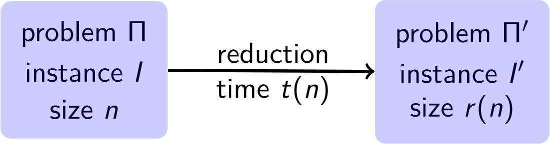
Fine-Grained Complexity in P

Édouard Bonnet

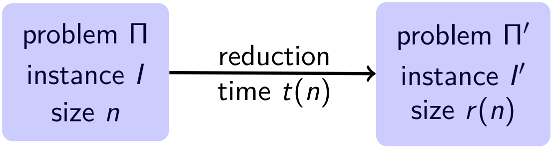
LIP, ENS Lyon

October 2nd, 2018, GT CoA, Paris

Reductions


problem Π
instance I
size n
reduction
time $t(n)$
problem Π′
instance I'
size $r(n)$

Reductions



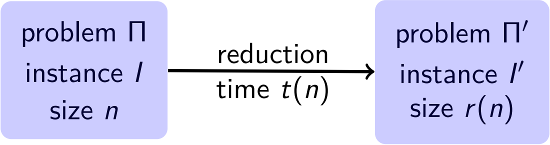
Two interpretations

Positive: we can solve Π in $t(n) + f(r(n))$ with f is the time for Π'

Negative: **Π' cannot be solved in $\mathbf{f(n)}$**
since we know/assume that Π is not solvable in $t(n) + f(r(n))$

Reductions and fine-grained complexity

Complexity with the classes TIME($f(n)$) and reference problems



Two interpretations

Positive: we can solve Π in $t(n) + f(r(n))$ with f is the time for Π′

Negative: **Π′ cannot be solved in f(n)**
since we know/assume that Π is not solvable in $t(n) + f(r(n))$

The three main hypotheses

Strong Exponential Time Hypothesis (SETH): $\forall \varepsilon > 0$, $\exists k$ s.t. k-SAT is not in time $2^{(1-\varepsilon)n}$.

The three main hypotheses

Strong Exponential Time Hypothesis (SETH): $\forall \varepsilon > 0$, $\exists k$ s.t. k-SAT is not in time $2^{(1-\varepsilon)n}$ by a classical (randomized) algorithm.

The three main hypotheses

Strong Exponential Time Hypothesis (SETH): $\forall \varepsilon > 0$, $\exists k$ s.t. k-SAT is not in time $2^{(1-\varepsilon)n}$ by a classical (randomized) algorithm.

3-SUM Hypothesis: Finding x, y, z such that $x + y + z = 0$ in a list of n integers of $[-n^4, n^4]$ is not in time $O(n^{2-\varepsilon})$ for any $\varepsilon > 0$.

The three main hypotheses

Strong Exponential Time Hypothesis (SETH): $\forall \varepsilon > 0$, $\exists k$ s.t. k-SAT is not in time $2^{(1-\varepsilon)n}$ by a classical (randomized) algorithm.

3-SUM Hypothesis: Finding x, y, z such that $x + y + z = 0$ in a list of n integers of $[-n^4, n^4]$ is not in time $O(n^{2-\varepsilon})$ for any $\varepsilon > 0$.

All-Pairs Shortest-Path (APSP) Hypothesis: $\exists c$, APSP with edge weights in $[-n^c, n^c]$ is not solvable in time $O(n^{3-\varepsilon})$ for $\varepsilon > 0$.

The three main hypotheses

SETH: $\textsc{Sat}$ is not solvable in 1.99^n.

- k-$\textsc{Sat}$ is solvable in $2^{(1-\Theta(\frac{1}{k}))n}$
- $\textsc{Sat}$ is solvable $2^{(1-\Theta(\frac{1}{\log m/n}))n}$

3-SUM Hypothesis: 3-SUM is not solvable in $n^{1.99}$

- Solvable in $n^2 \frac{(\log \log n)^{O(1)}}{\log^2 n}$ even with real inputs
- Linear decision tree with depth $O(n\ log^2 n)$

APSP Hypothesis: APSP is not solvable in $n^{2.99}$

- solvable in cubic time by Floyd-Warshall algorithm
- improved to $n^3/2^{O(\sqrt{\log n})}$

SETH

In 1999, Impagliazzo and Paturi introduce ETH[1] and mention a stronger version of it in their conclusion

$$\text{SETH} \Rightarrow \text{ETH} \Rightarrow \text{P} \neq \text{NP}$$

- ETH and SETH are then mainly used for NP-hard problems

[1] $\exists \delta > 0$, 3-SAT cannot be solved in $2^{\delta n}$

SETH

In 1999, Impagliazzo and Paturi introduce ETH[1] and mention a stronger version of it in their conclusion

$$\text{SETH} \Rightarrow \text{ETH} \Rightarrow \text{P} \neq \text{NP}$$

- ETH and SETH are then mainly used for NP-hard problems
- In 2005, SETH is used for the first time for a problem in P

ORTHOGONAL VECTORS,

[1] $\exists\delta > 0$, 3-SAT cannot be solved in $2^{\delta n}$

SETH

In 1999, Impagliazzo and Paturi introduce ETH[1] and mention a stronger version of it in their conclusion

$$\text{SETH} \Rightarrow \text{ETH} \Rightarrow \text{P} \neq \text{NP}$$

- ETH and SETH are then mainly used for NP-hard problems
- In 2005, SETH is used for the first time for a problem in P
- 2014-, dozens of papers show SETH-hardness of problems in P

Orthogonal Vectors, Diameter, Fréchet Distance, Edit Distance, Longest Common Subsequence, Furthest Pair, dynamic problems, problems from Machine Learning, Model Checking, Language Theory etc.

[1] $\exists \delta > 0$, 3-Sat cannot be solved in $2^{\delta n}$

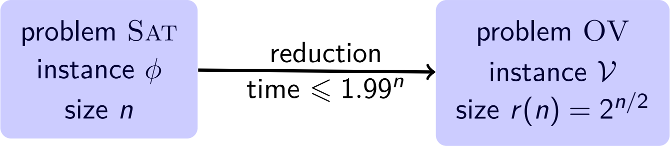
Psychological barrier?

If we treat k-$\textsc{Sat} \to \mathrm{OV}$ as an outlier,
why 15 years between defining SETH and using it in P?

Psychological barrier?

A: If you know too much (P, NP, NP-completeness), you might disregard a reduction from a hard problem to an easy one

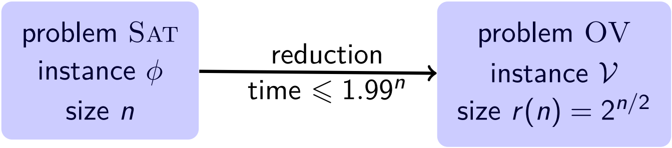
B: If you know less (summing and composing functions)



Psychological barrier?

A: If you know too much (P, NP, NP-completeness), you might disregard a reduction from a hard problem to an easy one

B: If you know less (summing and composing functions)



B: "Both the time and the blow-up can be exponential, great."

Sat → Orthogonal Vectors

arbitrary equipartition of X: $x_1, x_2, \ldots, x_{\frac{n}{2}}, x_{\frac{n}{2}+1}, x_{\frac{n}{2}+2}, \ldots, x_n$

Find an assignment

- A of the red variables and
- B of the blue variables

such that all the clauses are satisfied by A or by B

Sat → Orthogonal Vectors

arbitrary equipartition of X: $x_1, x_2, \ldots, x_{\frac{n}{2}}$, $x_{\frac{n}{2}+1}, x_{\frac{n}{2}+2}, \ldots, x_n$

Find an assignment

- A of the red variables and
- B of the blue variables

such that all the clauses are satisfied by A or by B

	R	B	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8
A_1										
A_2										
A_3										
A_4										
B_1										
B_2										
B_3										
B_4										

Sat → Orthogonal Vectors

arbitrary equipartition of X: $x_1, x_2, \ldots, x_{\frac{n}{2}}, x_{\frac{n}{2}+1}, x_{\frac{n}{2}+2}, \ldots, x_n$

Find an assignment

- A of the red variables and
- B of the blue variables

such that all the clauses are satisfied by A or by B

	R	B	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8
A_1	1	0								
A_2										
A_3										
A_4										
B_1										
B_2										
B_3										
B_4										

A_1 assigns red variables

Sat → Orthogonal Vectors

arbitrary equipartition of X: $x_1, x_2, \ldots, x_{\frac{n}{2}}, x_{\frac{n}{2}+1}, x_{\frac{n}{2}+2}, \ldots, x_n$

Find an assignment

- A of the red variables and
- B of the blue variables

such that all the clauses are satisfied by A or by B

	R	B	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8
A_1	1	0	1							
A_2										
A_3										
A_4										
B_1										
B_2										
B_3										
B_4										

A_1 does *not* satisfy C_1

Sat → Orthogonal Vectors

arbitrary equipartition of X: $x_1, x_2, \ldots, x_{\frac{n}{2}}, x_{\frac{n}{2}+1}, x_{\frac{n}{2}+2}, \ldots, x_n$

Find an assignment

- A of the red variables and
- B of the blue variables

such that all the clauses are satisfied by A or by B

	R	B	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8
A_1	1	0	1	0						
A_2										
A_3										
A_4										
B_1										
B_2										
B_3										
B_4										

A_1 satisfies C_2

Sat → Orthogonal Vectors

arbitrary equipartition of X: $x_1, x_2, \ldots, x_{\frac{n}{2}}$, $x_{\frac{n}{2}+1}, x_{\frac{n}{2}+2}, \ldots, x_n$

Find an assignment

- A of the red variables and
- B of the blue variables

such that all the clauses are satisfied by A or by B

	R	B	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8
A_1	1	0	1	0	0	1	0	0	1	0
A_2										
A_3										
A_4										
B_1										
B_2										
B_3										
B_4										

first vector $(1, 0, 1, 0, 0, 1, 1, 0, 1, 0)$

Sat → Orthogonal Vectors

arbitrary equipartition of X: $x_1, x_2, \ldots, x_{\frac{n}{2}}$, $x_{\frac{n}{2}+1}, x_{\frac{n}{2}+2}, \ldots, x_n$

Find an assignment

- A of the red variables and
- B of the blue variables

such that all the clauses are satisfied by A or by B

	R	B	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8
A_1	1	0	1	0	0	1	0	0	1	0
A_2	1	0	0	0	0	1	1	1	0	1
A_3	1	0	0	1	0	1	0	0	1	1
A_4	1	0	0	0	1	1	0	1	1	1
B_1	0	1	1	1	0	0	1	1	1	0
B_2	0	1	0	1	0	1	0	1	0	0
B_3	0	1	1	1	1	1	0	0	0	1
B_4	0	1	0	1	0	0	1	0	0	1

SAT → ORTHOGONAL VECTORS

arbitrary equipartition of X: $x_1, x_2, \dots, x_{\frac{n}{2}}$, $x_{\frac{n}{2}+1}, x_{\frac{n}{2}+2}, \dots, x_n$

Find an assignment

- A of the red variables and
- B of the blue variables

such that all the clauses are satisfied by A or by B

	R	B	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8
A_1	1	**0**	1	**0**	**0**	1	**0**	**0**	1	**0**
A_2	1	0	0	0	0	1	1	1	0	1
A_3	1	0	0	1	0	1	0	0	1	1
A_4	1	0	0	0	1	1	0	1	1	1
B_1	0	1	1	1	0	0	1	1	1	0
B_2	0	1	0	1	0	1	0	1	0	0
B_3	0	1	1	1	1	1	0	0	0	1
B_4	**0**	1	**0**	1	**0**	**0**	1	**0**	**0**	1

Consequence for OV under the SETH

From a SAT-instance on n variables and m clauses, we created $N := 2^{\frac{n}{2}+1}$ vectors in dimension $d := m + 2$

Consequence for OV under the SETH

From a $\textsc{Sat}$-instance on n variables and m clauses, we created $N := 2^{\frac{n}{2}+1}$ vectors in dimension $d := m + 2$

An algorithm solving OV in time $2^{o(d)} N^{2-\varepsilon}$
would solve $\textsc{Sat}$ in $2^{o(m)} 2^{n(1-\varepsilon/2)} \to$ breaking SETH

Sharp contrast with the simple algorithms in $O(N^2 d)$ and $O(2^d N)$

DIAMETER

diam(G) = largest distance between a pair of vertices of G

- In weighted graphs, nothing known better than APSP
- In unweigthed graphs, solvable in $\tilde{O}(n^{\omega})$
- In unweighted sparse ($m = \Theta(n)$) graphs, solvable in $O(n^2)$
- $\frac{3}{2}$-approximable in $\tilde{O}(m^{1.5})$
- In sparse graphs, $\frac{3}{2}$-approximable in $\tilde{O}(n^{1.5})$

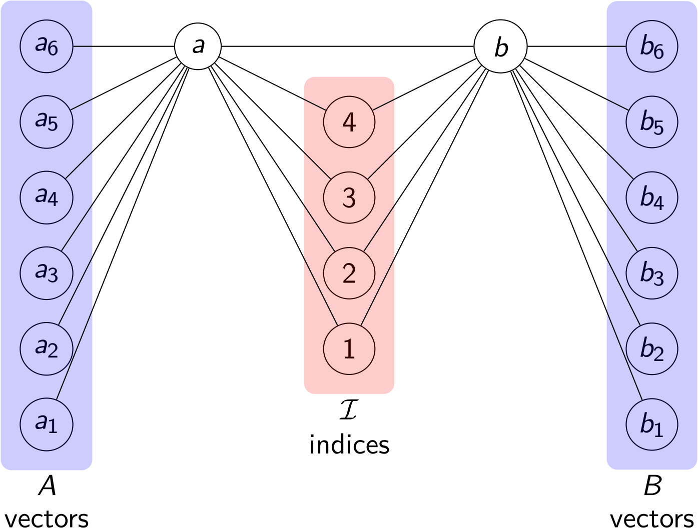
Diameter

diam(G) = largest distance between a pair of vertices of G

- In weighted graphs, nothing known better than APSP
- In unweigthed graphs, solvable in $\tilde{O}(n^{\omega})$
- **In unweighted sparse graphs, solvable in $\mathbf{O(n^2)}$**
- $\frac{3}{2}$-approximable in $\tilde{O}(m^{1.5})$
- **In sparse graphs, $\frac{3}{2}$-approximable in $\tilde{\mathbf{O}}(\mathbf{n^{1.5}})$**

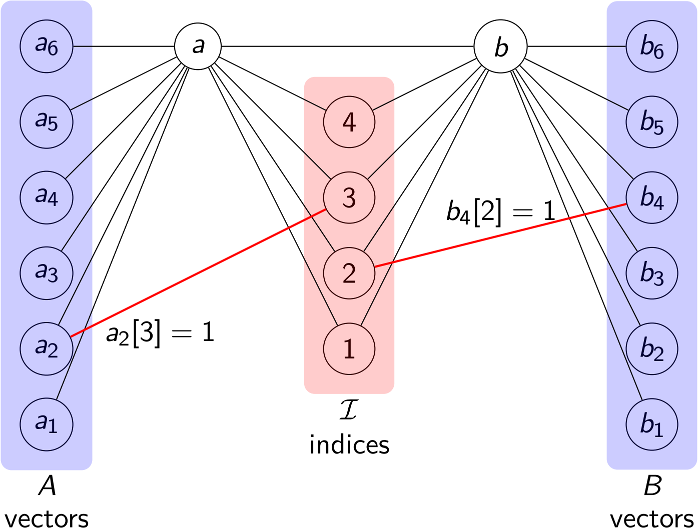
Linear reduction from Orthogonal Vectors:
no $\mathbf{n^{1.99}}$ algorithm even to $(\frac{3}{2} - \varepsilon)$-approximate Diameter
on unweighted sparse instances, assuming the SETH.

Orthogonal Vectors → Diameter



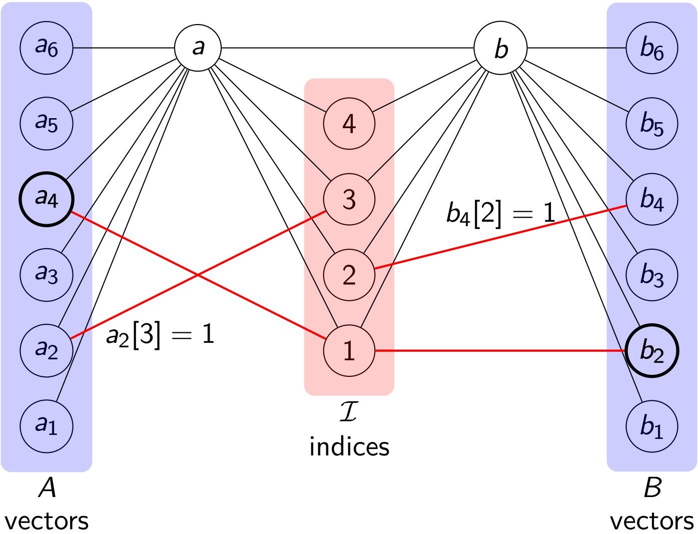
So far, all the pairs but of $A \times B$ are at distance $\leqslant 2$

Orthogonal Vectors → Diameter



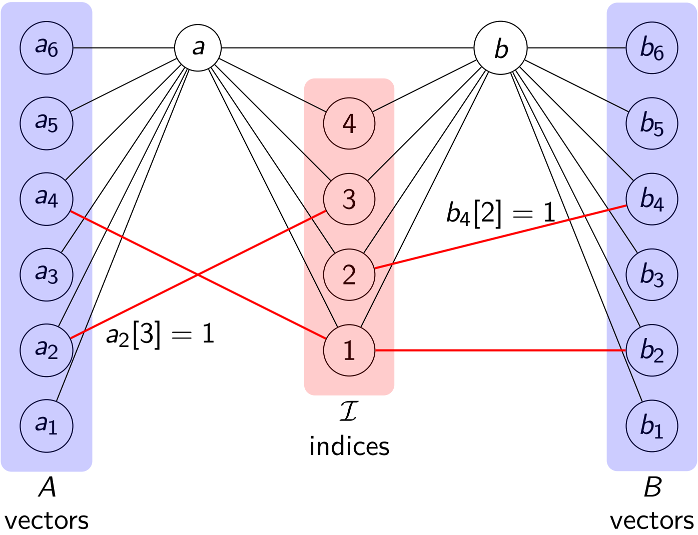
we put an edge between vector v and index i iff $v[i] = 1$

Orthogonal Vectors → Diameter



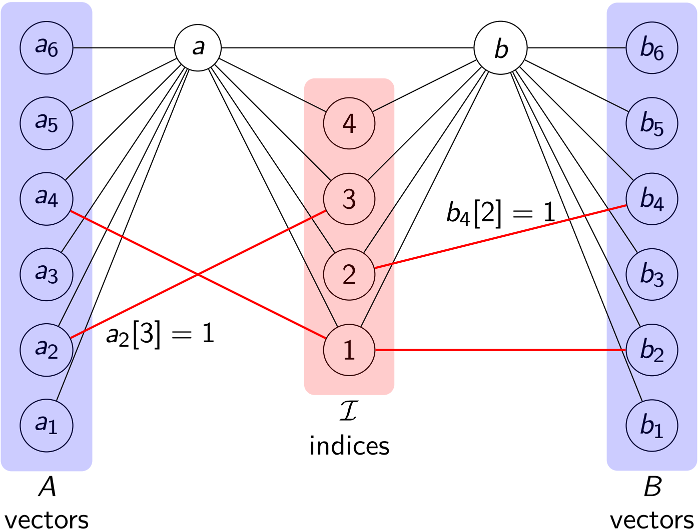
A pair (a4,b2) is at distance 2 $\Leftrightarrow \langle a_4, b_2 \rangle \neq 0$

Orthogonal Vectors → Diameter



$\mathrm{diam}(G) = 3 \Leftrightarrow \exists (a_i, b_j)$ at distance 3 $\Leftrightarrow$ orthogonal pair

Orthogonal Vectors → Diameter



If no orthogonal pair, $\mathrm{diam}(G) = 2$

3-SUM Hardness

Introduced in 1995 by Gajentaan and Overmars to explain why **some geometric problems require quadratic time**

- are there three aligned points?
- are there three lines meeting at a point? (same by duality)
- is there a hole in a union of triangles?
- computing the area of a union of triangles
- is a rectangle covered by a set of infinite strips?
- Line separator of a non-intersecting axis-parallel segments?
- motion planning problems
- visibility problems

3-SUM Hardness

Introduced in 1995 by Gajentaan and Overmars to explain why **some geometric problems require quadratic time**

- are there three aligned points?
- are there three lines meeting at a point? (same by duality)
- is there a hole in a union of triangles?
- computing the area of a union of triangles
- is a rectangle covered by a set of infinite strips?
- Line separator of a non-intersecting axis-parallel segments?
- motion planning problems
- visibility problems

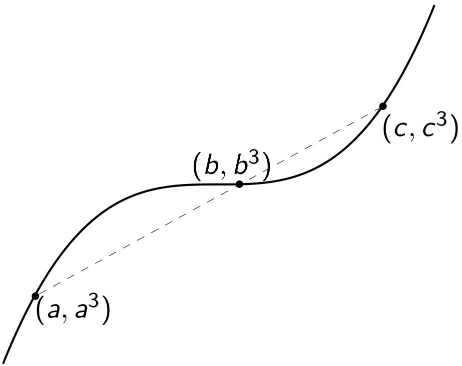
the quadratic algorithm is not easy

3-SUM → 3 Collinear Points

Each integer x is mapped to the point (x, x^3)

3-SUM → 3 Collinear Points

Each integer x is mapped to the point (x, x^3)

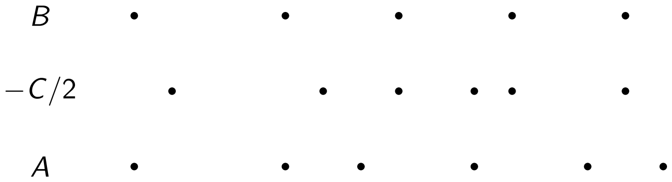


If a, b, c are pairwise distinct

$a + b + c = 0 \Leftrightarrow (a, a^3), (b, b^3), (c, c^3)$ are aligned

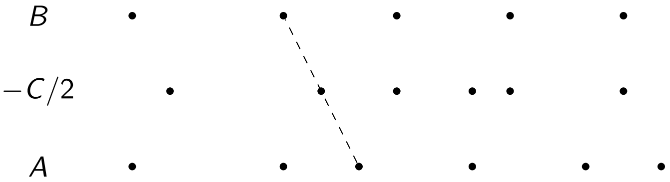
GeomBase

Convenient 3-points-on-a-line 3-SUM-hard variant



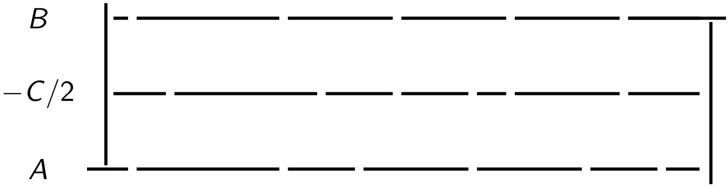
GeomBase

Convenient 3-points-on-a-line 3-SUM-hard variant



GeomBase → Segments Separation

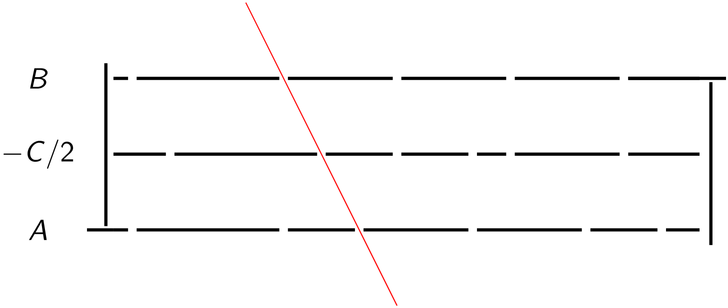
Convenient 3-points-on-a-line 3-SUM-hard variant



Points become tiny holes between horizontal segments

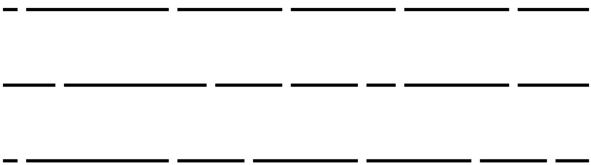
GeomBase → Segments Separation

Convenient 3-points-on-a-line 3-SUM-hard variant



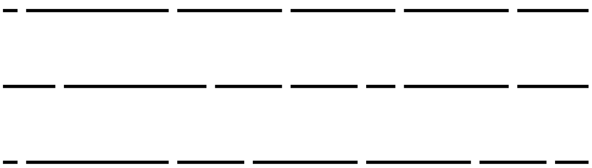
Points become tiny holes between horizontal segments

Segments Separation → Covered Rectangle



Rotation + Duality! Points ↔ Lines & Vertical Segments ↔ Strips

Segments Separation → Covered Rectangle



Rotation + Duality! Points ↔ Lines & Vertical Segments ↔ Strips

Point not on any strip ⇔ Line separating the segments

APSP Hardness

This hypothesis has emerged more recently, introduced by Ryan Williams and Virginia Vassilevska Williams in 2010

APSP is in time $n^{3-\varepsilon}$ iff so is one of:

- finding a triangle with negative weight
- finding the radius of a weighted graph
- does a given matrix represent a metric?
- finding a shortest cycle in a graph with non-negative weights
- $(min, +)$ matrix multiplication
- computing the Wiener index of a weighted graph
- betweenness centrality of a vertex in a weighted graph

APSP Hardness

This hypothesis has emerged more recently, introduced by Ryan Williams and Virginia Vassilevska Williams in 2010

APSP is in time $n^{3-\varepsilon}$ iff so is one of:

- finding a triangle with negative weight
- finding the radius of a weighted graph
- does a given matrix represent a metric?
- finding a shortest cycle in a graph with non-negative weights
- $(min, +)$ matrix multiplication
- computing the Wiener index of a weighted graph
- betweenness centrality of a vertex in a weighted graph

The hypothesis of weighted problems

Let us recall that unweighted APSP can be solved in n^{ω}

APSP Hardness

This hypothesis has emerged more recently, introduced by Ryan Williams and Virginia Vassilevska Williams in 2010

APSP is in time $n^{3-\varepsilon}$ iff so is one of:

- finding a triangle with negative weight
- finding the radius of a weighted graph
- does a given matrix represent a metric?
- finding a shortest cycle in a graph with non-negative weights
- $(min, +)$ matrix multiplication
- computing the Wiener index of a weighted graph
- betweenness centrality of a vertex in a weighted graph

The hypothesis of weighted problems

Let us recall that unweighted APSP can be solved in n^{ω}

Tree Edit Distance in truly subcubic time is APSP-hard.

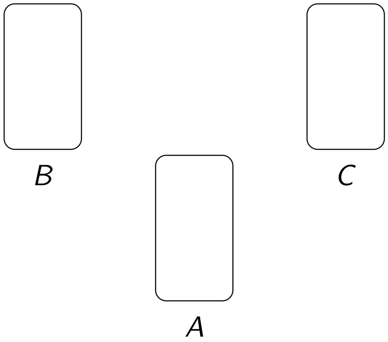
$\mathrm{APNT} \to$ Negative Triangle: a Turing reduction

All-Pairs Negative Triangle (APNT):
Given a tripartite graph on (A, B, C), is there, for every pair $b \in B, c \in C$, a vertex $a \in A$ such that abc is a negative triangle?

One can show that APSP and APNT are equivalent

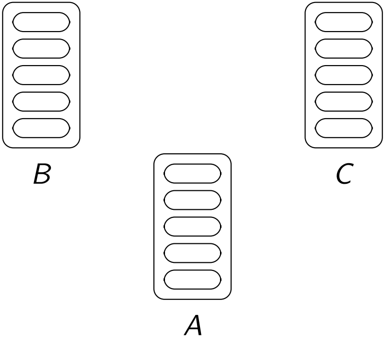
APNT → Negative Triangle: a Turing reduction

All-Pairs Negative Triangle (APNT):
Given a tripartite graph on (A, B, C), is there, for every pair $b \in B, c \in C$, a vertex $a \in A$ such that abc is a negative triangle?



APNT → Negative Triangle: a Turing reduction

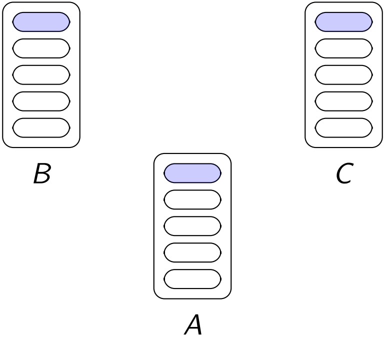
All-Pairs Negative Triangle (APNT):
Given a tripartite graph on (A, B, C), is there, for every pair $b \in B, c \in C$, a vertex $a \in A$ such that abc is a negative triangle?



Arbitrary partitions into $t = n^{2/3}$ groups of size $n/t = n^{1/3}$

APNT → Negative Triangle: a Turing reduction

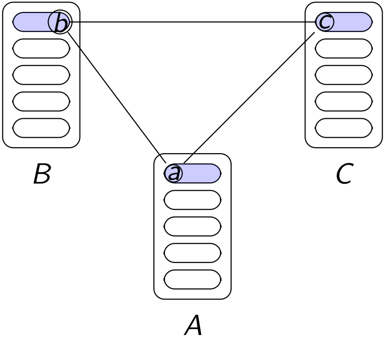
All-Pairs Negative Triangle (APNT):
Given a tripartite graph on (A, B, C), is there, for every pair $b \in B, c \in C$, a vertex $a \in A$ such that abc is a negative triangle?



For each triple of classes, call Negative Triangle

APNT → Negative Triangle: a Turing reduction

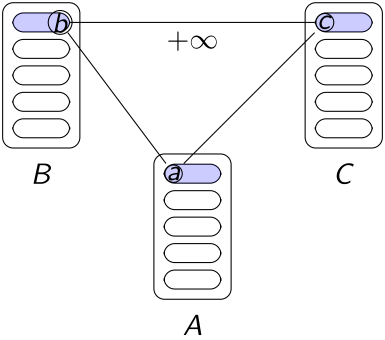
All-Pairs Negative Triangle (APNT):
Given a tripartite graph on (A, B, C), is there, for every pair $b \in B, c \in C$, a vertex $a \in A$ such that abc is a negative triangle?



Write down that the pair bc is satisfied by a and remove bc

APNT → Negative Triangle: a Turing reduction

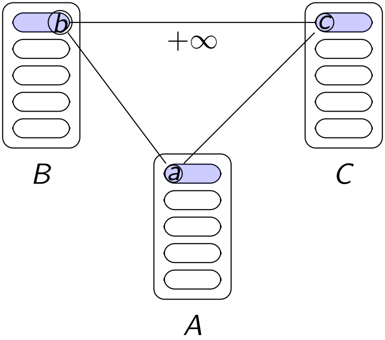
All-Pairs Negative Triangle (APNT):
Given a tripartite graph on (A, B, C), is there, for every pair $b \in B, c \in C$, a vertex $a \in A$ such that abc is a negative triangle?



Write down that the pair bc is satisfied by a and remove bc

APNT → Negative Triangle: a Turing reduction

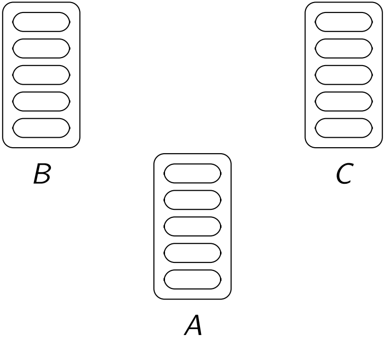
All-Pairs Negative Triangle (APNT):
Given a tripartite graph on (A, B, C), is there, for every pair $b \in B, c \in C$, a vertex $a \in A$ such that abc is a negative triangle?



Continue with the same triple of classes while possible

APNT → Negative Triangle: a Turing reduction

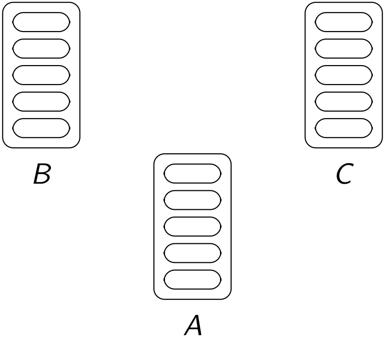
All-Pairs Negative Triangle (APNT):
Given a tripartite graph on (A, B, C), is there, for every pair $b \in B, c \in C$, a vertex $a \in A$ such that abc is a negative triangle?



Report if all the pairs of $B \times C$ were satisfied

APNT → NEGATIVE TRIANGLE: a Turing reduction

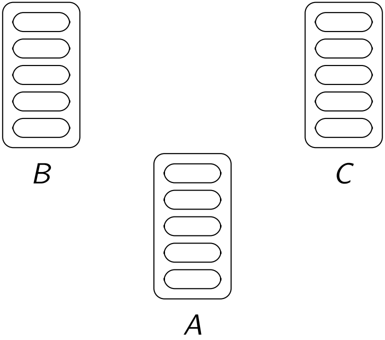
All-Pairs Negative Triangle (APNT):
Given a tripartite graph on (A, B, C), is there, for every pair $b \in B, c \in C$, a vertex $a \in A$ such that abc is a negative triangle?



Number of calls to NEGATIVE TRIANGLE: $\leqslant n^2 + t^3 = O(n^2)$

APNT → Negative Triangle: a Turing reduction

All-Pairs Negative Triangle (APNT):
Given a tripartite graph on (A, B, C), is there, for every pair $b \in B, c \in C$, a vertex $a \in A$ such that abc is a negative triangle?



Size of the subinstances: $3n/t = O(n^{1/3})$

SETH, 3-SUMH, APSPH

Are they pairwise incomparable?

SETH, 3-SUMH, APSPH

Are they pairwise incomparable?

Can we only use SETH by designing fine-grained reductions

- from k-SAT to 3-SUM
- from k-SAT to APSP?

Non-deterministic Strong Exponential Time Hypothesis

Non-deterministic Strong Exponential Time Hypothesis

k-TAUT: Are all the assignments of a k-DNF formula satisfying?

NSETH: $\forall \varepsilon > 0$, $\exists k$, k-TAUT is not in NTIME$(2^{(1-\varepsilon)n})$.

Non-deterministic Strong Exponential Time Hypothesis

k-TAUT: Are all the assignments of a k-DNF formula satisfying?

NSETH: $\forall \varepsilon > 0$, $\exists k$, k-TAUT is not in NTIME($2^{(1-\varepsilon)n}$).

- $\neg$NSETH would imply non-trivial circuit lower bounds

Non-deterministic Strong Exponential Time Hypothesis

k-TAUT: Are all the assignments of a k-DNF formula satisfying?

NSETH: $\forall \varepsilon > 0$, $\exists k$, k-TAUT is not in NTIME($2^{(1-\varepsilon)n}$).

- $\neg$NSETH would imply non-trivial circuit lower bounds
- NSETH is false if randomization is allowed

3-SUM in truly subquadratic co-nondeterministic time

3-SUM is in coNTIME($\tilde{O}(n^{1.5})$)

3-SUM in truly subquadratic co-nondeterministic time

3-SUM is in coNTIME($\tilde{O}(n^{1.5})$)

Certificate for non-existence of a triple summing to 0:

- a prime p among the first $n^{1.5}$ primes $\mathbb{P}_{n^{1.5}}$,
- an integer $t = \tilde{O}(n^{1.5})$, and
- a set S of t triples all summing to 0 modulo p but not to 0

3-SUM in truly subquadratic co-nondeterministic time

3-SUM is in coNTIME($\tilde{O}(n^{1.5})$)

Certificate for non-existence of a triple summing to 0:

- a prime p among the first $n^{1.5}$ primes $\mathbb{P}_{n^{1.5}}$,
- an integer $t = \tilde{O}(n^{1.5})$, and
- a set S of t triples all summing to 0 modulo p but not to 0

Why does such a certificate exist?

$$|\{(a_i, a_j, a_k, p) \mid a_i + a_j + a_k = 0 \mod p\}| \leqslant n^3 \log(3n^c) = \tilde{O}(n^3)$$

$$\exists p \in \mathbb{P}_{n^{1.5}}, |\{(a_i, a_j, a_k) \mid x + y + z = 0 \mod p\}| = \tilde{O}(n^{1.5})$$

3-SUM in truly subquadratic co-nondeterministic time

3-SUM is in coNTIME($\tilde{O}(n^{1.5})$)

Certificate for non-existence of a triple summing to 0:

- a prime p among the first $n^{1.5}$ primes $\mathbb{P}_{n^{1.5}}$,
- an integer $t = \tilde{O}(n^{1.5})$, and
- a set S of t triples all summing to 0 modulo p but not to 0

Given (p, t, S), we check that:

- all triples of S sum to a non-zero value multiple of p
- expand $(\Sigma_i x^{a_i \mod p})^3$ with FFT
- check that the coefficients of x^0, x^p, x^{2p} sum to t

Consequences for the unification

3-SUM is in coNTIME($\tilde{O}(n^{1.5})$)

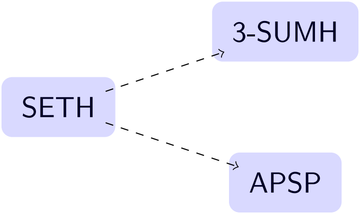
APSP is in coNTIME($\tilde{O}(n^{2+\frac{6+\omega}{9}})$)

Consequences for the unification

3-SUM is in coNTIME($\tilde{O}(n^{1.5})$)

APSP is in coNTIME($\tilde{O}(n^{2+\frac{6+\omega}{9}})$)

A fine-grained **deterministic** reduction from k-SAT to either of these problems would break NSETH



No known implication, the dashed ones are ruled out under NSETH

Things I did not mention

Log shavings and friends

- **Reductions from Circuit Sat** to consolidate a lower bound
- **2 hypotheses** *implying* SETH, 3-SUMH, and APSPH!

Things I did not mention

Log shavings and friends

- **Reductions from Circuit Sat** to consolidate a lower bound
- **2 hypotheses** *implying* SETH, 3-SUMH, and APSPH!
- **FPT in P:** typically algorithms in $k^c n$ or $2^k n$ to circumvent a quadratic/cubic lower bound
- **Distributed PCPs:** hardness of approximation in P

Things I did not mention

Log shavings and friends

- **Reductions from Circuit Sat** to consolidate a lower bound
- **2 hypotheses** *implying* SETH, 3-SUMH, and APSPH!
- **FPT in P:** typically algorithms in $k^c n$ or $2^k n$ to circumvent a quadratic/cubic lower bound
- **Distributed PCPs:** hardness of approximation in P

Thanks for your attention!