Popular Conjectures and Dynamic Problems

Amir Abboud

Stanford University

“Hardness in P” workshop, STOC 15’

This talk:

➡ Overview of recent lower bounds for dynamic problems

➡ Simple and powerful proofs

➡ Interesting Open Questions.

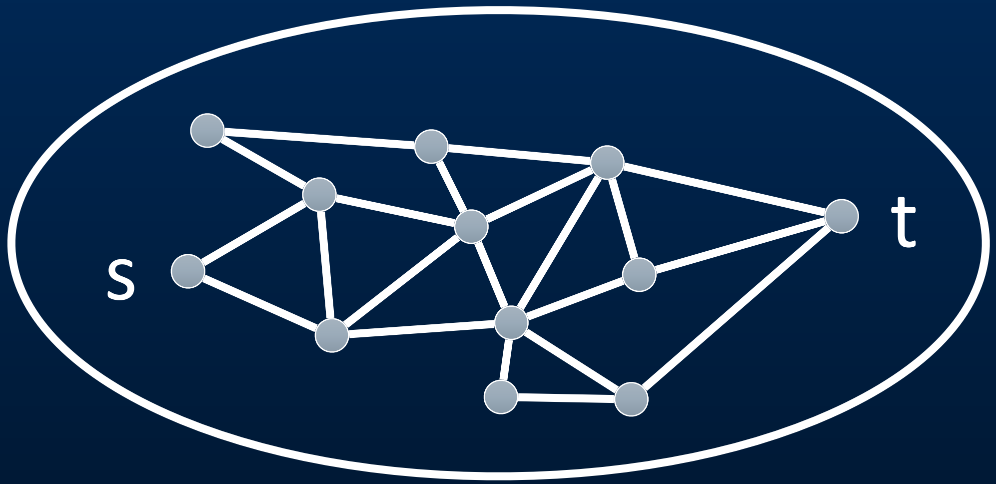
Dynamic Problems

Dynamic (undirected) Connectivity

Input: an undirected graph G

Updates: Add or remove edges.

Query: Are s and t connected?



Trivial algorithm: O(m) updates.
[Thorup STOC 01']: $O(\log m\,(\log\log m)^3)$ amortized time per update.

[Pătraşcu - Demaine STOC 05']:
$\Omega(\log m)$ Cell-probe lower bound.

Great!

Dynamic Problems

Dynamic (directed) Reachability

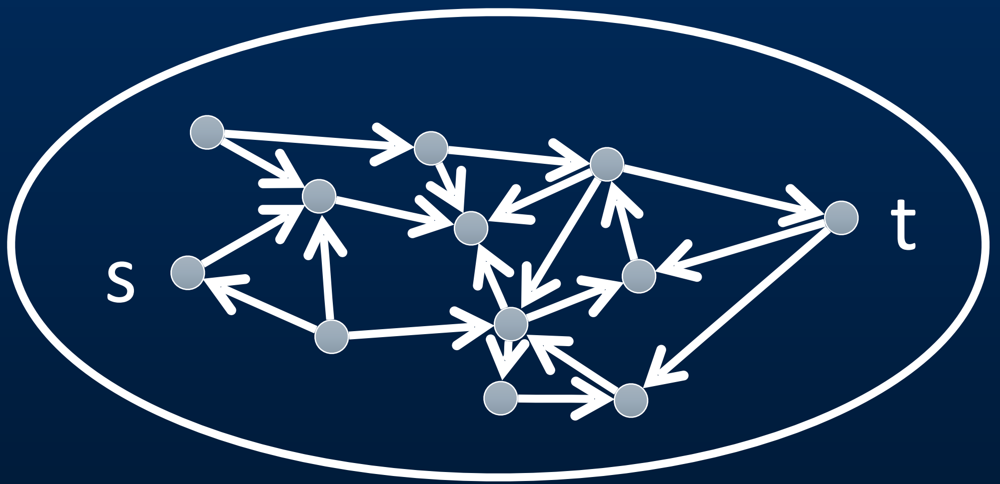
Input: A directed graph G.

Updates: Add or remove edges.

Query:

s,t-Reach: Is there a path from s to t?

#SSR: How many nodes can s reach?



Trivial algorithm: O(m) time updates

Using fast matrix multiplication
[Sankowski FOCS 04'] $O(n^{1.57})$

Not great.

Best cell probe lower bound still $\Omega(\log m)$

Many Examples

Problem	Upper bound	(Unconditional) Lower bound
s,t-Reach	O(m) or O(n	$\Omega(\log m)$
#SSR		
Strongly Connected Components		
Maximum Matching		
Connectivity with node updates	O(m	
Approximate Diameter	O(mn)	

Many successes for the partially dynamic setting and related problems.

Huge gaps - what is the right answer?

This talk:

Much higher lower bounds
via the "Hardness in P" approach

3SUM Lower Bounds

Theorem [Pătraşcu STOC10']: The 3-SUM conjecture implies polynomial lower bounds for many dynamic problems.

3-SUM: Given n integers, are there 3 that sum to 0?

The 3-SUM Conjecture: "No $O(n^{2-e})$ time algorithm"

A very cool series of reductions…

Problem	Upper bound	(3-SUM) Lower bound
s,t-Reach	O(m) or O(n	m
#SSR		
Connectivity with node updates	O(m	

for some a>0

No poly log updates for Reachability!

3SUM Lower Bounds

[A-VW FOCS 14'], [Kopelowitz - Pettie - Porat. Arxiv 14']

Optimized Pătraşcu's reductions and added problems to the list

Problem	Upper bound	(3-SUM) Lower bound
s,t-Reach	O(m) or O(n	m
#SSR		
Strongly Connected Components		
Maximum Matching		
Connectivity with node updates	O(m	
Approximate Diameter	O(mn)	

Some steps in the reduction are lossy - stuck at $m^{1/3}$.

3SUM might not be the most appropriate…

BMM Lower Bounds

[A-VW FOCS 14']

The BMM conjecture implies tight lower bounds for combinatorial algorithms!

The BMM conjecture:

"No $O(n^{3-e})$ time combinatorial algorithm for Boolean Matrix Multiplication"

<table>
<tr><th>Problem</th><th>(combinatorial)
Upper bound</th><th>(BMM)
Lower bound</th><th>(3-SUM)
Lower bound</th></tr>
<tr><td>#SSR</td><td rowspan="4">O(m)</td><td rowspan="4">m</td><td rowspan="5">m</td></tr>
<tr><td>Strongly Connected Components</td></tr>
<tr><td>s,t-Reach</td></tr>
<tr><td>Maximum Matching</td></tr>
<tr><td>Approximate Diameter</td><td>O(mn)</td><td>m</td></tr>
</table>

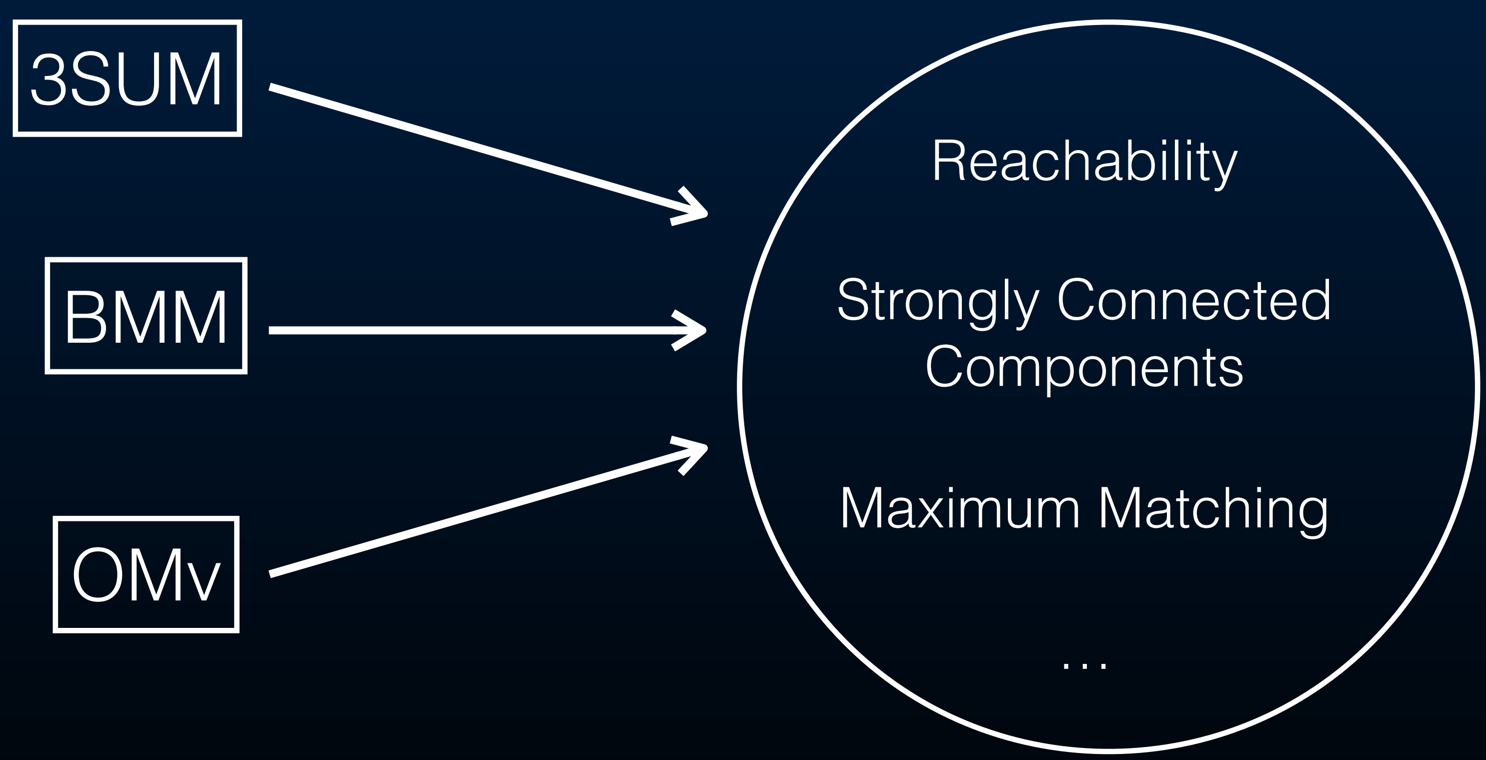
Any improvement for these problems will probably have to use fast matrix mult.

OMv Lower Bounds

[Henzinger - Krinninger - Nanongkai - Saranurak STOC 15']

Most BMM lower bounds hold for non-combinatorial algorithms as well, under the Online Matrix Vector Multiplication Conjecture.

More details tomorrow!



Each "lower bound" has different advantages.

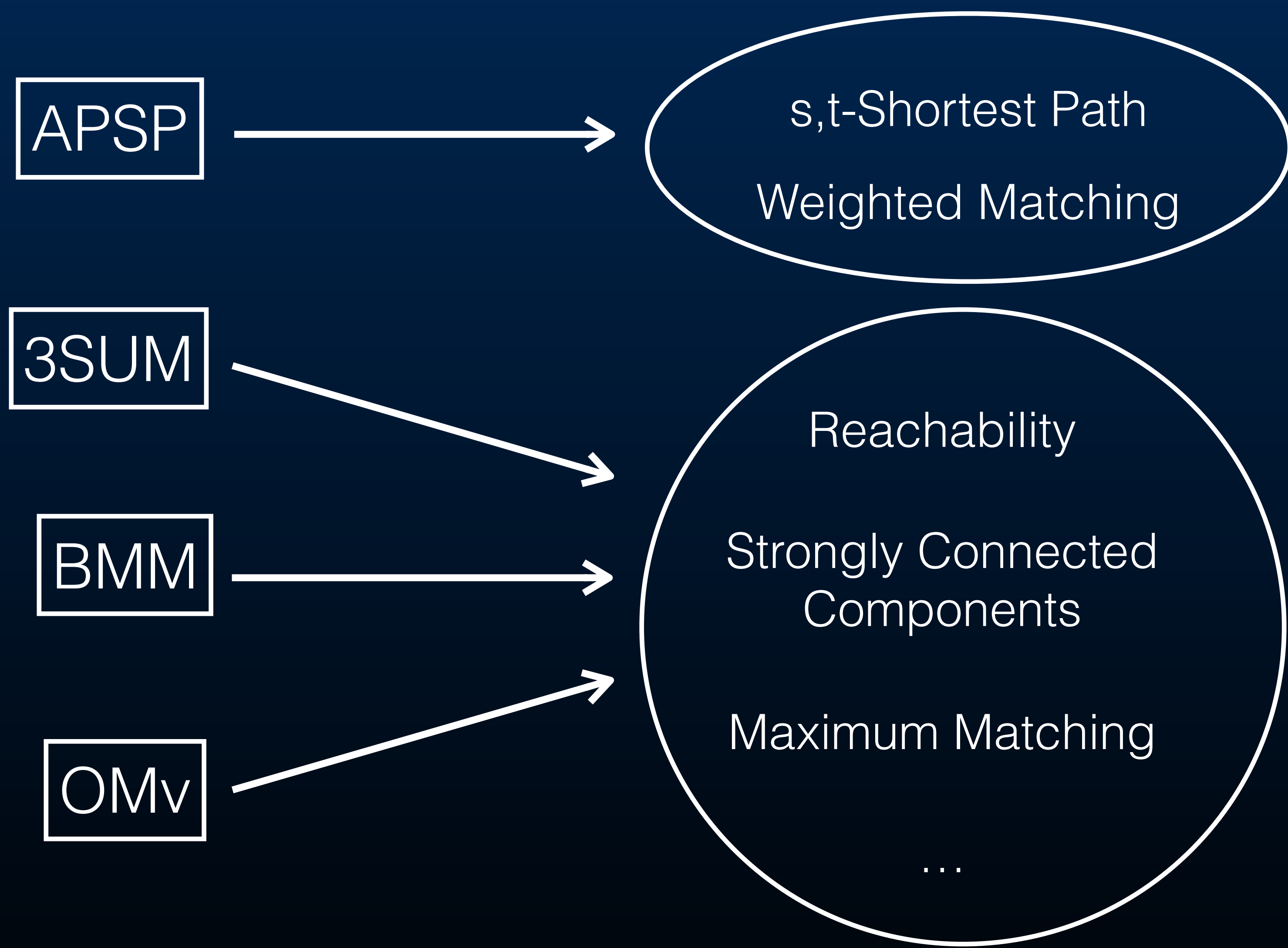
APSP Lower Bounds

[A-VW FOCS 14']

The APSP conjecture implies tight lower bounds for some weighted problems.

The APSP conjecture:

"No $O(n^{3-e})$ time algorithm for All-Pairs-Shortest-Paths"



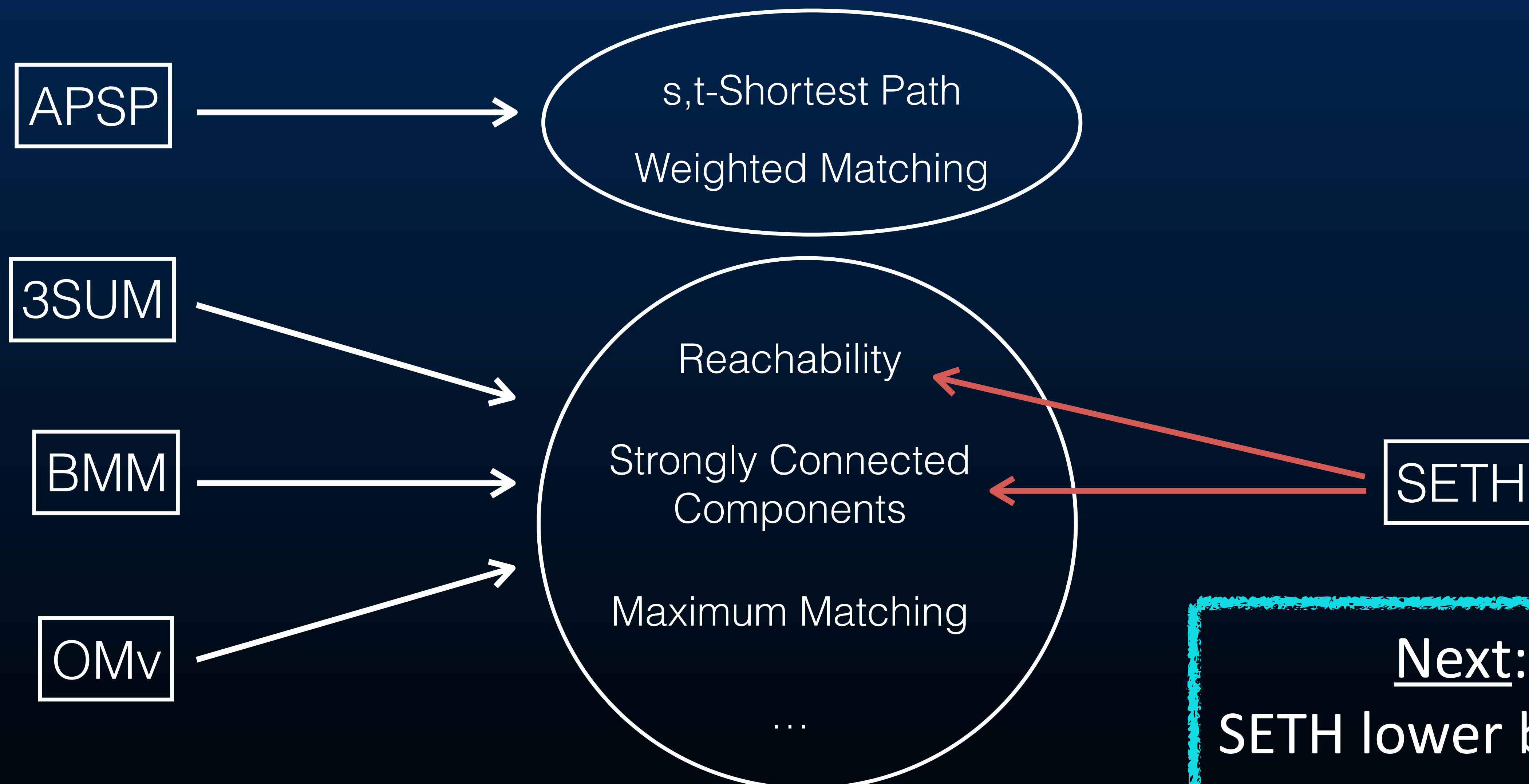
$\Omega(n^2)$ per update!

Different conjectures are better for explaining different barriers

SETH Lower Bounds

[A-VW FOCS 14'] SETH implies very high lower bounds!

SETH:
"CNF-SAT cannot be solved in *(2-e)*^{n} time"



Next:
SETH lower bounds with cool proofs!

Talk outline:

➡ Overview

➡ Lower bound for dynamic Reachability

➡ Lower bound for dynamic Diameter

➡ Conclusions

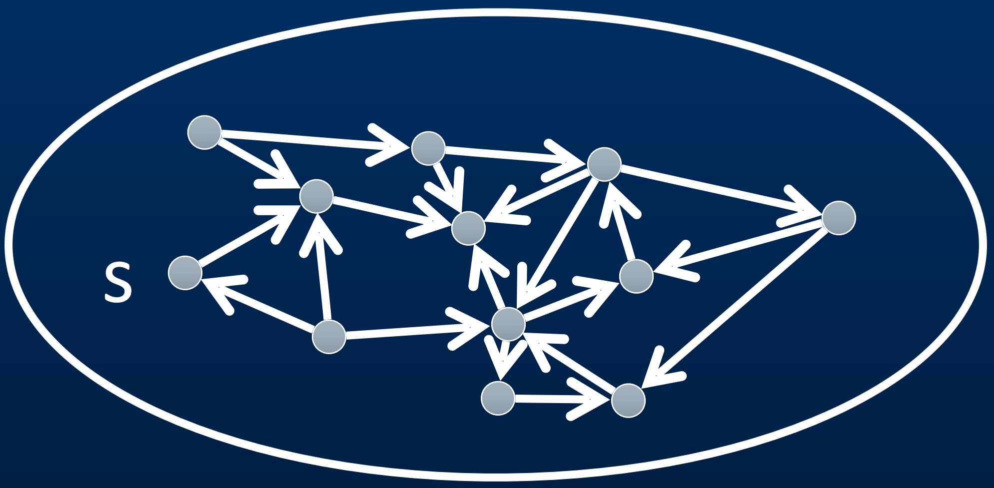
Single Source Reachability

Input: A directed graph G.

Updates: Add or remove edges.

Query:

#SSR: How many nodes can s reach?



Trivial algorithm: O(m) updates.
OMv lower bound: $\Omega(m^{1/2})$ updates.

Theorem [**A.** - VW FOCS 14']:
If **dynamic #SSR** can be solved with $O(m^{1-e})$ update and query times, then **OVP** can be solved in $O(n^{2-e})$ time (and SETH is false).

Theorem [**A.** - VW FOCS 14']:
If **dynamic #SSR** can be solved with $O(m^{1-e})$ update and query times,
then **OVP** can be solved in $O(n^{2-e})$ time (and SETH is false).

Proof outline:

Orthogonal Vectors

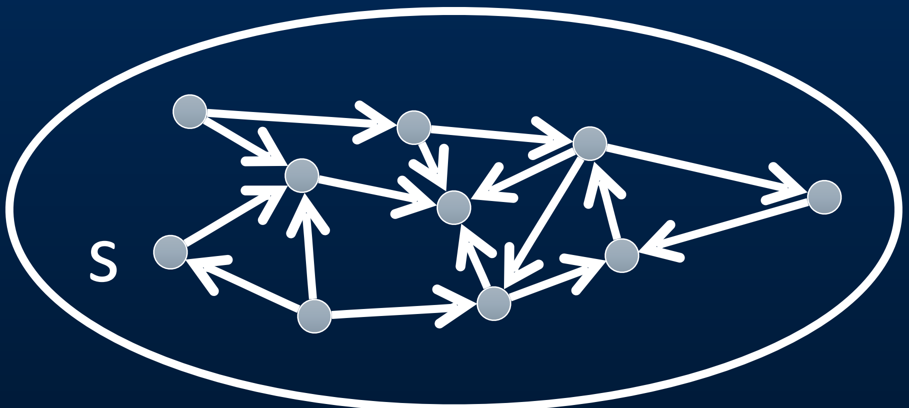
(0,0,1,…,1)	(0,0,1,…,1)
(0,1,1,…,1)	(1,0,1,…,1)
…	…
(1,0,1,…,0)	(1,1,1,…,1)

(1,0,1,…,0)

(0,0,1,…,1)

Given two lists of n vectors in $\{0,1\}^d$
is there an orthogonal pair?

dynamic #SSR



#SSR asks how many nodes can s reach?

Graph G on ***m=O(nd)*** *nodes and edges,*
O(nd) updates and queries

OVP in ***~O(n^1.9^)*** time ⟵ ***O(nd)*** updates/queries in ***~O(n^1.9^)*** time ⟵ Amortized ***O(m^0.9^)*** update/query time

(refutes SETH)

d=polylog(n), m=~O(n)

Previous talk [Roditty - VW STOC 13']:

If **diameter** be solved in $O(n^{2-e})$ times,

then OVP can be solved in $O(n^{2-e})$ time (and SETH is false).

Orthogonal Vectors

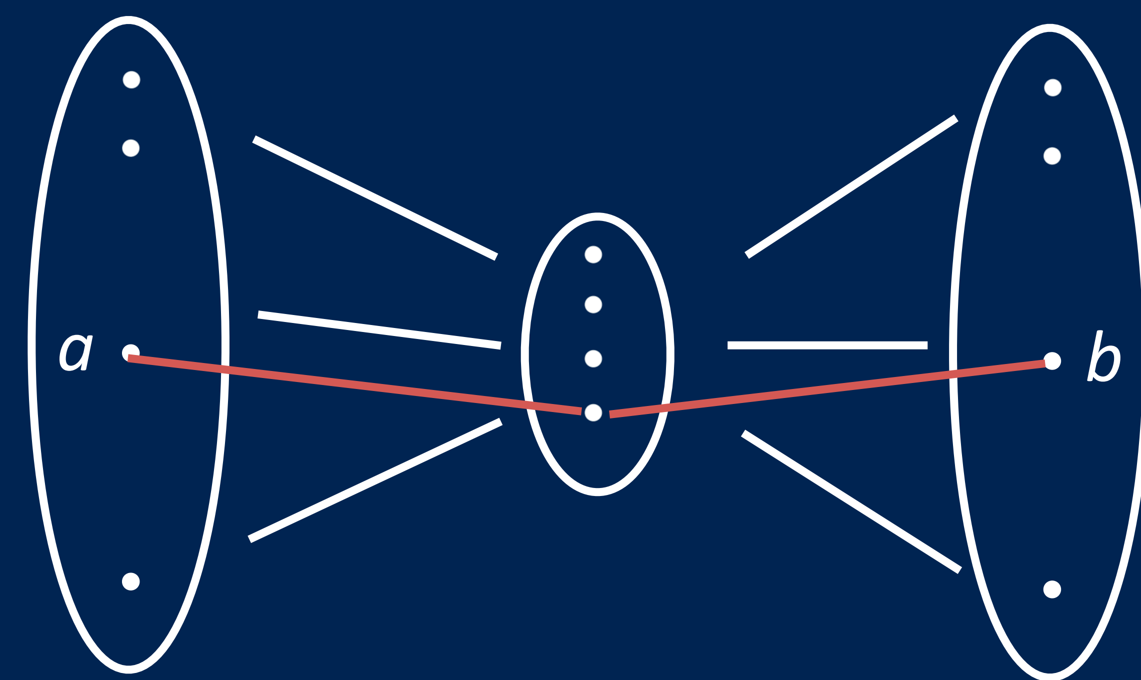
(0,0,1,…,1)	(0,0,1,…,1)
(0,1,1,…,1)	(1,0,1,…,1)
…	…
(1,0,1,…,0)	(1,1,1,…,1)

(1,0,1,…,0)

(0,0,1,…,1)

Given two lists of n vectors in $\{0,1\}^d$
is there an orthogonal pair?

(static) diameter



d(a,b) = 2 if not orth.
d(a,b)>2 if orth.

Theorem [**A.** - VW FOCS 14']:
If **dynamic #SSR** can be solved with $O(m^{1-e})$ update and query times, then **OVP** can be solved in $O(n^{2-e})$ time (and SETH is false).

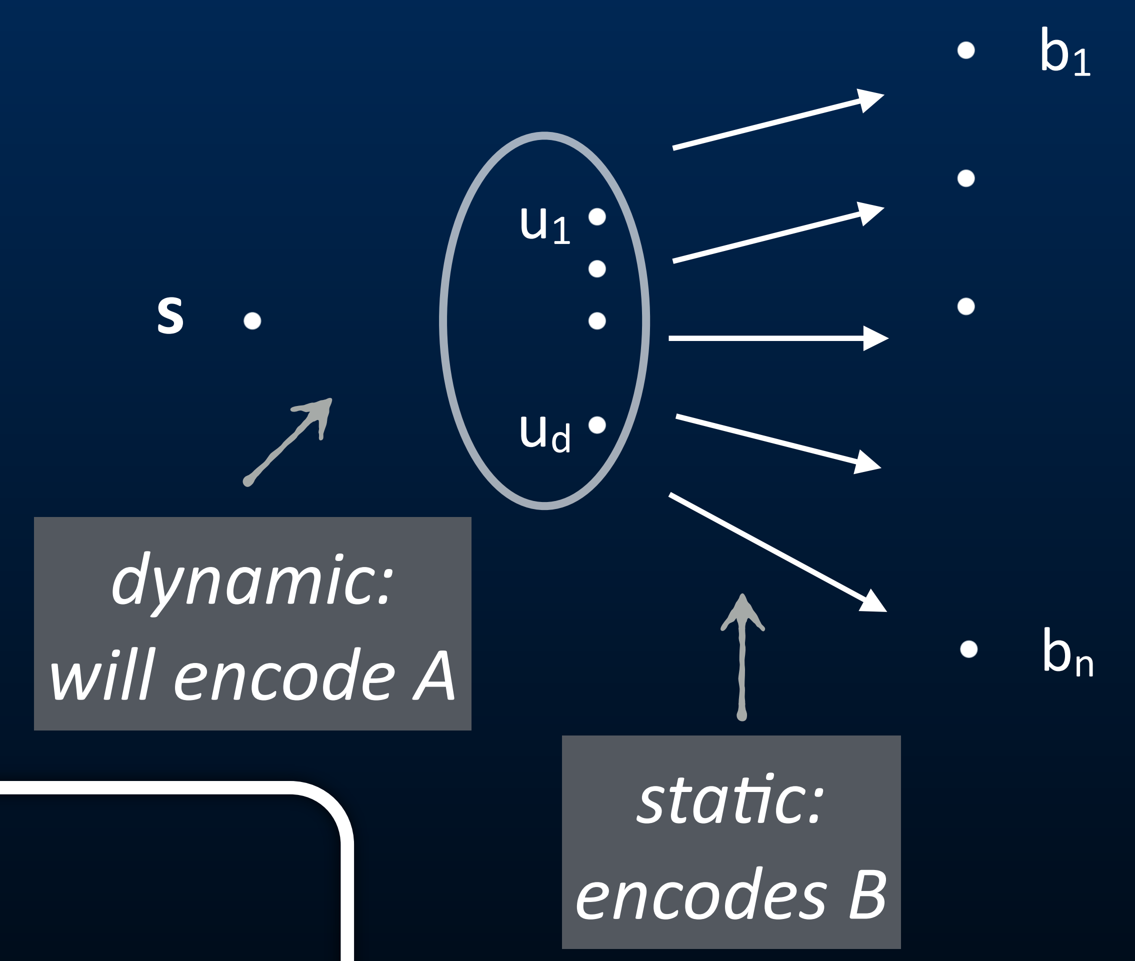
Proof:

Orthogonal Vectors → **dynamic #SSR**

a_1	(0,0,1,…,1)	(0,0,1,…,1)	b_1
	(0,1,1,…,1)	(1,0,1,…,1)	
	…	…	
a_n	(1,0,1,…,0)	(1,1,1,…,1)	b_n

(1,0,1,…,0)

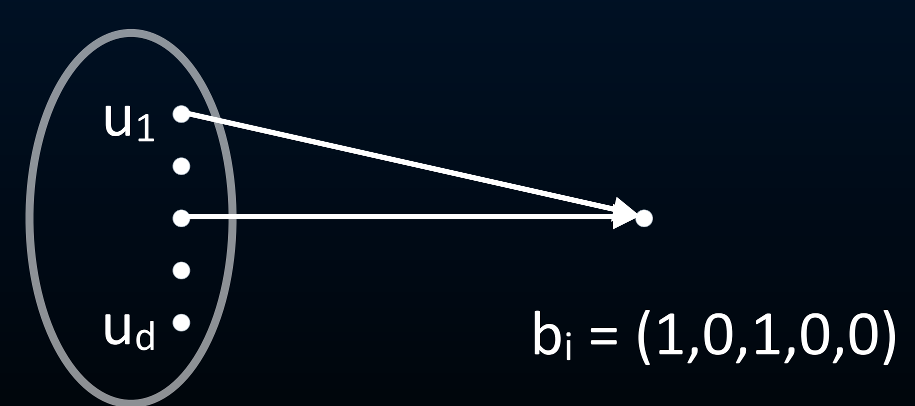
(0,0,1,…,1)



add edge

$u_j \longrightarrow b_i$

iff $b_i[j]=1$



Theorem [**A.** - VW FOCS 14']:
If **dynamic #SSR** can be solved with $O(m^{1-e})$ update and query times, then **OVP** can be solved in $O(n^{2-e})$ time (and SETH is false).

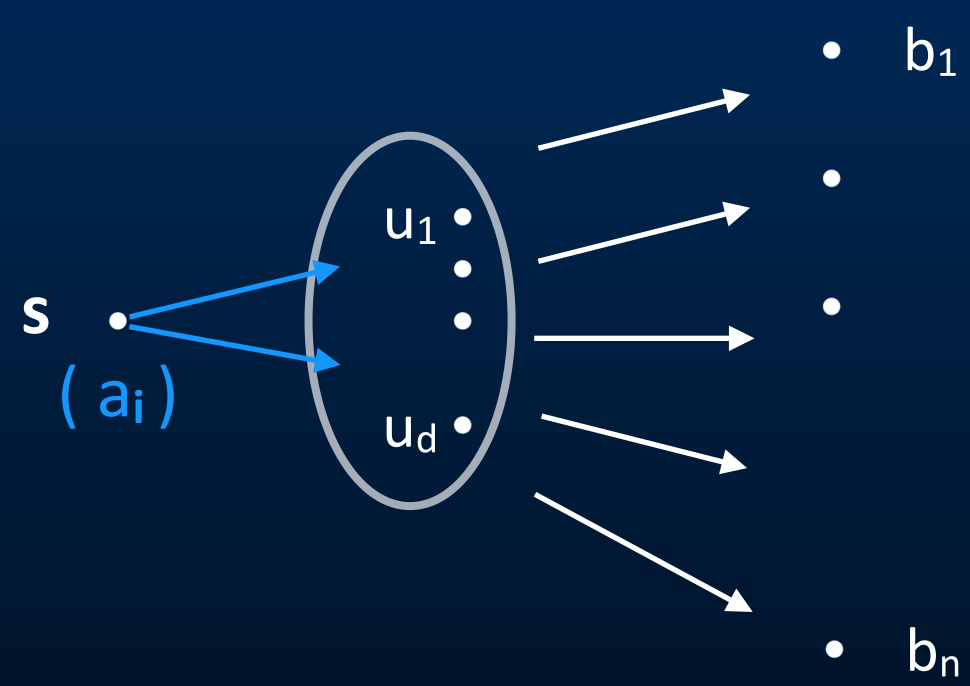
Proof:

Orthogonal Vectors → **dynamic #SSR**

a_1	(0,0,1,…,1)	(0,0,1,…,1)	b_1
	(0,1,1,…,1)	(1,0,1,…,1)	
	…	…	
a_n	(1,0,1,…,0)	(1,1,1,…,1)	b_n

(1,0,1,…,0)

(0,0,1,…,1)



add edge

$u_j \longrightarrow b_i$

iff $b_i[j]=1$

For each a_i:

1. add edges $s \longrightarrow u_j$ iff $a_i[j]=1$
2. ask #SSR(s)

Theorem [**A.** - VW FOCS 14']:
If **dynamic #SSR** can be solved with $O(m^{1-e})$ update and query times, then **OVP** can be solved in $O(n^{2-e})$ time (and SETH is false).

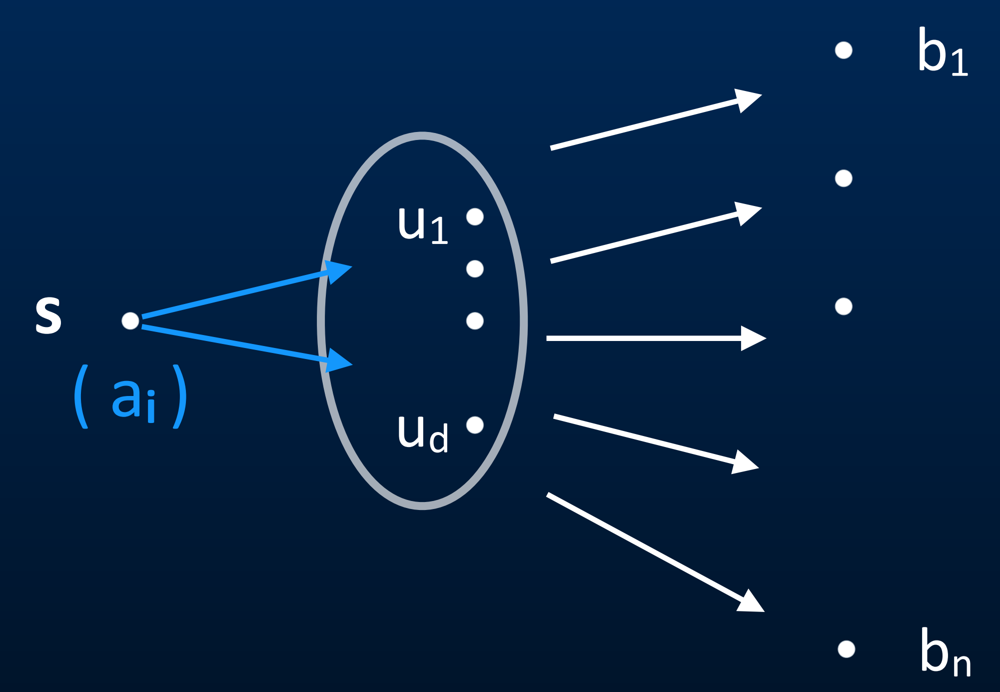
Proof:

Orthogonal Vectors → **dynamic #SSR**

a_1	(0,0,1,…,1)	(0,0,1,…,1)	b_1
	(0,1,1,…,1)	(1,0,1,…,1)	
	…	…	
a_n	(1,0,1,…,0)	(1,1,1,…,1)	b_n

(1,0,1,…,0)

(0,0,1,…,1)



add edge

$u_j \longrightarrow b_i$

iff $b_i[j]=1$

Observation:
s cannot reach ***b*** iff a_i and ***b*** are orthogonal.

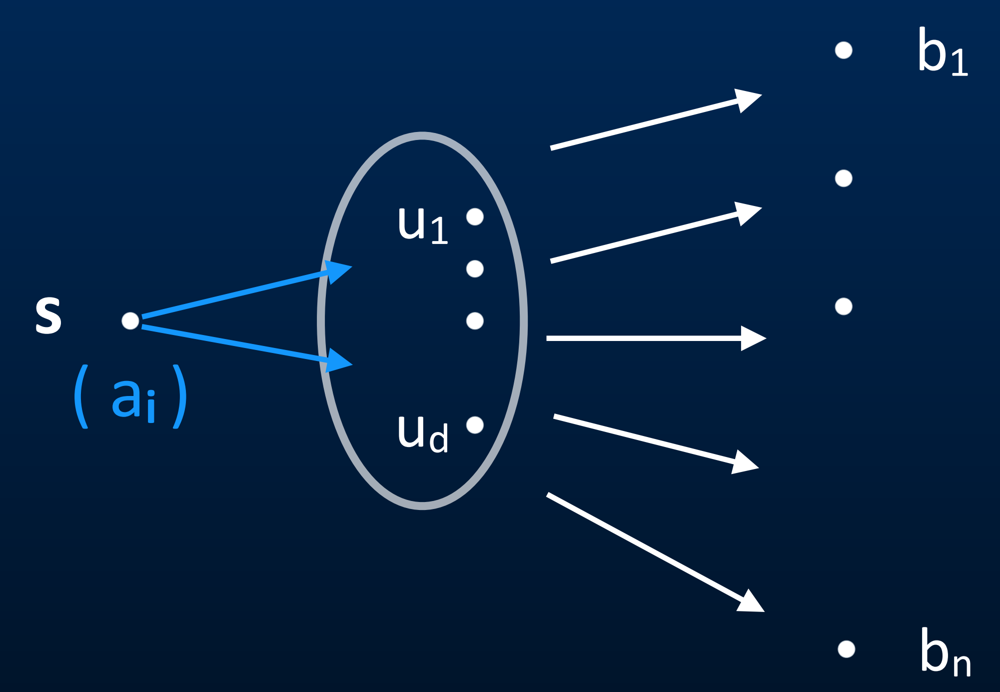
Theorem [A. - VW FOCS 14']:
If **dynamic #SSR** can be solved with $O(m^{1-e})$ update and query times,
then **OVP** can be solved in $O(n^{2-e})$ time (and SETH is false).

Proof: **Orthogonal Vectors** → **dynamic #SSR**

a_1	(0,0,1,…,1)	(0,0,1,…,1)	b_1
	(0,1,1,…,1)	(1,0,1,…,1)	
	…	…	
a_n	(1,0,1,…,0)	(1,1,1,…,1)	b_n

(1,0,1,…,0)

(0,0,1,…,1)



add edge

$u_j \longrightarrow b_i$

iff $b_i[j]=1$

For each a_i:

1. add edges $s \longrightarrow u_j$ iff $a_i[j]=1$
2. ask #SSR(s),
   if $< n + (1s \text{ in } a_i)$, output "yes".
3. remove edges and move on to next a_i

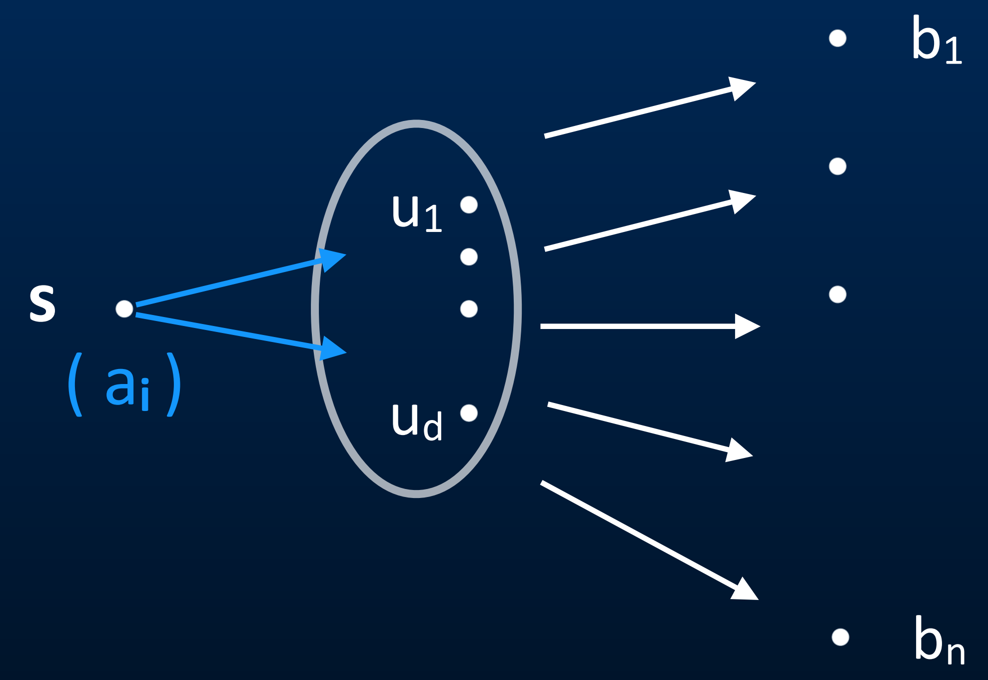
Theorem [A. - VW FOCS 14']:
If **dynamic #SSR** can be solved with $O(m^{1-e})$ update and query times, then **OVP** can be solved in $O(n^{2-e})$ time (and SETH is false).

Proof: **Orthogonal Vectors** → **dynamic #SSR**

a_1	(0,0,1,…,1)	(0,0,1,…,1)	b_1
	(0,1,1,…,1)	(1,0,1,…,1)	
	…	…	
a_n	(1,0,1,…,0)	(1,1,1,…,1)	b_n

(1,0,1,…,0)

(0,0,1,…,1)



$\boxed{O(nd) \text{ updates}, \; m = O(nd) \text{ edges}}$

~Ω(m) per update!

For each a_i:

1. add edges $s \longrightarrow u_j$ iff $a_i[j]=1$
2. ask #SSR(s), and if $< n + (1s \text{ in } a_i)$, output "yes".
3. remove edges and move on to next a_i

With additional gadgets, lower bounds for:

Strongly Connected Components

Undirected Connectivity with node updates

and more.

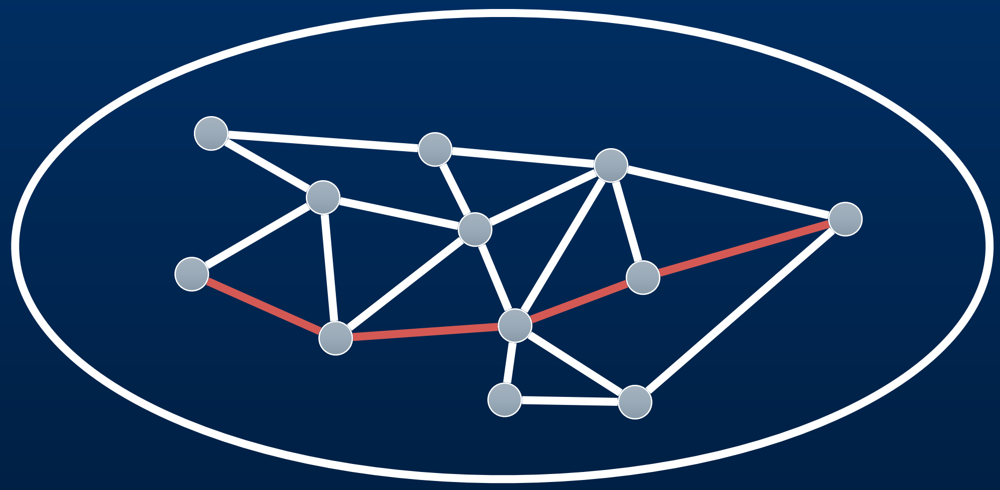
Next: even higher lower bounds!

Dynamic Diameter

Input: an undirected graph G

Updates: Add or remove edges.

Query: What is the diameter of G?



Upper bounds for dynamic All-Pairs-Shortest-Paths:
Naive: ~$O(mn)$ per update.
[Demetrescu-Italiano 03', Thorup 04']: amortized ~$O(n^2)$.

Theorem [**A** - VW FOCS 14']:
1.3-approximation for the diameter of a sparse graph under edge updates with amortized $O(m^{2-e})$ updates refutes SETH!

Theorem [**A** - VW FOCS 14']:
1.3-approximation for the diameter of a sparse graph under edge updates with amortized *$O(m^{2-e})$* updates refutes SETH!

Proof outline:

Three Orthogonal Vectors

(0,0,…,1)	(1,0,…,1)	(1,0,…,1)
(0,1,…,1)	(0,1,…,0)	(0,0,…,1)
…	…	…
(1,0,…,0)	(1,0,…,1)	(1,1,…,0)

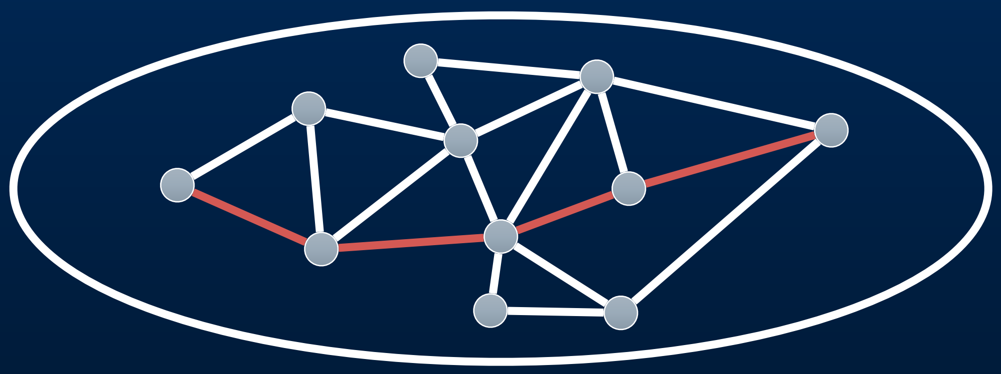
(1,0,1,…,0)

(0,1,1,…,0)

(1,1,1,…,0)

Given three lists of n vectors in $\{0,1\}^d$ is there an "orthogonal" triple?

dynamic Diameter



d=polylog(n)

Lemma: **3-OVP** in ***$\sim O(n^{3-e})$*** time refutes **SETH**

Theorem [**A** - VW FOCS 14’]:
1.3-approximation for the diameter of a sparse graph under edge updates with amortized $O(m^{2-e})$ updates refutes SETH!

Proof outline:

Three Orthogonal Vectors

(0,0,…,1)	(1,0,…,1)	(1,0,…,1)
(0,1,…,1)	(0,1,…,0)	(0,0,…,1)
…	…	…
(1,0,…,0)	(1,0,…,1)	(1,1,…,0)

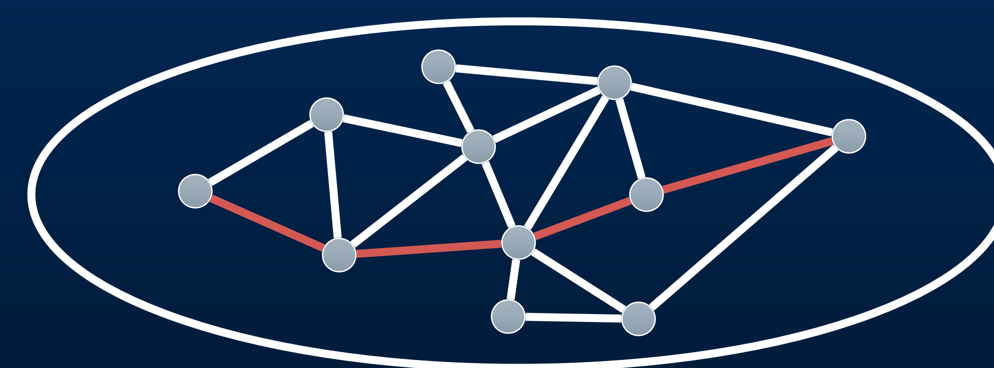
(1,0,1,…,0)

(0,1,1,…,0)

(1,1,1,…,0)

Given three lists of n vectors in $\{0,1\}^d$ is there an “orthogonal” triple?

dynamic Diameter



is the diameter 3 or more?

Graph G on ***m=O(nd)*** *nodes and edges*, **O(nd)** updates and queries

3-OVP in ***~O(n^{2.9})*** time

(refutes SETH)

O(nd) updates/queries in ***~O(n^{2.9})*** time

Amortized ***O(m^{1.9})*** update/query time

d=polylog(n), m=~O(n)

Theorem [A - VW FOCS 14']:
1.3-approximation for the diameter of a sparse graph under edge updates with amortized $O(m^{2-e})$ updates refutes SETH!

Proof:

Three Orthogonal Vectors → dynamic Diameter

A	B	C
(0,0,…,1)	(1,0,…,1)	(1,0,…,1)
(0,1,…,1)	(0,1,…,0)	(0,0,…,1)
…	…	…
(1,0,…,0)	(1,0,…,1)	(1,1,…,0)

(1,0,1,…,0)

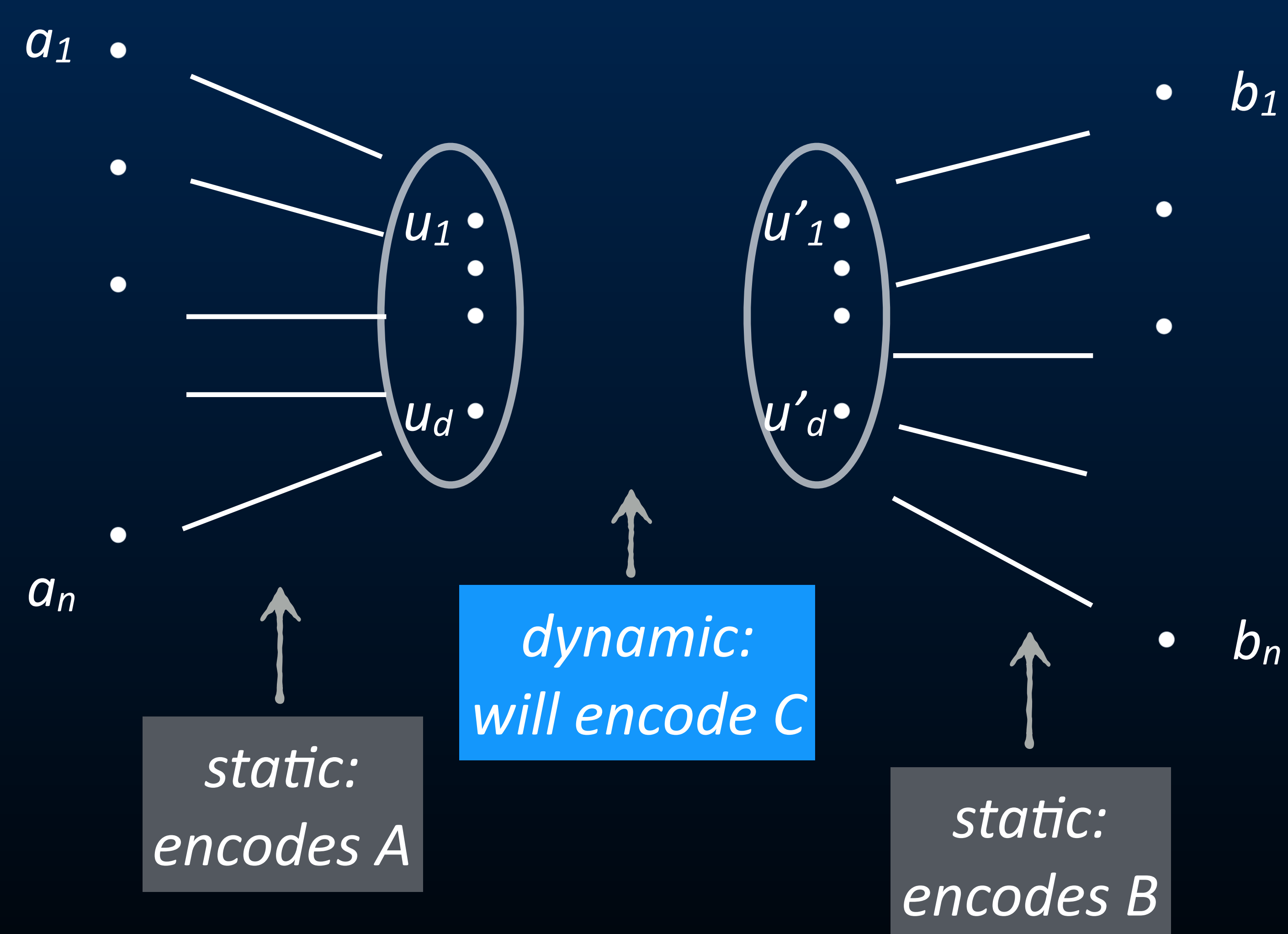
(0,1,1,…,0)

(1,1,1,…,0)

add edge

u'_j •——• b_i

iff $b_i[j]=1$



Theorem [**A** - VW FOCS 14']:

1.3-approximation for the diameter of a sparse graph under edge updates with amortized $O(m^{2-e})$ updates refutes SETH!

Proof:

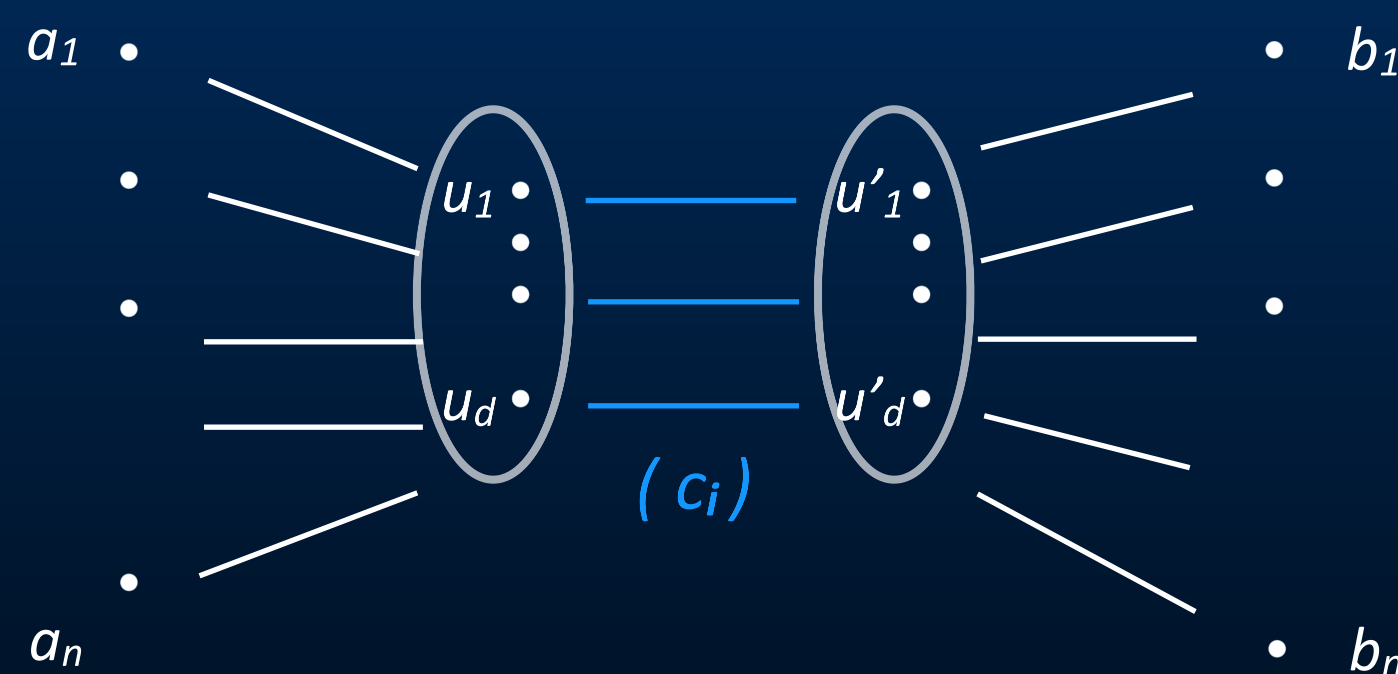
Three Orthogonal Vectors → dynamic Diameter

A	B	C
(0,0,…,1)	(1,0,…,1)	(1,0,…,1)
(0,1,…,1)	(0,1,…,0)	(0,0,…,1)
…	…	…
(1,0,…,0)	(1,0,…,1)	(1,1,…,0)

(1,0,1,…,0)

(0,1,1,…,0)

(1,1,1,…,0)



add edge

u'_j •——• b_i

iff $b_i[j]=1$

For each c_i:

1. add edges u_j •——• u'_j iff $c_i[j]=1$
2. ask Diameter query.

Theorem [A - VW FOCS 14']:
1.3-approximation for the diameter of a sparse graph under edge updates with amortized $O(m^{2-e})$ updates refutes SETH!

Proof:

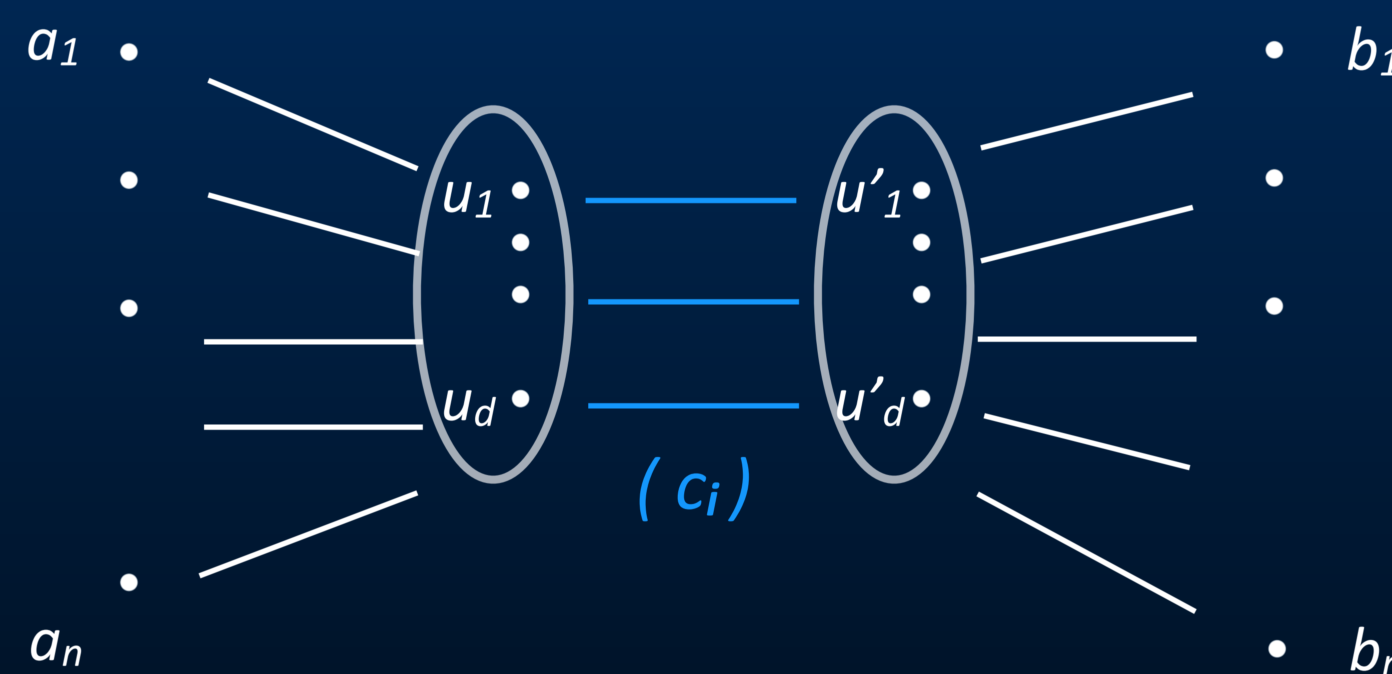
Three Orthogonal Vectors → **dynamic Diameter**

A	B	C
(0,0,…,1)	(1,0,…,1)	(1,0,…,1)
(0,1,…,1)	(0,1,…,0)	(0,0,…,1)
…	…	…
(1,0,…,0)	(1,0,…,1)	(1,1,…,0)

(1,0,1,…,0)

(0,1,1,…,0)

(1,1,1,…,0)



add edge

u'_j ●——● b_i

iff $b_i[j]=1$

Observation:
The distance from *a* to *b* is more than ***3*** iff *a,b,*c_i are an orthogonal triple.

(no coordinate with all three 1's)

Theorem [**A** - VW FOCS 14']:
1.3-approximation for the diameter of a sparse graph under edge updates with amortized $O(m^{2-e})$ updates refutes SETH!

Proof:

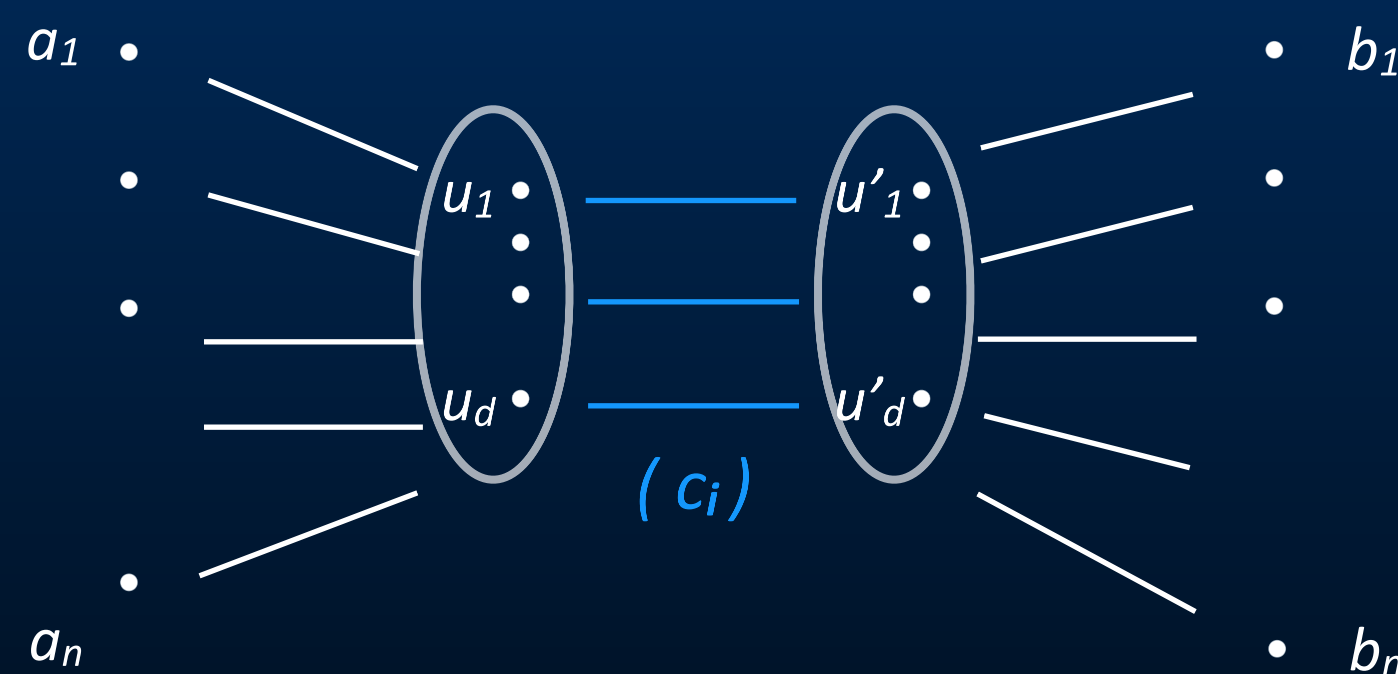
Three Orthogonal Vectors → dynamic Diameter

A	B	C
(0,0,…,1)	(1,0,…,1)	(1,0,…,1)
(0,1,…,1)	(0,1,…,0)	(0,0,…,1)
…	…	…
(1,0,…,0)	(1,0,…,1)	(1,1,…,0)

(1,0,1,…,0)

(0,1,1,…,0)

(1,1,1,…,0)



$O(nd)$ updates,
$m = O(nd)$ edges

~$\Omega(n^2)$ per update!

For each c_i:

1. add edges u_j —— u'_j iff $c_i[j]=1$
2. Query. If Diameter > 3, output "yes".
3. remove edges and move on to next c_i

Conclusions:

Very high lower bounds for fundamental problems

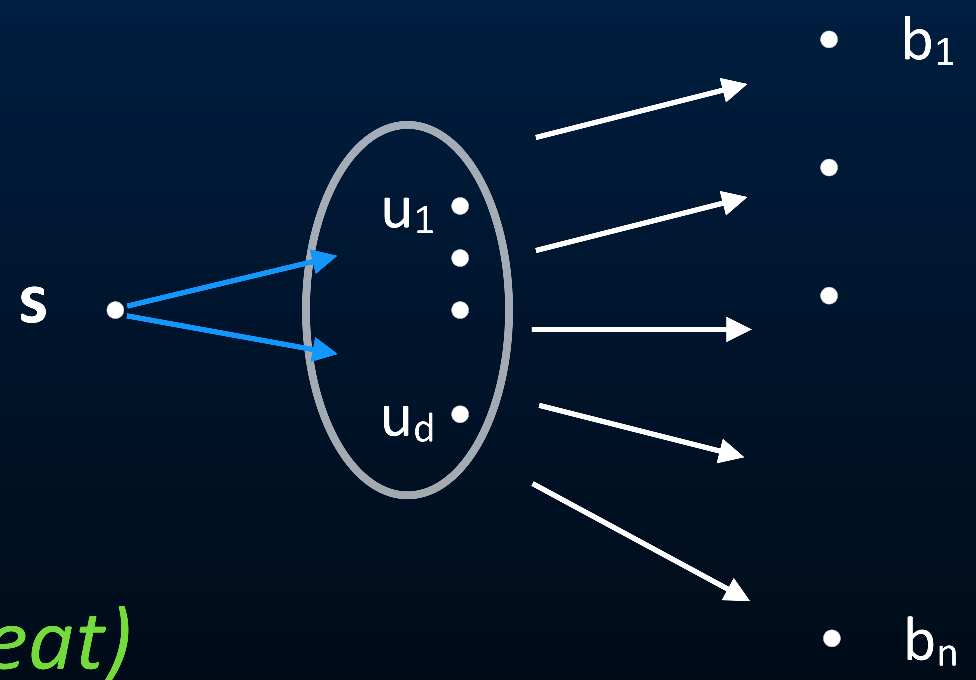
After identifying the conjecture, the proofs are very simple!

Many interesting open questions…

Open: Lower bound for decremental reachability

[Henzinger-Krinninger-Nanongkai ICALP 15']:
$O(mn^{9/10})$ total update time.

Lower bound for worst case updates:



add, query, remove (and repeat)

add, query, "backtrack" (and repeat)

Barrier for better lower bounds: the incremental case is O(m)

<u>Open</u>: Explain the gaps between randomized and deterministic upper bounds.

“deterministic conjectures” might be needed

<u>Tomorrow:</u>

Lower bounds with much better guarantees!

[A-VW-Yu STOC 15']

Even if ***at least one*** of APSP, 3SUM, SETH is true,
then Single Source Reachability requires linear updates!

Thanks for listening!

More reductions after coffee!