction of $O(n\sqrt{|\mathcal{P}|})$ sized reachability oracle that has constant query time, along with a matching (up to the word size) lower bound. It (almost) settles down the question on the optimal sized dual fault-tolerant pairwise reachability oracle. For single failure, we achieve a bound of $O(n + \min\{|\mathcal{P}|\sqrt{n}, n\sqrt{|\mathcal{P}|}\})$ on the size of oracle with constant query time. We complement our upper bound with a $\Omega(n^{2/3}|\mathcal{P}|^{1/2})$ size (in bits) lower bound, refuting the possibility of getting a linear-sized oracle for a single failure. We would like to pose the problem of closing the current gap between the upper and lower bound for a single fault-tolerant oracle as an open problem.

In the case of reachability preserver, we show an upper bound of $O(n^{4/3}|\mathcal{P}|^{1/3})$ edges for any n-node graph and a vertex-pair set $\mathcal{P}$, for dual failures setting. This improves the naive bound of $O(n|\mathcal{P}|)$ preserver (obtained from the result known for single-source setting). In addition, we obtain a lower bound of $\Omega(n\sqrt{|\mathcal{P}|})$ on the size of our reachability preserver under dual failures. Prior to our work it was known that for single failure, we can have preserver with $O(n + \min\{|\mathcal{P}|\sqrt{n},\ n\sqrt{|\mathcal{P}|}\})$ edges. Thus our lower bound provides a striking difference between the single and dual fault-tolerant setting as it gives a separation of $n^{1/4}$ factor for $|\mathcal{P}| = \Theta(\sqrt{n})$. One immediate open question after our work is whether a sparser pairwise reachability preserver exists for dual failures.

Next, we go beyond the dual failures and consider the question of getting sparse pairwise reachability preserver resilient up to k failures for any $k \geq 1$. Can there always exist a pairwise preserver of size $o(2^k n|\mathcal{P}|)$ for any k failures? We answer this question affirmatively by providing a randomized polynomial-time construction of a $o(2^k n|\mathcal{P}|)$ size preserver. We leave the question of finding an optimal k fault-tolerant pairwise preserver as an important open problem.

References

1 Amir Abboud and Greg Bodwin. Reachability preservers: New extremal bounds and approximation algorithms. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018*, pages 1865–1883, 2018.

2 Surender Baswana, Shreejit Ray Chaudhury, Keerti Choudhary, and Shahbaz Khan. Dynamic DFS in undirected graphs: Breaking the o(m) barrier. *SIAM J. Comput.*, 48(4):1335–1363, 2019.

3 Surender Baswana, Keerti Choudhary, Moazzam Hussain, and Liam Roditty. Approximate single source fault tolerant shortest path. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018*, pages 1901–1915, 2018.

4 Surender Baswana, Keerti Choudhary, and Liam Roditty. Fault tolerant subgraph for single source reachability: generic and optimal. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016*, pages 509–518, 2016.

5 Surender Baswana and Neelesh Khanna. Approximate shortest paths avoiding a failed vertex: Near optimal data structures for undirected unweighted graphs. *Algorithmica*, 66(1):18–50, 2013.

6 Davide Bilò, Fabrizio Grandoni, Luciano Gualà, Stefano Leucci, and Guido Proietti. Improved purely additive fault-tolerant spanners. In *Algorithms - ESA 2015 - 23rd Annual European Symposium, Proceedings*, pages 167–178, 2015.

7 Davide Bilò, Luciano Gualà, Stefano Leucci, and Guido Proietti. Multiple-edge-fault-tolerant approximate shortest-path trees. In *33rd Symposium on Theoretical Aspects of Computer Science, STACS 2016*, pages 18:1–18:14, 2016.

8 Greg Bodwin. Linear size distance preservers. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 600–615, 2017.

9 Greg Bodwin. A note on distance-preserving graph sparsification. *arXiv preprint arXiv:2001.07741*, 2020.

10 Greg Bodwin, Fabrizio Grandoni, Merav Parter, and Virginia Vassilevska Williams. Preserving distances in very faulty graphs. In *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017*, pages 73:1–73:14, 2017.

11 Diptarka Chakraborty and Keerti Choudhary. New extremal bounds for reachability and strong-connectivity preservers under failures. In Artur Czumaj, Anuj Dawar, and Emanuela Merelli, editors, *47th International Colloquium on Automata, Languages, and Programming, ICALP 2020, July 8-11, 2020, Saarbrücken, Germany (Virtual Conference)*, volume 168 of *LIPIcs*, pages 25:1–25:20, 2020.

12 Diptarka Chakraborty and Debarati Das. Sparse Weight Tolerant Subgraph for Single Source Shortest Path. In David Eppstein, editor, *16th Scandinavian Symposium and Workshops on Algorithm Theory (SWAT 2018)*, volume 101 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 15:1–15:15, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

13 T-H Hubert Chan, Michael Dinitz, and Anupam Gupta. Spanners with slack. In *European Symposium on Algorithms*, pages 196–207. Springer, 2006.

14 Shiri Chechik. Fault-tolerant compact routing schemes for general graphs. *Inf. Comput.*, 222:36–44, 2013.

15 Shiri Chechik, Michael Langberg, David Peleg, and Liam Roditty. Fault-tolerant spanners for general graphs. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009*, pages 435–444, 2009.

16 Shiri Chechik, Michael Langberg, David Peleg, and Liam Roditty. f-sensitivity distance oracles and routing schemes. In *18th Annual European Symposium on Algorithms - ESA (1)*, pages 84–96, 2010.

17 Keerti Choudhary. An optimal dual fault tolerant reachability oracle. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016*, pages 130:1–130:13, 2016.

18 Don Coppersmith and Michael Elkin. Sparse sourcewise and pairwise distance preservers. *SIAM J. Discrete Math.*, 20(2):463–501, 2006.

19 Artur Czumaj and Hairong Zhao. Fault-tolerant geometric spanners. *Discrete & Computational Geometry*, 32(2):207–230, 2004.

20 Camil Demetrescu, Mikkel Thorup, Rezaul Alam Chowdhury, and Vijaya Ramachandran. Oracles for distances avoiding a failed node or link. *SIAM J. Comput.*, 37(5):1299–1318, 2008.

21 Michael Dinitz. Compact routing with slack. In *Proceedings of the twenty-sixth annual ACM symposium on Principles of distributed computing*, pages 81–88, 2007.

22 Michael Dinitz and Robert Krauthgamer. Fault-tolerant spanners: better and simpler. In *Proceedings of the 30th Annual ACM Symposium on Principles of Distributed Computing, PODC 2011*, pages 169–178, 2011.

23 Ran Duan and Seth Pettie. Dual-failure distance and connectivity oracles. In *Proceedings of the 20th Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2009*, pages 506–515, 2009.

24 Ran Duan and Seth Pettie. Connectivity oracles for failure prone graphs. In *Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC 2010, Cambridge, Massachusetts, USA, 5-8 June 2010*, pages 465–474, 2010.

25 Ran Duan and Seth Pettie. Connectivity oracles for graphs subject to vertex failures. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 490–509, 2017.

26 Manoj Gupta and Shahbaz Khan. Multiple source dual fault tolerant BFS trees. In *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017*, pages 127:1–127:15, 2017.

27 Goran Konjevod, Andrea Werneck Richa, Donglin Xia, and Hai Yu. Compact routing with slack in low doubling dimension. In *Proceedings of the twenty-sixth annual ACM symposium on Principles of distributed computing*, pages 71–80, 2007.

28 Ilan Kremer, Noam Nisan, and Dana Ron. On randomized one-round communication complexity. *Comput. Complex.*, 8(1):21–49, 1999.

29 Thomas Lengauer and Robert Endre Tarjan. A fast algorithm for finding dominators in a flowgraph. *ACM Trans. Program. Lang. Syst.*, 1(1):121–141, 1979.

30 Tamás Lukovszki. New results of fault tolerant geometric spanners. In *Algorithms and Data Structures, 6th International Workshop, WADS '99, Proceedings*, pages 193–204, 1999.

31 Hiroshi Nagamochi and Toshihide Ibaraki. A linear-time algorithm for finding a sparse k-connected spanning subgraph of a k-connected graph. *Algorithmica*, 7(5&6):583–596, 1992.

32 Merav Parter. Dual failure resilient BFS structure. In *Proceedings of the 2015 ACM Symposium on Principles of Distributed Computing, PODC 2015*, pages 481–490, 2015.

33 Merav Parter and David Peleg. Sparse fault-tolerant BFS trees. In *Algorithms - ESA 2013 - 21st Annual European Symposium, Proceedings*, pages 779–790, 2013.

34 Merav Parter and David Peleg. Fault tolerant approximate BFS structures. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014*, pages 1073–1092, 2014.

35 Mihai Patrascu and Mikkel Thorup. Planning for fast connectivity updates. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2007), October 20-23, 2007, Providence, RI, USA, Proceedings*, pages 263–271, 2007.

36 Robert Endre Tarjan. Edge-disjoint spanning trees and depth-first search. *Acta Informatica*, 6:171–185, 1976. `doi:10.1007/BF00268499`.

37 Jan van den Brand and Thatchaphol Saranurak. Sensitive distance and reachability oracles for large batch updates. In *60th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2019, Baltimore, Maryland, USA, November 9-12, 2019*, pages 424–435, 2019.