Near-Optimal Deterministic Single-Source Distance Sensitivity Oracles

Davide Bilò, Sarel Cohen, Tobias Friedrich, and Martin Schirneck

29th European Symposium on Algorithms - September 6–8, 2021

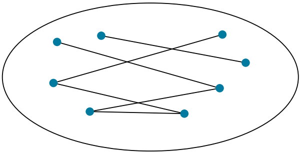
UNIVERSITÀ DEGLI STUDI DI SASSARI

Hasso Plattner Institut
Digital Engineering · Universität Potsdam

Universität Potsdam

Fault-Tolerant Data Structures

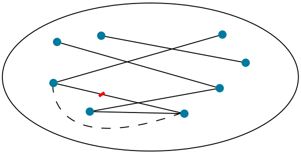
a.k.a. sensitivity data structures, algorithm for emergency planning, or failure-prone graphs



Maintain graph property $P(G)$ (distances, connectivity, ...) under edge failures.

Fault-Tolerant Data Structures

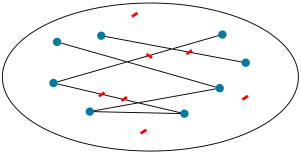
a.k.a. sensitivity data structures, algorithm for emergency planning, or failure-prone graphs



Maintain graph property $P(G)$ (distances, connectivity, ...) under edge failures.

Fault-Tolerant Data Structures

a.k.a. sensitivity data structures, algorithm for emergency planning, or failure-prone graphs

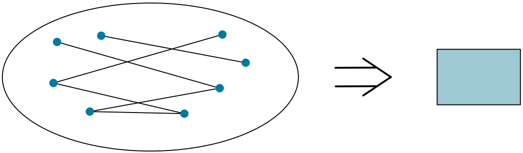


Maintain graph property $P(G)$ (distances, connectivity, ...) under edge failures.

- Fault tolerance: failures in batches, maximum number f is known.

Fault-Tolerant Data Structures

a.k.a. sensitivity data structures, algorithm for emergency planning, or failure-prone graphs

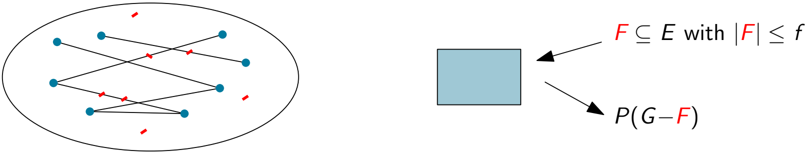


Maintain graph property $P(G)$ (distances, connectivity, ...) under edge failures.

- Fault tolerance: failures in batches, maximum number f is known.
- Data structure: preprocess once, query when needed.

Fault-Tolerant Data Structures

a.k.a. sensitivity data structures, algorithm for emergency planning, or failure-prone graphs

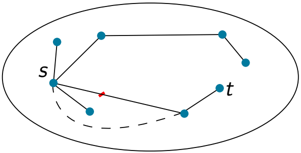


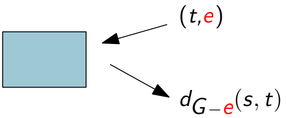
Maintain graph property $P(G)$ (distances, connectivity, ...) under edge failures.

- Fault tolerance: failures in batches, maximum number f is known.
- Data structure: preprocess once, query when needed.

Fault-Tolerant Data Structures

a.k.a. sensitivity data structures, algorithm for emergency planning, or failure-prone graphs





Maintain graph property $P(G)$ (distances, connectivity, ...) under edge failures.

- Fault tolerance: failures in batches, maximum number f is known.
- Data structure: preprocess once, query when needed.

This talk: $P(G) = d_G(s, \cdot)$ - distances from source s, for $f = 1$ failure.

Single-source distance sensitivity oracle (single-source DSO).

Single-Source Replacement Paths Problem

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

SSRP: given G with source s, compute $d_{G-e}(s,t)$ for all targets t and edges e.

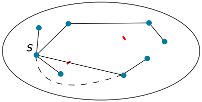
Single-Source Replacement Paths Problem

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

SSRP: given G with source s, compute $d_{G-e}(s,t)$ for all targets t and edges e.

Relevant replacement distances: e on a shortest path tree rooted in s.

$\Rightarrow$ Output size $O(n^2)$.



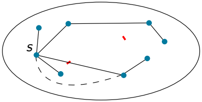
Single-Source Replacement Paths Problem

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

SSRP: given G with source s, compute $d_{G-e}(s,t)$ for all targets t and edges e.

Relevant replacement distances: e on a shortest path tree rooted in s.

$\Rightarrow$ Output size $O(n^2)$.



Undirected graphs:

- Chechik & Cohen [SODA 2019]: $\widetilde{O}(m\sqrt{n} + n^2)$ time SSRP algorithm.
  - Randomized, combinatorial, unweighted graphs.
    ("Combinatorial" algorithms: no fast matrix multiplication.)

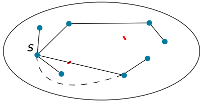
Single-Source Replacement Paths Problem

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

SSRP: given G with source s, compute $d_{G-e}(s,t)$ for all targets t and edges e.

Relevant replacement distances: e on a shortest path tree rooted in s.

$\Rightarrow$ Output size $O(n^2)$.



Undirected graphs:

- Chechik & Cohen [SODA 2019]: $\widetilde{O}(m\sqrt{n} + n^2)$ time SSRP algorithm.
  - Randomized, combinatorial, unweighted graphs.
    (“Combinatorial” algorithms: no fast matrix multiplication.)
- Grandoni & Vassilevska Williams [FOCS 2012]: $\widetilde{O}(Mn^{\omega})$ time SSRP algorithm.
  - Randomized, algebraic, integer edge weights in $[1, M]$.
    ($\omega < 2.3729$ matrix multiplication exponent.)

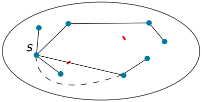
Single-Source Replacement Paths Problem

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

SSRP: given G with source s, compute $d_{G-e}(s,t)$ for all targets t and edges e.

Relevant replacement distances: e on a shortest path tree rooted in s.

$\Rightarrow$ Output size $O(n^2)$.



Undirected graphs:

- Chechik & Cohen [SODA 2019]: $\widetilde{O}(m\sqrt{n} + n^2)$ time SSRP algorithm.
  - Randomized, combinatorial, unweighted graphs.
    (“Combinatorial” algorithms: no fast matrix multiplication.)
- Grandoni & Vassilevska Williams [FOCS 2012]: $\widetilde{O}(Mn^{\omega})$ time SSRP algorithm.
  - Randomized, algebraic, integer edge weights in $[1, M]$.
    ($\omega < 2.3729$ matrix multiplication exponent.)

Both running times tight up to $n^{o(1)}$ factors (under reasonable assumptions).

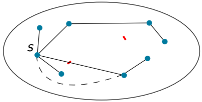
Single-Source Replacement Paths Problem

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

SSRP: given G with source s, compute $d_{G-e}(s,t)$ for all targets t and edges e.

Relevant replacement distances: e on a shortest path tree rooted in s.

$\Rightarrow$ Output size $O(n^2)$.



Undirected graphs:

- Chechik & Cohen [SODA 2019]: $\widetilde{O}(m\sqrt{n} + n^2)$ time SSRP algorithm.
  - Randomized, combinatorial, unweighted graphs.
    (“Combinatorial” algorithms: no fast matrix multiplication.)
- Grandoni & Vassilevska Williams [FOCS 2012]: $\widetilde{O}(Mn^{\omega})$ time SSRP algorithm.
  - Randomized, algebraic, integer edge weights in $[1, M]$.
    ($\omega < 2.3729$ matrix multiplication exponent.)

Both running times tight up to $n^{o(1)}$ factors (under reasonable assumptions).

Tabulation: single-source DSOs with $O(n^2)$ space and $O(1)$ query time.

Subquadratic Space Single-Source DSOs

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Quadratic space is not optimal:

- Gupta & Singh [ICALP 2018]: $\widetilde{O}(n^{3/2})$ space single-source DSO.
  - Randomized, combinatorial, unweighted graphs, $\widetilde{O}(mn)$ preprocessing time, $\widetilde{O}(1)$ query time.

Subquadratic Space Single-Source DSOs

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Quadratic space is not optimal:

- Gupta & Singh [ICALP 2018]: $\widetilde{O}(n^{3/2})$ space single-source DSO.
  - Randomized, combinatorial, unweighted graphs,
    $\widetilde{O}(mn)$ preprocessing time, $\widetilde{O}(1)$ query time.
- Bilò et al. [STACS 2018]: $\widetilde{O}(n^{3/2})$ space single-source DSO.
  - Deterministic, combinatorial, unweighted graphs,
    $\widetilde{O}(mn)$ preprocessing time, $\widetilde{O}(\sqrt{n})$ query time.

Our Result

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Idea: Best of both worlds - reduce single-source DSOs to SSRP efficiently!

Our Result

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Idea: Best of both worlds - reduce single-source DSOs to SSRP efficiently!

Upper Bounds

There are deterministic single-source DSOs with $\widetilde{O}(1)$ query time and

(i) combinatorial $\widetilde{O}(m\sqrt{n} + n^2)$ preprocessing time
for unweighted graphs, taking $O(n^{3/2})$ space;

Our Result

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Idea: Best of both worlds - reduce single-source DSOs to SSRP efficiently!

Upper Bounds

There are deterministic single-source DSOs with $\widetilde{O}(1)$ query time and

(i) combinatorial $\widetilde{O}(m\sqrt{n} + n^2)$ preprocessing time for unweighted graphs, taking $O(n^{3/2})$ space;

(ii) algebraic $\widetilde{O}(Mn^{\omega})$ preprocessing time for graphs with integer edge weights in $[1, M]$, taking $O(M^{1/2}n^{3/2})$ space.

Our Result

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Idea: Best of both worlds - reduce single-source DSOs to SSRP efficiently!

Upper Bounds

There are deterministic single-source DSOs with $\widetilde{O}(1)$ query time and

(i) combinatorial $\widetilde{O}(m\sqrt{n} + n^2)$ preprocessing time for unweighted graphs, taking $O(n^{3/2})$ space;

(ii) algebraic $\widetilde{O}(Mn^{\omega})$ preprocessing time for graphs with integer edge weights in $[1, M]$, taking $O(M^{1/2}n^{3/2})$ space.

Space Lower Bound

Any single-source DSO must take $\Omega(M^{1/2}n^{3/2})$ *bits* of space on n-vertex graphs with integer edge weights in $[1, M]$, unconditionally.

Reducing Single-Source DSOs to SSRP

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Ressources: shortest path tree rooted in s & all relevant distances $d_{G-e}(s,t)$.

Task: construct in time $O(n^2)$ a single-source DSO taking $O(M^{1/2}n^{3/2})$ space.

Reducing Single-Source DSOs to SSRP

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Ressources: shortest path tree rooted in s & all relevant distances $d_{G-e}(s,t)$.

Task: construct in time $O(n^2)$ a single-source DSO taking $O(M^{1/2}n^{3/2})$ space.

Greedily compute a set $B \subseteq V$ with $|B| = O(\sqrt{n})$ such that
every s-t-path in the tree has an $x \in B$ among its last $\sqrt{n}$ vertices.

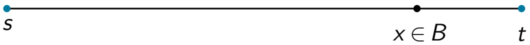
Reducing Single-Source DSOs to SSRP

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Ressources: shortest path tree rooted in s & all relevant distances $d_{G-e}(s,t)$.

Task: construct in time $O(n^2)$ a single-source DSO taking $O(M^{1/2}n^{3/2})$ space.

Greedily compute a set $B \subseteq V$ with $|B| = O(\sqrt{n})$ such that
every s-t-path in the tree has an $x \in B$ among its last $\sqrt{n}$ vertices.



Where is the failing edge? - Three cases:

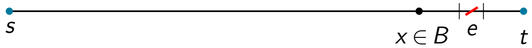
Reducing Single-Source DSOs to SSRP

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Ressources: shortest path tree rooted in s & all relevant distances $d_{G-e}(s,t)$.

Task: construct in time $O(n^2)$ a single-source DSO taking $O(M^{1/2}n^{3/2})$ space.

Greedily compute a set $B \subseteq V$ with $|B| = O(\sqrt{n})$ such that
every s-t-path in the tree has an $x \in B$ among its last $\sqrt{n}$ vertices.



Where is the failing edge? - Three cases:

1. Near case: e between x and t. (a.k.a. subtree problem)

HPI

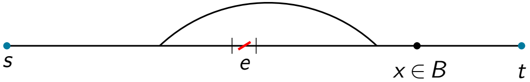
Reducing Single-Source DSOs to SSRP

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Ressources: shortest path tree rooted in s & all relevant distances $d_{G-e}(s,t)$.

Task: construct in time $O(n^2)$ a single-source DSO taking $O(M^{1/2}n^{3/2})$ space.

Greedily compute a set $B \subseteq V$ with $|B| = O(\sqrt{n})$ such that every s-t-path in the tree has an $x \in B$ among its last $\sqrt{n}$ vertices.



Where is the failing edge? - Three cases:

1. Near case: e between x and t. (a.k.a. subtree problem)
2. Far case I: e between s and x, detour merges before x. (a.k.a. jumping paths)

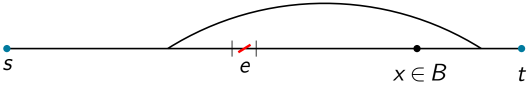
Reducing Single-Source DSOs to SSRP

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Ressources: shortest path tree rooted in s & all relevant distances $d_{G-e}(s,t)$.

Task: construct in time $O(n^2)$ a single-source DSO taking $O(M^{1/2}n^{3/2})$ space.

Greedily compute a set $B \subseteq V$ with $|B| = O(\sqrt{n})$ such that
every s-t-path in the tree has an $x \in B$ among its last $\sqrt{n}$ vertices.



Where is the failing edge? - Three cases:

1. Near case: e between x and t. (a.k.a. subtree problem)
2. Far case I: e between s and x, detour merges before x. (a.k.a. jumping paths)
3. Far case II: e between s and x, detour merges after x. (a.k.a. departing paths)

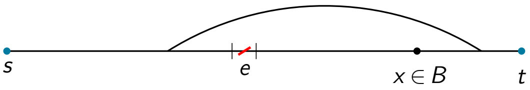
Reducing Single-Source DSOs to SSRP

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Ressources: shortest path tree rooted in s & all relevant distances $d_{G-e}(s,t)$.

Task: construct in time $O(n^2)$ a single-source DSO taking $O(M^{1/2} n^{3/2})$ space.

Greedily compute a set $B \subseteq V$ with $|B| = O(\sqrt{n})$ such that
every s-t-path in the tree has an $x \in B$ among its last $\sqrt{n}$ vertices.

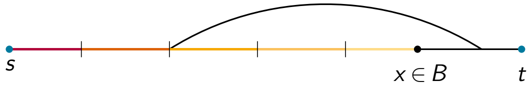


Where is the failing edge? - Three cases:

1. Near case: e between x and t. (a.k.a. subtree problem)
2. Far case I: e between s and x, detour merges before x. (a.k.a. jumping paths)
3. Far case II: e between s and x, detour merges after x. (a.k.a. departing paths)

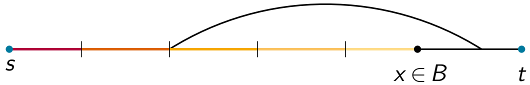
Compressing the Far Case II

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.



Compressing the Far Case II

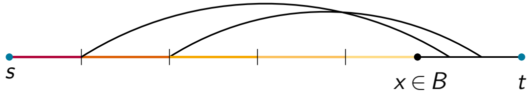
Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.



Claim: $O(\sqrt{Mn})$ segments of same replacement distance.

Compressing the Far Case II

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

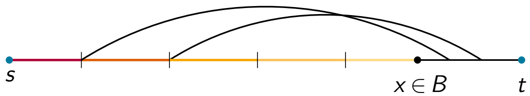


Claim: $O(\sqrt{Mn})$ segments of same replacement distance.

- *Respresentative* replacement paths with distinct lengths $\geq d_G(s,t)$.

Compressing the Far Case II

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

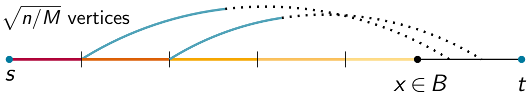


Claim: $O(\sqrt{Mn})$ segments of same replacement distance.

- *Respresentative* replacement paths with distinct lengths $\geq d_G(s,t)$.
  - Only $2\sqrt{Mn}$ representatives with length $< d_G(s,t) + 2\sqrt{Mn}$.

Compressing the Far Case II

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

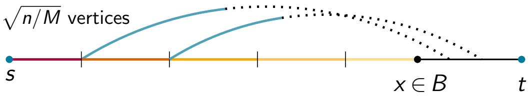


Claim: $O(\sqrt{Mn})$ segments of same replacement distance.

- *Respresentative* replacement paths with distinct lengths $\geq d_G(s,t)$.
  - Only $2\sqrt{Mn}$ representatives with length $< d_G(s,t) + 2\sqrt{Mn}$.
- Identify long representatives by their *stub* - first $\sqrt{n/M}$ vertices on detour.

Compressing the Far Case II

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

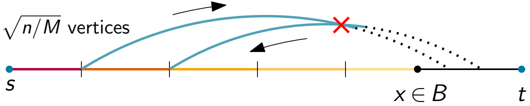


Claim: $O(\sqrt{Mn})$ segments of same replacement distance.

- *Respresentative* replacement paths with distinct lengths $\geq d_G(s,t)$.
  - Only $2\sqrt{Mn}$ representatives with length $< d_G(s,t) + 2\sqrt{Mn}$.
- Identify long representatives by their *stub* - first $\sqrt{n/M}$ vertices on detour.
- Stubs cannot intersect! (So there are only $\sqrt{Mn}$ stubs.)

Compressing the Far Case II

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

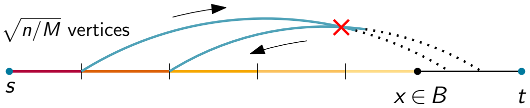


Claim: $O(\sqrt{Mn})$ segments of same replacement distance.

- *Respresentative* replacement paths with distinct lengths $\geq d_G(s,t)$.
  - Only $2\sqrt{Mn}$ representatives with length $< d_G(s,t) + 2\sqrt{Mn}$.
- Identify long representatives by their *stub* - first $\sqrt{n/M}$ vertices on detour.
- Stubs cannot intersect! (So there are only $\sqrt{Mn}$ stubs.)
  - Otherwise segment has replacement path of length $< d_G(s,t) + 2\sqrt{Mn}$.

Compressing the Far Case II

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.



Claim: $O(\sqrt{Mn})$ segments of same replacement distance.

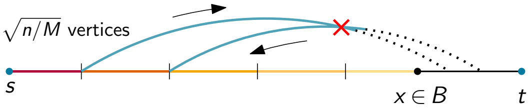
- *Respresentative* replacement paths with distinct lengths $\geq d_G(s,t)$.
  - Only $2\sqrt{Mn}$ representatives with length $< d_G(s,t) + 2\sqrt{Mn}$.
- Identify long representatives by their *stub* - first $\sqrt{n/M}$ vertices on detour.
- Stubs cannot intersect! (So there are only $\sqrt{Mn}$ stubs.)
  - Otherwise segment has replacement path of length $< d_G(s,t) + 2\sqrt{Mn}$.

At most $3\sqrt{Mn}$ representatives/segments/distances in far case II.

- Scan $d_{G-e}(s,t)$ along s-x-path - $O(n)$ time & $O(\sqrt{Mn})$ space/target t.

Compressing the Far Case II

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.



Claim: $O(\sqrt{Mn})$ segments of same replacement distance.

- *Respresentative* replacement paths with distinct lengths $\geq d_G(s,t)$.
  - Only $2\sqrt{Mn}$ representatives with length $< d_G(s,t) + 2\sqrt{Mn}$.
- Identify long representatives by their *stub* - first $\sqrt{n/M}$ vertices on detour.
- Stubs cannot intersect! (So there are only $\sqrt{Mn}$ stubs.)
  - Otherwise segment has replacement path of length $< d_G(s,t) + 2\sqrt{Mn}$.

At most $3\sqrt{Mn}$ representatives/segments/distances in far case II.

- Scan $d_{G-e}(s,t)$ along s-x-path - $O(n)$ time & $O(\sqrt{Mn})$ space/target t.

Also holds for vertex failures, but crucially depends on G being undirected.

Derandomization & Subquadratic Preprocessing

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).

DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Seen: Use SSRP to construct single-source DSOs taking $O(M^{1/2}n^{3/2})$ space.

Derandomization & Subquadratic Preprocessing

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Seen: Use SSRP to construct single-source DSOs taking $O(M^{1/2}n^{3/2})$ space.

Reduction is deterministic, but best SSRP algorithms are randomized.

Derandomization & Subquadratic Preprocessing

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Seen: Use SSRP to construct single-source DSOs taking $O(M^{1/2}n^{3/2})$ space.

Reduction is deterministic, but best SSRP algorithms are randomized.

- We derandomize the algorithms by Chechik & Cohen [SODA 2019] and Grandoni & Vassilevska Williams [FOCS 2012] in the same running time.
- Get deterministic single-source DSOs.

Derandomization & Subquadratic Preprocessing

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Seen: Use SSRP to construct single-source DSOs taking $O(M^{1/2}n^{3/2})$ space.

Reduction is deterministic, but best SSRP algorithms are randomized.

- We derandomize the algorithms by Chechik & Cohen [SODA 2019] and Grandoni & Vassilevska Williams [FOCS 2012] in the same running time.
- Get deterministic single-source DSOs.

Reduction requires quadratic time, SSRP algorithms as well.
Is this necessary for single-source DSOs (on sparse enough graphs)?

Derandomization & Subquadratic Preprocessing

Undirected graphs, n vertices, m edges, $m \geq n$, space in $O(\log n)$-bits machine words, $\widetilde{O}(.)$ hides polylog(n).
DSO - distance sensitivity oracle, SSRP - single-source replacement paths.

Seen: Use SSRP to construct single-source DSOs taking $O(M^{1/2}n^{3/2})$ space.

Reduction is deterministic, but best SSRP algorithms are randomized.

- We derandomize the algorithms by Chechik & Cohen [SODA 2019] and Grandoni & Vassilevska Williams [FOCS 2012] in the same running time.
- Get deterministic single-source DSOs.

Reduction requires quadratic time, SSRP algorithms as well.
Is this necessary for single-source DSOs (on sparse enough graphs)?

Subquadratic Preprocessing

For undirected graphs with integer edge weights in $[1, M]$ having $m = O(n^{(5/4)-\varepsilon}/M^{7/4})$ edges, there is a *randomized* single-source DSO with $\widetilde{O}(n^{2-\varepsilon/2})$ preprocessing time, $\widetilde{O}(1)$ query time, and $O(M^{1/2}n^{3/2})$ space.

Conclusion

1. Improved deterministic single-source DSOs for undirected graphs with near-optimal preprocessing time, query time, and space.

2. First subquadratic preprocessing algorithm for sparse graphs.

Conclusion

1. Improved deterministic single-source DSOs for undirected graphs with near-optimal preprocessing time, query time, and space.
2. First subquadratic preprocessing algorithm for sparse graphs.

Open Problems

- All-pairs DSOs for digraphs must take $\Omega(n^2)$ bits of space. [Thorup & Zwick JACM 2005]
  Is this true for the single-source case?

Conclusion

1. Improved deterministic single-source DSOs for undirected graphs with near-optimal preprocessing time, query time, and space.
2. First subquadratic preprocessing algorithm for sparse graphs.

Open Problems

- All-pairs DSOs for digraphs must take $\Omega(n^2)$ bits of space. [Thorup & Zwick JACM 2005]
  Is this true for the single-source case?
- How to solve SSRP efficiently for vertex failures?

Conclusion

1. Improved deterministic single-source DSOs for undirected graphs with near-optimal preprocessing time, query time, and space.
2. First subquadratic preprocessing algorithm for sparse graphs.

Open Problems

- All-pairs DSOs for digraphs must take $\Omega(n^2)$ bits of space. [Thorup & Zwick JACM 2005]
  Is this true for the single-source case?
- How to solve SSRP efficiently for vertex failures?

Thank you.