Counting 4-Patterns in Permutations Is Equivalent to Counting 4-Cycles in Graphs

Bartłomiej Dudek[1] Paweł Gawrychowski[1]

[1]University of Wrocław

December 14, 2020

Problems concerning permutations

What is the minimal number of swaps of neighboring elements needed to sort a permutation?

$$3 \quad 1 \quad 2 \quad 5 \quad 4$$

Kendall '38

[Kendall's τ rank / bubble-sort] distance counts inversions.

Is the permutation sortable with a stack?

Knuth '68

π can be sorted by a stack iff π avoids 231 (e.g. 1 $\underline{3}$ $\underline{4}$ 5 $\underline{2}$ can't be) .

Problems concerning permutations

What is the minimal number of swaps of neighboring elements needed to sort a permutation?

$$1 \quad 3 \quad 2 \quad 5 \quad 4$$

Kendall '38

[Kendall's τ rank / bubble-sort] distance counts inversions.

Is the permutation sortable with a stack?

Knuth '68

π can be sorted by a stack iff π avoids 231 (e.g. 1 $\underline{3}$ $\underline{4}$ 5 $\underline{2}$ can't be) .

Problems concerning permutations

What is the minimal number of swaps of neighboring elements needed to sort a permutation?

$$1 \quad 2 \quad 3 \quad 5 \quad 4$$

Kendall '38

[Kendall's τ rank / bubble-sort] distance counts inversions.

Is the permutation sortable with a stack?

Knuth '68

π can be sorted by a stack iff π avoids 231 (e.g. 1 3 4 5 2 can't be) .

Problems concerning permutations

What is the minimal number of swaps of neighboring elements needed to sort a permutation?

$1 \quad 2 \quad 3 \quad 4 \quad 5$

Kendall '38

[Kendall's τ rank / bubble-sort] distance counts inversions.

Is the permutation sortable with a stack?

Knuth '68

π can be sorted by a stack iff π avoids 231 (e.g. 1 $\underline{3}$ $\underline{4}$ 5 $\underline{2}$ can't be) .

Problems concerning permutations

What is the minimal number of swaps of neighboring elements needed to sort a permutation?

$1 \quad 2 \quad 3 \quad 4 \quad 5$

Kendall '38

[Kendall's τ rank / bubble-sort] distance counts inversions.

Is the permutation sortable with a stack?

$3 \quad 2 \quad 1 \quad 5 \quad 4 \quad \rightarrow$

Knuth '68

π can be sorted by a stack iff π avoids 231 (e.g. 1 3 4 5 2 can't be) .

Problems concerning permutations

What is the minimal number of swaps of neighboring elements needed to sort a permutation?

$1 \quad 2 \quad 3 \quad 4 \quad 5$

Kendall '38

[Kendall's τ rank / bubble-sort] distance counts inversions.

Is the permutation sortable with a stack?

$2 \quad 1 \quad 5 \quad 4 \quad \rightarrow$

3

Knuth '68

π can be sorted by a stack iff π avoids 231 (e.g. 1 <u>3</u> <u>4</u> 5 <u>2</u> can't be) .

Problems concerning permutations

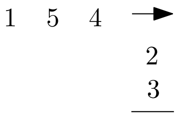
What is the minimal number of swaps of neighboring elements needed to sort a permutation?

$1 \quad 2 \quad 3 \quad 4 \quad 5$

Kendall '38

[Kendall's τ rank / bubble-sort] distance counts inversions.

Is the permutation sortable with a stack?



Knuth '68

π can be sorted by a stack iff π avoids 231 (e.g. 1 3 4 5 2 can't be) .

Problems concerning permutations

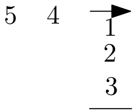
What is the minimal number of swaps of neighboring elements needed to sort a permutation?

$1 \quad 2 \quad 3 \quad 4 \quad 5$

Kendall '38

[Kendall's τ rank / bubble-sort] distance counts inversions.

Is the permutation sortable with a stack?



Knuth '68

π can be sorted by a stack iff π avoids 231 (e.g. 1 3 4 5 2 can't be) .

Problems concerning permutations

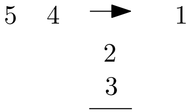
What is the minimal number of swaps of neighboring elements needed to sort a permutation?

$$1 \quad 2 \quad 3 \quad 4 \quad 5$$

Kendall '38

[Kendall's τ rank / bubble-sort] distance counts inversions.

Is the permutation sortable with a stack?



Knuth '68

π can be sorted by a stack iff π avoids 231 (e.g. 1 3 4 5 2 can't be) .

Problems concerning permutations

What is the minimal number of swaps of neighboring elements needed to sort a permutation?

$1 \quad 2 \quad 3 \quad 4 \quad 5$

Kendall '38

[Kendall's τ rank / bubble-sort] distance counts inversions.

Is the permutation sortable with a stack?

$5 \quad 4 \quad \longrightarrow \quad 1 \quad 2$

$\underline{3}$

Knuth '68

π can be sorted by a stack iff π avoids 231 (e.g. 1 3 4 5 2 can't be) .

Problems concerning permutations

What is the minimal number of swaps of neighboring elements needed to sort a permutation?

$1 \quad 2 \quad 3 \quad 4 \quad 5$

Kendall '38

[Kendall's τ rank / bubble-sort] distance counts inversions.

Is the permutation sortable with a stack?

$5 \quad 4 \quad \longrightarrow \quad 1 \quad 2 \quad 3$

Knuth '68

π can be sorted by a stack iff π avoids 231 (e.g. 1 3 4 5 2 can't be) .

Problems concerning permutations

What is the minimal number of swaps of neighboring elements needed to sort a permutation?

$1 \quad 2 \quad 3 \quad 4 \quad 5$

Kendall '38

[Kendall's τ rank / bubble-sort] distance counts inversions.

Is the permutation sortable with a stack?

$4 \quad \longrightarrow \quad 1 \quad 2 \quad 3$

5

Knuth '68

π can be sorted by a stack iff π avoids 231 (e.g. 1 <u>3</u> <u>4</u> 5 <u>2</u> can't be) .

Problems concerning permutations

What is the minimal number of swaps of neighboring elements needed to sort a permutation?

$1 \quad 2 \quad 3 \quad 4 \quad 5$

Kendall '38

[Kendall's τ rank / bubble-sort] distance counts inversions.

Is the permutation sortable with a stack?

$\longrightarrow \quad 1 \quad 2 \quad 3$

4

5

Knuth '68

π can be sorted by a stack iff π avoids 231 (e.g. 1 3 4 5 2 can't be) .

Problems concerning permutations

What is the minimal number of swaps of neighboring elements needed to sort a permutation?

1 2 3 4 5

Kendall '38

[Kendall's τ rank / bubble-sort] distance counts inversions.

Is the permutation sortable with a stack?

→ 1 2 3 4

5

Knuth '68

π can be sorted by a stack iff π avoids 231 (e.g. 1 3 4 5 2 can't be) .

Problems concerning permutations

What is the minimal number of swaps of neighboring elements needed to sort a permutation?

$1 \quad 2 \quad 3 \quad 4 \quad 5$

Kendall '38

[Kendall's τ rank / bubble-sort] distance counts inversions.

Is the permutation sortable with a stack?

$\longrightarrow \quad 1 \quad 2 \quad 3 \quad 4 \quad 5$

Knuth '68

π can be sorted by a stack iff π avoids 231 (e.g. 1 $\underline{3}$ $\underline{4}$ 5 $\underline{2}$ can't be) .

Order-isomorphism and permutation patterns

1 <u>5</u> 2 <u>4</u> 3 <u>6</u>

↓

$5\ 4\ 6 \approx 2\ 1\ 3$

- 21: inversion (e.g. <u>3</u> 2 <u>1</u>)
- 1234: increasing subsequence (e.g. 8 <u>1</u> 5 <u>3</u> 2 <u>4</u> <u>7</u> 6)

How efficiently can we count a pattern in a permutation?

Order-isomorphism and permutation patterns

1 <u>5</u> 2 <u>4</u> 3 <u>6</u>

$\downarrow$

5 4 6 $\approx$ 2 1 3

- 21: inversion (e.g. <u>3</u> 2 <u>1</u>)
- 1234: increasing subsequence (e.g. 8 <u>1</u> 5 <u>3</u> 2 <u>4</u> <u>7</u> 6)

How efficiently can we count a pattern in a permutation?

Order-isomorphism and permutation patterns

$$1\ \underline{5}\ 2\ \underline{4}\ 3\ \underline{6}$$

$$\downarrow$$

$$5\ 4\ 6 \approx 2\ 1\ 3$$

- 21: inversion (e.g. $\underline{3}\ 2\ \underline{1}$)
- 1234: increasing subsequence (e.g. $8\ \underline{1}\ 5\ \underline{3}\ 2\ \underline{4}\ \underline{7}\ 6$)

How efficiently can we count a pattern in a permutation?

Order-isomorphism and permutation patterns

1 5 2 4 3 6

↓

5 4 6 ≈ 2 1 3

- 21: inversion (e.g. 3 2 1)
- 1234: increasing subsequence (e.g. 8 1 5 3 2 4 7 6)

How efficiently can we count a pattern in a permutation?

Order-isomorphism and permutation patterns

1 5 2 4 3 6

$\downarrow$

$5\ 4\ 6 \approx 2\ 1\ 3$

- 21: inversion (e.g. 3 2 1)
- 1234: increasing subsequence (e.g. 8 1 5 3 2 4 7 6)

How efficiently can we count a pattern in a permutation?

History of pattern detection

For patterns of length k:

Year	Authors	Runtime
1998	Bose et al.	NP-hard
-	trivial	$\mathcal{O}(n^k)$
2001	Albert et al.	$\mathcal{O}(n^{2k/3+1})$
2008	Ahal & Rabinovich	$n^{0.47k+o(k)}$
2013	Guillemot & Marx	$2^{\mathcal{O}(k^2 \log k)} n$
2013	Fox	$2^{\mathcal{O}(k^2)} n$

Patterns of constant length can be detected in $\mathcal{O}(n)$ time.

History of pattern detection

For patterns of length k:

Year	Authors	Runtime
1998	Bose et al.	NP-hard
-	trivial	$\mathcal{O}(n^k)$
2001	Albert et al.	$\mathcal{O}(n^{2k/3+1})$
2008	Ahal & Rabinovich	$n^{0.47k+o(k)}$
2013	Guillemot & Marx	$2^{\mathcal{O}(k^2 \log k)} n$
2013	Fox	$2^{\mathcal{O}(k^2)} n$

Patterns of constant length can be detected in $\mathcal{O}(n)$ time.

History of pattern detection

For patterns of length k:

Year	Authors	Runtime
1998	Bose et al.	NP-hard
-	trivial	$\mathcal{O}(n^k)$
2001	Albert et al.	$\mathcal{O}(n^{2k/3+1})$
2008	Ahal & Rabinovich	$n^{0.47k+o(k)}$
2013	Guillemot & Marx	$2^{\mathcal{O}(k^2 \log k)} n$
2013	Fox	$2^{\mathcal{O}(k^2)} n$

Patterns of constant length can be detected in $\mathcal{O}(n)$ time.

History of counting patterns

For patterns of length k:

Year	Authors	Runtime
-	trivial	$\mathcal{O}(n^k)$
2001	Albert et al.	$\mathcal{O}(n^{2k/3+1})$
2019	Berendsohn et al.	$n^{k/4+o(k)}$

Berendsohn et al. [IPEC'19]

If patterns of length k can be counted in $f(k)n^{o(k/\log k)}$, then ETH fails.

History of counting patterns

For patterns of length k:

Year	Authors	Runtime
-	trivial	$\mathcal{O}(n^k)$
2001	Albert et al.	$\mathcal{O}(n^{2k/3+1})$
2019	Berendsohn et al.	$n^{k/4+o(k)}$

Berendsohn et al. [IPEC'19]

If patterns of length k can be counted in $f(k)n^{o(k/\log k)}$, then ETH fails.

History of counting patterns

For patterns of length k:

Year	Authors	Runtime
-	trivial	$\mathcal{O}(n^k)$
2001	Albert et al.	$\mathcal{O}(n^{2k/3+1})$
2019	Berendsohn et al.	$n^{k/4+o(k)}$

Berendsohn et al. [IPEC'19]

If patterns of length k can be counted in $f(k)n^{o(k/\log k)}$, then ETH fails.

How difficult is it to count short patterns?

Folklore

Patterns of length $k \leq 3$ can be counted in $\tilde{\mathcal{O}}(n)$ time.

4-patterns: $\mathcal{O}(n^2)$ [HellerH'16, WeihsDL'16, WeihsDM'18]

Even-Zohar and Leng [SODA'21]

- eight 4-patterns can be counted in $\tilde{\mathcal{O}}(n)$ time
- remaining 4-patterns can be counted in $\tilde{\mathcal{O}}(n^{1.5})$ time

This work

- Can we do better? $\mathcal{O}(n^{1.48})$
- Can we do even better? (probably) no $\mathcal{O}(n^{4/3-\varepsilon})$

How difficult is it to count short patterns?

Folklore

Patterns of length $k \leq 3$ can be counted in $\tilde{\mathcal{O}}(n)$ time.

4-patterns: $\mathcal{O}(n^2)$ [HellerH'16, WeihsDL'16, WeihsDM'18]

Even-Zohar and Leng [SODA'21]

- eight 4-patterns can be counted in $\tilde{\mathcal{O}}(n)$ time
- remaining 4-patterns can be counted in $\tilde{\mathcal{O}}(n^{1.5})$ time

This work

- Can we do better? $\mathcal{O}(n^{1.48})$
- Can we do even better? (probably) no $\mathcal{O}(n^{4/3-\varepsilon})$

How difficult is it to count short patterns?

Folklore

Patterns of length $k \leq 3$ can be counted in $\tilde{\mathcal{O}}(n)$ time.

4-patterns: $\mathcal{O}(n^2)$ [HellerH'16, WeihsDL'16, WeihsDM'18]

Even-Zohar and Leng [SODA'21]

- eight 4-patterns can be counted in $\tilde{\mathcal{O}}(n)$ time
- remaining 4-patterns can be counted in $\tilde{\mathcal{O}}(n^{1.5})$ time

This work

- Can we do better? $\mathcal{O}(n^{1.48})$
- Can we do even better? (probably) no $\mathcal{O}(n^{4/3-\varepsilon})$

How difficult is it to count short patterns?

Folklore

Patterns of length $k \leq 3$ can be counted in $\tilde{\mathcal{O}}(n)$ time.

4-patterns: $\mathcal{O}(n^2)$ [HellerH'16, WeihsDL'16, WeihsDM'18]

Even-Zohar and Leng [SODA'21]

- eight 4-patterns can be counted in $\tilde{\mathcal{O}}(n)$ time
- remaining 4-patterns can be counted in $\tilde{\mathcal{O}}(n^{1.5})$ time

This work

- Can we do better? $\mathcal{O}(n^{1.48})$
- Can we do even better? (probably) no $\mathcal{O}(n^{4/3-\varepsilon})$

How difficult is it to count short patterns?

Folklore

Patterns of length $k \leq 3$ can be counted in $\tilde{\mathcal{O}}(n)$ time.

4-patterns: $\mathcal{O}(n^2)$ [HellerH'16, WeihsDL'16, WeihsDM'18]

Even-Zohar and Leng [SODA'21]

- eight 4-patterns can be counted in $\tilde{\mathcal{O}}(n)$ time
- remaining 4-patterns can be counted in $\tilde{\mathcal{O}}(n^{1.5})$ time

This work

- Can we do better? $\mathcal{O}(n^{1.48})$
- Can we do even better? (probably) no $\mathcal{O}(n^{4/3-\varepsilon})$

How difficult is it to count short patterns?

Folklore

Patterns of length $k \leq 3$ can be counted in $\tilde{\mathcal{O}}(n)$ time.

4-patterns: $\mathcal{O}(n^2)$ [HellerH'16, WeihsDL'16, WeihsDM'18]

Even-Zohar and Leng [SODA'21]

- eight 4-patterns can be counted in $\tilde{\mathcal{O}}(n)$ time
- remaining 4-patterns can be counted in $\tilde{\mathcal{O}}(n^{1.5})$ time

This work

- Can we do better? $\mathcal{O}(n^{1.48})$
- Can we do even better? (probably) no $\mathcal{O}(n^{4/3-\varepsilon})$

How difficult is it to count short patterns?

Folklore

Patterns of length $k \leq 3$ can be counted in $\tilde{\mathcal{O}}(n)$ time.

4-patterns: $\mathcal{O}(n^2)$ [HellerH'16, WeihsDL'16, WeihsDM'18]

Even-Zohar and Leng [SODA'21]

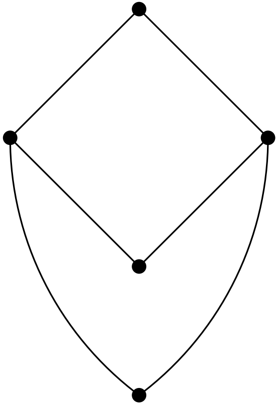
- eight 4-patterns can be counted in $\tilde{\mathcal{O}}(n)$ time
- remaining 4-patterns can be counted in $\tilde{\mathcal{O}}(n^{1.5})$ time

This work

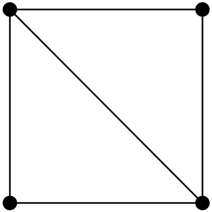
- Can we do better? $\mathcal{O}(n^{1.48})$
- Can we do even better? (probably) no $\mathcal{O}(n^{4/3-\varepsilon})$

Counting 4-cycles

Input: simple, undirected graph
Output: number of simple cycles of length 4



3



1

History of $2k$-cycles

Year	Authors	Runtime	Variant
	Folklore	$\mathcal{O}(n^3)$	
1997	Alon et al.	$\mathcal{O}(n^\omega)$	count 4-cycles
		$\mathcal{O}(m^{4/3})$	find a 4-cycle
1997	Yuster and Zwick	$\mathcal{O}(n^2)$	find a $2k$-cycle
2015	Vassilevska Williams et al.	$\mathcal{O}(m^{1.48})$	count 4-cycles
2017	Dahlgaard et al.	$\mathcal{O}(m^{2k/(k+1)})$	find a $2k$-cycle

Conjecture [Yuster and Zwick, J. Discr. Math.'97]

For every $\varepsilon > 0$ no algorithm detects 4-cycles in $\mathcal{O}(n^{2-\varepsilon})$ time.

$\Downarrow$

Conjecture [Dahlgaard et al., STOC'17]

For every $\varepsilon > 0$ no algorithm detects 4-cycles in $\mathcal{O}(m^{4/3-\varepsilon})$ time.

History of $2k$-cycles

Year	Authors	Runtime	Variant
	Folklore	$\mathcal{O}(n^3)$	
1997	Alon et al.	$\mathcal{O}(n^\omega)$	count 4-cycles
		$\mathcal{O}(m^{4/3})$	find a 4-cycle
1997	Yuster and Zwick	$\mathcal{O}(n^2)$	find a $2k$-cycle
2015	Vassilevska Williams et al.	$\mathcal{O}(m^{1.48})$	count 4-cycles
2017	Dahlgaard et al.	$\mathcal{O}(m^{2k/(k+1)})$	find a $2k$-cycle

Conjecture [Yuster and Zwick, J. Discr. Math.'97]

For every $\varepsilon > 0$ no algorithm detects 4-cycles in $\mathcal{O}(n^{2-\varepsilon})$ time.

$\Downarrow$

Conjecture [Dahlgaard et al., STOC'17]

For every $\varepsilon > 0$ no algorithm detects 4-cycles in $\mathcal{O}(m^{4/3-\varepsilon})$ time.

History of $2k$-cycles

Year	Authors	Runtime	Variant
	Folklore	$\mathcal{O}(n^3)$	
1997	Alon et al.	$\mathcal{O}(n^\omega)$	count 4-cycles
		$\mathcal{O}(m^{4/3})$	find a 4-cycle
1997	Yuster and Zwick	$\mathcal{O}(n^2)$	find a $2k$-cycle
2015	Vassilevska Williams et al.	$\mathcal{O}(m^{1.48})$	count 4-cycles
2017	Dahlgaard et al.	$\mathcal{O}(m^{2k/(k+1)})$	find a $2k$-cycle

Conjecture [Yuster and Zwick, J. Discr. Math.'97]

For every $\varepsilon > 0$ no algorithm detects 4-cycles in $\mathcal{O}(n^{2-\varepsilon})$ time.

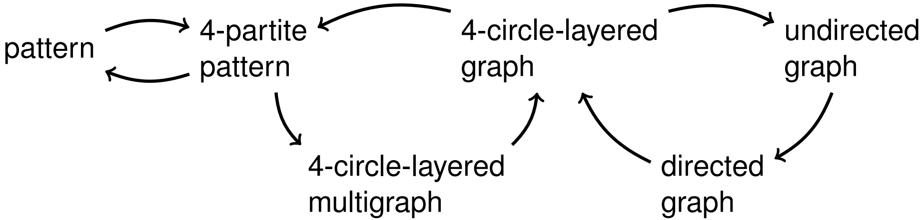
$$\Downarrow$$

Conjecture [Dahlgaard et al., STOC'17]

For every $\varepsilon > 0$ no algorithm detects 4-cycles in $\mathcal{O}(m^{4/3-\varepsilon})$ time.

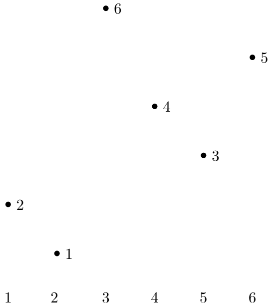
Our contribution

Counting 4-patterns in permutations is equivalent to counting 4-cycles in graphs



Geometric interpretation

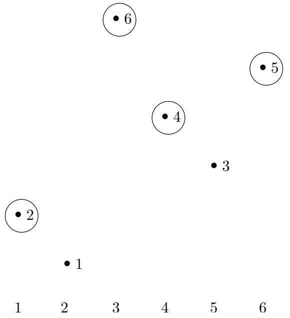
For permutation π, create set of points $S_\pi = \{(i, \pi(i)) : i \in [n]\}$:



- horizontal reflection ($\leftrightarrow$) reverses the pattern:
  $1423 \implies 3241$
- vertical reflection ($\updownarrow$) replaces element x with $(n+1) - x$:
  $1423 \implies 4132$

Geometric interpretation

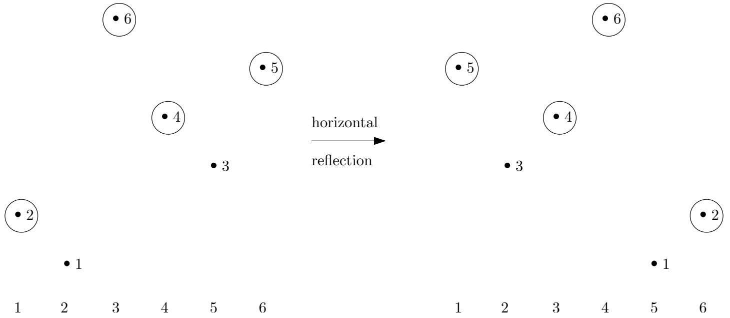
For permutation π, create set of points $S_\pi = \{(i, \pi(i)) : i \in [n]\}$:



- horizontal reflection ($\leftrightarrow$) reverses the pattern:
  1423 $\implies$ 3241
- vertical reflection ($\updownarrow$) replaces element x with $(n+1) - x$:
  1423 $\implies$ 4132

Geometric interpretation

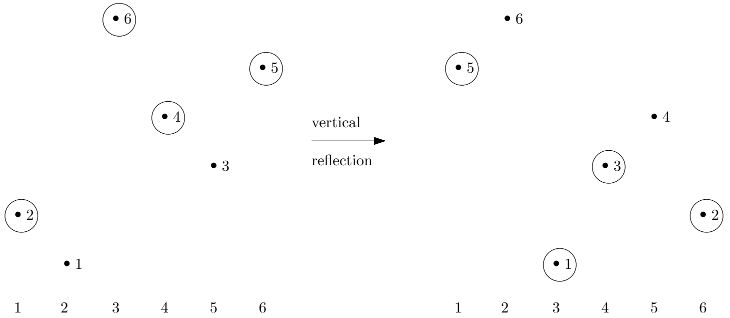
For permutation π, create set of points $S_\pi = \{(i, \pi(i)) : i \in [n]\}$:



- horizontal reflection ($\leftrightarrow$) reverses the pattern:
  1423 $\implies$ 3241
- vertical reflection ($\updownarrow$) replaces element x with $(n+1) - x$:
  1423 $\implies$ 4132

Geometric interpretation

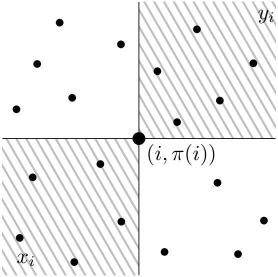
For permutation π, create set of points $S_\pi = \{(i, \pi(i)) : i \in [n]\}$:



- horizontal reflection ($\leftrightarrow$) reverses the pattern:
  $1423 \implies 3241$
- vertical reflection ($\updownarrow$) replaces element x with $(n+1) - x$:
  $1423 \implies 4132$

Warm-up: counting 123 and 132

How many occurrences of 123 have middle element at position *i*?



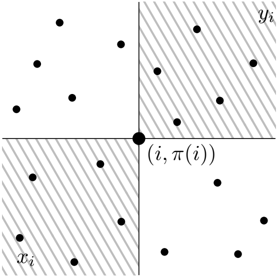
x_i, y_i can be retrieved with orthogonal range queries in $\mathcal{O}(\log n)$ time.

$$\#_{123}(\pi) = \sum_{i=1}^{n} x_i \cdot y_i$$

and then: $\#_{132}(\pi) = \sum_{i=1}^{n} \binom{y_i}{2} - \#_{123}(\pi)$

Warm-up: counting 123 and 132

How many occurrences of 123 have middle element at position i?



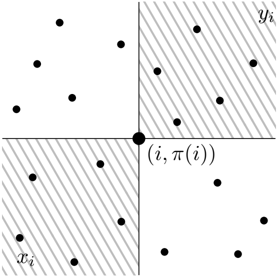
x_i, y_i can be retrieved with orthogonal range queries in $\mathcal{O}(\log n)$ time.

$$\#_{123}(\pi) = \sum_{i=1}^{n} x_i \cdot y_i$$

and then: $\#_{132}(\pi) = \sum_{i=1}^{n} \binom{y_i}{2} - \#_{123}(\pi)$

Warm-up: counting 123 and 132

How many occurrences of 123 have middle element at position i?



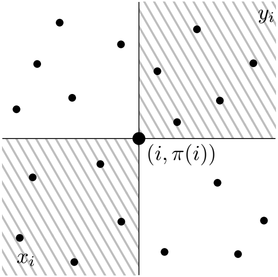
x_i, y_i can be retrieved with orthogonal range queries in $\mathcal{O}(\log n)$ time.

$$\#_{123}(\pi) = \sum_{i=1}^{n} x_i \cdot y_i$$

and then: $\#_{132}(\pi) = \sum_{i=1}^{n} \binom{y_i}{2} - \#_{123}(\pi)$

Warm-up: counting 123 and 132

How many occurrences of 123 have middle element at position *i*?



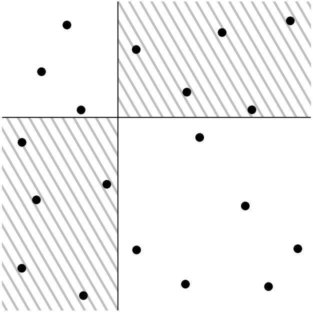
x_i, y_i can be retrieved with orthogonal range queries in $\mathcal{O}(\log n)$ time.

$$\#_{123}(\pi) = \sum_{i=1}^{n} x_i \cdot y_i$$

and then: $\#_{132}(\pi) = \sum_{i=1}^{n} \binom{y_i}{2} - \#_{123}(\pi)$

Let's make our lives simpler

Consider a division of the plane into quadrants:

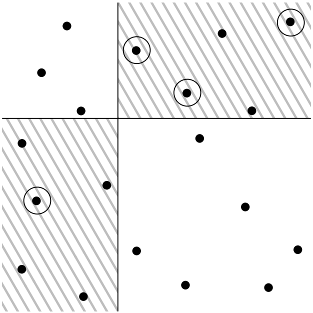


How many occurrences of 1324 look like $\frac{0|3}{1|0}$? $|BL| \cdot \#_{213}(TR)$
How many occurrences of 2134 look like $\frac{0|2}{2|0}$? $\#_{21}(BL) \cdot \#_{12}(TR)$

We disregard partitions: $\frac{2|2}{0|0}, \frac{3|1}{0|0}, \frac{4|0}{0|0}, \ldots$

Let's make our lives simpler

Consider a division of the plane into quadrants:



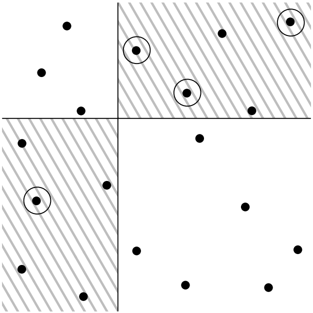
How many occurrences of 1324 look like $\frac{0|3}{1|0}$? $|BL| \cdot \#_{213}(TR)$

How many occurrences of 2134 look like $\frac{0|2}{2|0}$? $\#_{21}(BL) \cdot \#_{12}(TR)$

We disregard partitions: $\frac{2|2}{0|0}, \frac{3|1}{0|0}, \frac{4|0}{0|0}, \ldots$

Let's make our lives simpler

Consider a division of the plane into quadrants:



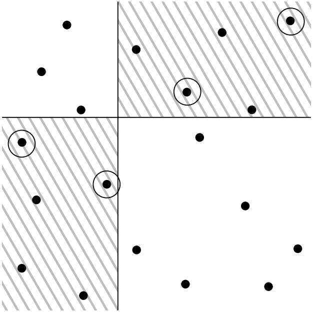
How many occurrences of 1324 look like $\frac{0|3}{1|0}$? $|BL| \cdot \#_{213}(TR)$

How many occurrences of 2134 look like $\frac{0|2}{2|0}$? $\#_{21}(BL) \cdot \#_{12}(TR)$

We disregard partitions: $\frac{2|2}{0|0}, \frac{3|1}{0|0}, \frac{4|0}{0|0}, \ldots$

Let’s make our lives simpler

Consider a division of the plane into quadrants:

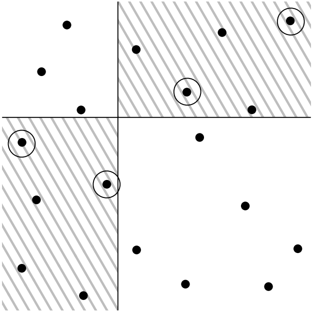


How many occurrences of 1324 look like $\frac{0|3}{1|0}$? $|BL| \cdot \#_{213}(TR)$
How many occurrences of 2134 look like $\frac{0|2}{2|0}$? $\#_{21}(BL) \cdot \#_{12}(TR)$

We disregard partitions: $\frac{2|2}{0|0}, \frac{3|1}{0|0}, \frac{4|0}{0|0}, \ldots$

Let's make our lives simpler

Consider a division of the plane into quadrants:

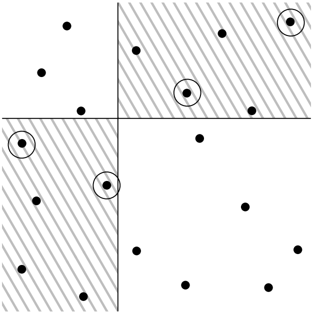


How many occurrences of 1324 look like $\frac{0|3}{1|0}$? $|BL| \cdot \#_{213}(TR)$
How many occurrences of 2134 look like $\frac{0|2}{2|0}$? $\#_{21}(BL) \cdot \#_{12}(TR)$

We disregard partitions: $\frac{2|2}{0|0}, \frac{3|1}{0|0}, \frac{4|0}{0|0}, \ldots$

Let's make our lives simpler

Consider a division of the plane into quadrants:

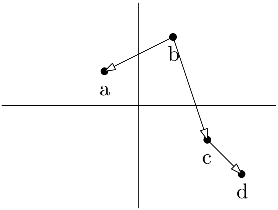


How many occurrences of 1324 look like $\frac{0|3}{1|0}$? $|BL| \cdot \#_{213}(TR)$
How many occurrences of 2134 look like $\frac{0|2}{2|0}$? $\#_{21}(BL) \cdot \#_{12}(TR)$

We disregard partitions: $\frac{2|2}{0|0}, \frac{3|1}{0|0}, \frac{4|0}{0|0}, \ldots$

3 quadrants

Consider 3421 that looks like $\frac{1|1}{0|2}$:



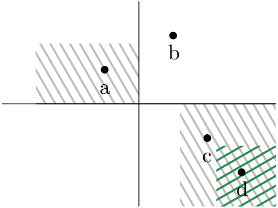
After preprocessing of BR, we count triples (a, c, d) that together with b form 3421 in $\mathcal{O}(\log n)$ time.

With similar techniques we can count patterns that look like $\frac{1|2}{0|1}$.

$\implies$ we can count patterns in 2 or 3 quadrants in $\tilde{\mathcal{O}}(n)$ time!

3 quadrants

Consider 3421 that looks like $\frac{1|1}{0|2}$:



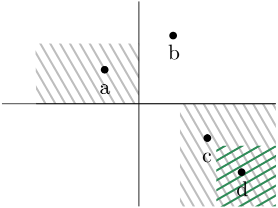
After preprocessing of BR, we count triples (a, c, d) that together with b form 3421 in $\mathcal{O}(\log n)$ time.

With similar techniques we can count patterns that look like $\frac{1|2}{0|1}$.

$\implies$ we can count patterns in 2 or 3 quadrants in $\tilde{\mathcal{O}}(n)$ time!

3 quadrants

Consider 3421 that looks like $\frac{1|1}{0|2}$:



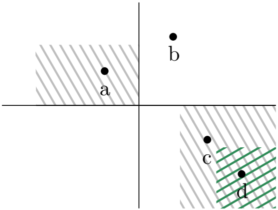
After preprocessing of BR, we count triples (a, c, d) that together with b form 3421 in $\mathcal{O}(\log n)$ time.

With similar techniques we can count patterns that look like $\frac{1|2}{0|1}$.

$\implies$ we can count patterns in 2 or 3 quadrants in $\tilde{\mathcal{O}}(n)$ time!

3 quadrants

Consider 3421 that looks like $\frac{1|1}{0|2}$:



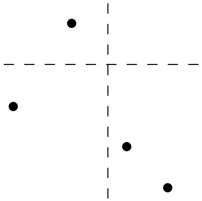
After preprocessing of *BR*, we count triples (a, c, d) that together with b form 3421 in $\mathcal{O}(\log n)$ time.

With similar techniques we can count patterns that look like $\frac{1|2}{0|1}$.

$\Longrightarrow$ we can count patterns in 2 or 3 quadrants in $\tilde{\mathcal{O}}(n)$ time!

4 quadrants?

Wait, 3421 cannot be spread between 4 quadrants $\frac{1|1}{1|1}$!

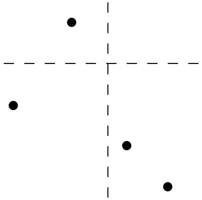


But there are many divisions of the plane :(

$\Longrightarrow$ Divide & Conquer!

4 quadrants?

Wait, 3421 cannot be spread between 4 quadrants $\frac{1|1}{1|1}$!

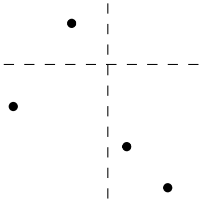


But there are many divisions of the plane :(

$\Longrightarrow$ Divide & Conquer!

4 quadrants?

Wait, 3421 cannot be spread between 4 quadrants $\frac{1|1}{1|1}$!

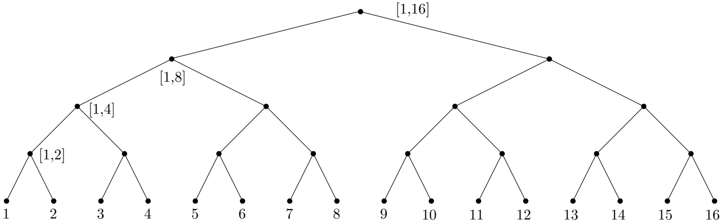


But there are many divisions of the plane :(

$\Longrightarrow$ Divide & Conquer!

Minimum Base Ranges

Full binary tree on $[n]$:

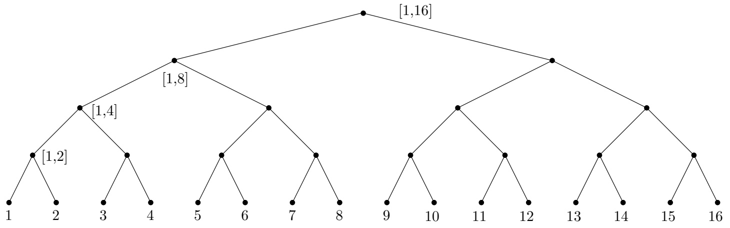


Observation: every element belongs to $\log n$ base ranges.

Minimum Base Range(S) $\equiv$ Range(Lowest Common Ancestor(S))

Minimum Base Ranges

Full binary tree on $[n]$:

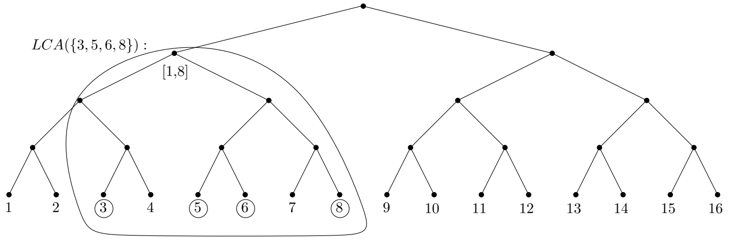


Observation: every element belongs to $\log n$ base ranges.

Minimum Base Range(S) $\equiv$ Range(Lowest Common Ancestor(S))

Minimum Base Ranges

Full binary tree on $[n]$:

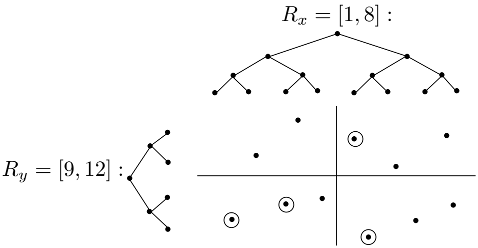


Observation: every element belongs to $\log n$ base ranges.

Minimum Base Range(S) $\equiv$ Range(Lowest Common Ancestor(S))

Which divisions to consider?

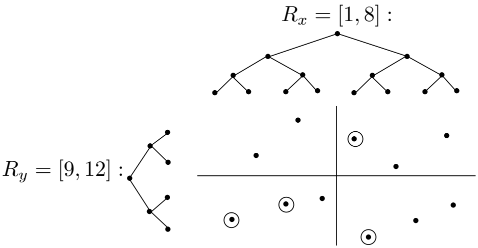
Idea: for every relevant pair of MBRs over *x*-s and *y*-s, we consider the division of the plane that halves each of the MBRs.



$(i, \pi(i))$ belongs to $\log^2 n$ pairs of MBRs $\implies \mathcal{O}(n \log^2 n)$ total size.

Which divisions to consider?

Idea: for every relevant pair of MBRs over *x*-s and *y*-s, we consider the division of the plane that halves each of the MBRs.



$(i, \pi(i))$ belongs to $\log^2 n$ pairs of MBRs $\implies \mathcal{O}(n \log^2 n)$ total size.

Summary

1. With MBRs we identified a number of relevant divisions of the plane with total size $\tilde{O}(n)$
2. For each instance of size s:
   - Check all splits of the pattern into 2 or 3 quadrants: $\frac{1|1}{2|0}, \frac{3|0}{0|1}, \ldots$
     - ★ Count occurrences of the pattern in $\tilde{O}(s)$ time

This gives $\tilde{O}(n)$-time algorithm but for the pattern that cannot be spread over 4 quadrants...

Corollary [cf. Even-Zohar and Leng]

All trivial 4-patterns $(1234, 1243, 2134, 2143, 4321, 4312, 3421, 3412)$ in permutations of length n can be counted in $\tilde{O}(n)$ time.

Summary

1. With MBRs we identified a number of relevant divisions of the plane with total size $\tilde{\mathcal{O}}(n)$
2. For each instance of size s:
   - Check all splits of the pattern into 2 or 3 quadrants: $\frac{1|1}{2|0}, \frac{3|0}{0|1}, \ldots$
     - Count occurrences of the pattern in $\tilde{\mathcal{O}}(s)$ time

This gives $\tilde{\mathcal{O}}(n)$-time algorithm but for the pattern that cannot be spread over 4 quadrants...

Corollary [cf. Even-Zohar and Leng]

All trivial 4-patterns $(1234, 1243, 2134, 2143, 4321, 4312, 3421, 3412)$ in permutations of length n can be counted in $\tilde{\mathcal{O}}(n)$ time.

Summary

1. With MBRs we identified a number of relevant divisions of the plane with total size $\tilde{O}(n)$
2. For each instance of size s:
   - Check all splits of the pattern into 2 or 3 quadrants: $\frac{1|1}{2|0}, \frac{3|0}{0|1}, \dots$
     - Count occurrences of the pattern in $\tilde{O}(s)$ time

This gives $\tilde{O}(n)$-time algorithm but for the pattern that cannot be spread over 4 quadrants...

Corollary [cf. Even-Zohar and Leng]

All trivial 4-patterns (1234, 1243, 2134, 2143, 4321, 4312, 3421, 3412) in permutations of length n can be counted in $\tilde{O}(n)$ time.

Summary

1. With MBRs we identified a number of relevant divisions of the plane with total size $\tilde{\mathcal{O}}(n)$
2. For each instance of size s:
   - Check all splits of the pattern into 2 or 3 quadrants: $\frac{1|1}{2|0}, \frac{3|0}{0|1}, \ldots$
     - ★ Count occurrences of the pattern in $\tilde{\mathcal{O}}(s)$ time

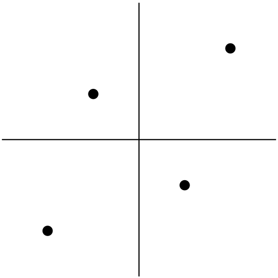
This gives $\tilde{\mathcal{O}}(n)$-time algorithm but for the pattern that cannot be spread over 4 quadrants...

Corollary [cf. Even-Zohar and Leng]

All trivial 4-patterns $(1234, 1243, 2134, 2143, 4321, 4312, 3421, 3412)$ in permutations of length n can be counted in $\tilde{\mathcal{O}}(n)$ time.

Non-trivial patterns

Some patterns can be spread over 4 quadrants, take 1324:

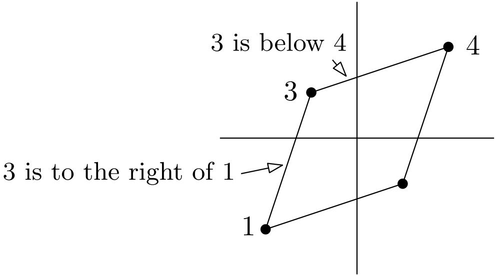


Problem: every point constrains two other.

We have a 4-cycle!

Non-trivial patterns

Some patterns can be spread over 4 quadrants, take 1324:

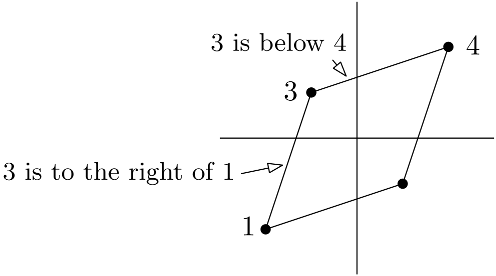


Problem: every point constrains two other.

We have a 4-cycle!

Non-trivial patterns

Some patterns can be spread over 4 quadrants, take 1324:

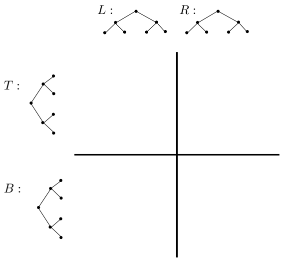


Problem: every point constrains two other.

We have a 4-cycle!

MBRs build graph

Build full binary tree on elements from each **half-axis**.

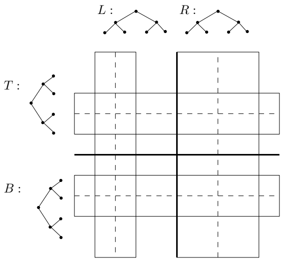


- Choose MBR in each half-axis.
- Pattern 1324 determines which half of each MBR to consider.
- Construct a 4-partite directed multigraph

$$\#_{1324_4}(\pi) = \#C_4(G)$$

MBRs build graph

Build full binary tree on elements from each **half-axis**.

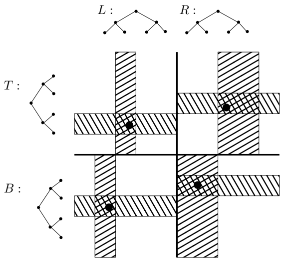


- Choose MBR in each half-axis.
- Pattern 1324 determines which half of each MBR to consider.
- Construct a 4-partite directed multigraph

$$\#_{1324_4}(\pi) = \#C_4(G)$$

MBRs build graph

Build full binary tree on elements from each **half-axis**.

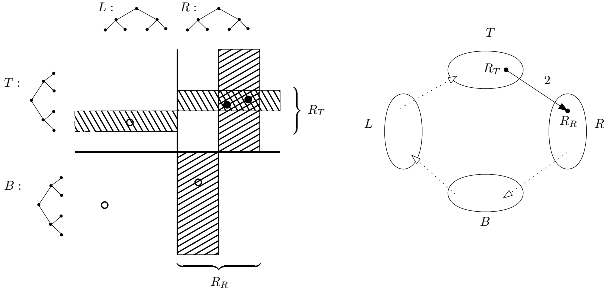


- Choose MBR in each half-axis.
- Pattern 1324 determines which half of each MBR to consider.
- Construct a 4-partite directed multigraph

$$\#_{1324_4}(\pi) = \#C_4(G)$$

MBRs build graph

Build full binary tree on elements from each **half-axis**.

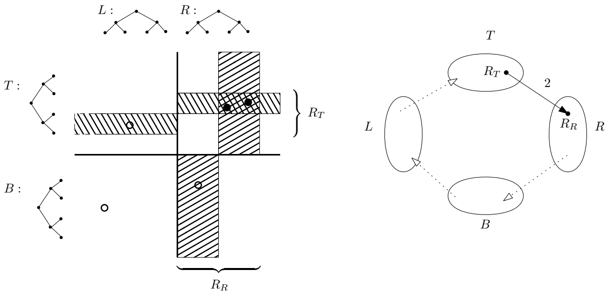


- Choose MBR in each half-axis.
- Pattern 1324 determines which half of each MBR to consider.
- Construct a 4-partite directed multigraph

$$\#_{1324_4}(\pi) = \#C_4(G)$$

MBRs build graph

Build full binary tree on elements from each **half-axis**.

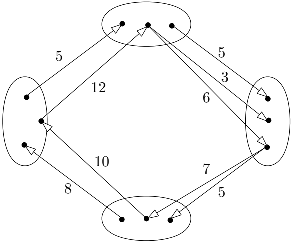


- Choose MBR in each half-axis.
- Pattern 1324 determines which half of each MBR to consider.
- Construct a 4-partite directed multigraph

$$\#_{1324_4}(\pi) = \#C_4(G)$$

Summary

We reduced counting 1324_4 on n points to counting 4-cycles in a 4-circle-layered multigraph on $\tilde{O}(n)$ edges where $\text{MULT}(e) \leq n$:



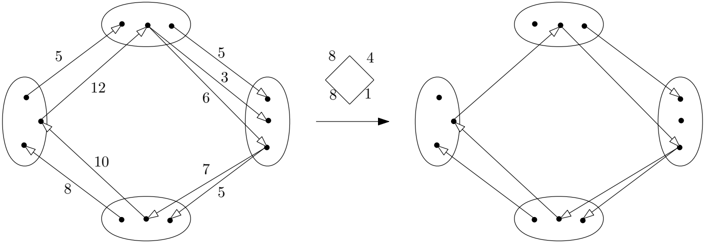
Guess the power of 2 on each side of the graph and multiply by $2^{\sum p_i}$.

Theorem

We reduced counting a pattern on n points to a number of instances of counting C_4 in 4-circle-layered graphs that have in total $\tilde{O}(n)$ edges.

Summary

We reduced counting 1324_4 on n points to counting 4-cycles in a 4-circle-layered multigraph on $\tilde{O}(n)$ edges where $\text{MULT}(e) \leq n$:



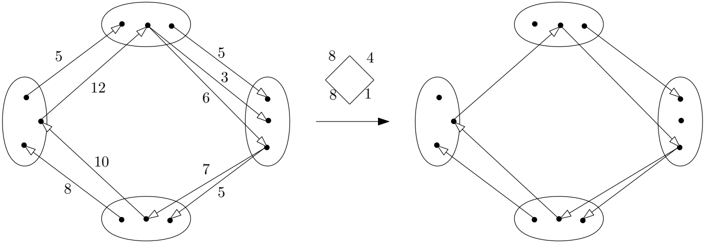
Guess the power of 2 on each side of the graph and multiply by $2^{\sum p_i}$.

Theorem

We reduced counting a pattern on n points to a number of instances of counting C_4 in 4-circle-layered graphs that have in total $\tilde{O}(n)$ edges.

Summary

We reduced counting 1324_4 on n points to counting 4-cycles in a 4-circle-layered multigraph on $\tilde{O}(n)$ edges where $\textsc{mult}(e) \leq n$:

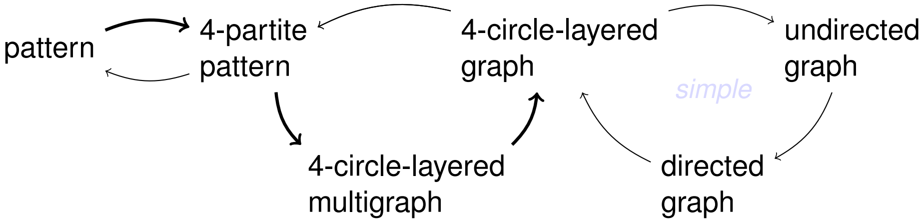


Guess the power of 2 on each side of the graph and multiply by $2^{\sum p_i}$.

Theorem

We reduced counting a pattern on n points to a number of instances of counting C_4 in 4-circle-layered graphs that have in total $\tilde{O}(n)$ edges.

Where are we?

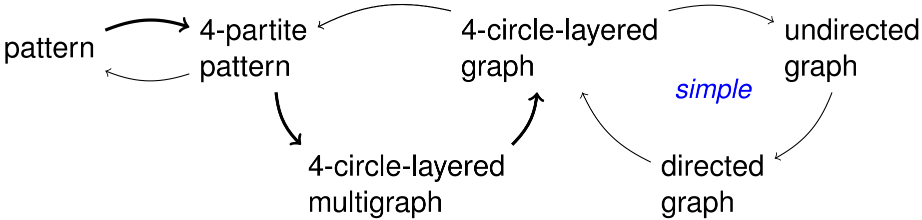


Theorem

Counting 4-cycles in simple graphs in $\mathcal{O}(m^{\delta})$ time gives $\tilde{\mathcal{O}}(n^{\delta})$-time algorithm for counting non-trivial 4-patterns.

$\implies$ an $\mathcal{O}(n^{1.48})$-time algorithm for $\#_{1324}(\pi)$.

Where are we?

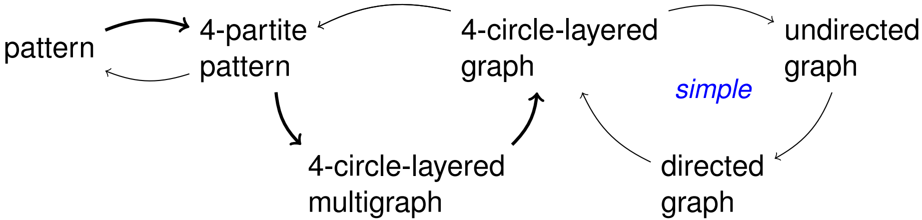


Theorem

Counting 4-cycles in simple graphs in $\mathcal{O}(m^\delta)$ time gives $\tilde{\mathcal{O}}(n^\delta)$-time algorithm for counting non-trivial 4-patterns.

$\implies$ an $\mathcal{O}(n^{1.48})$-time algorithm for $\#_{1324}(\pi)$.

Where are we?

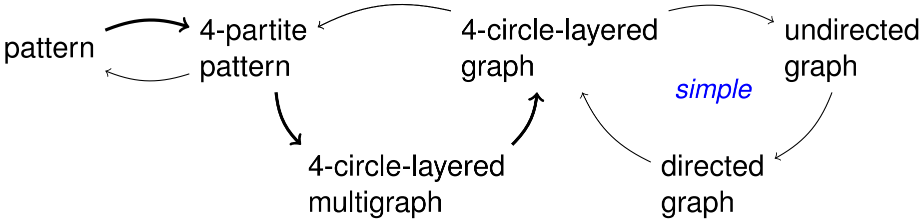


Theorem

Counting 4-cycles in simple graphs in $\mathcal{O}(m^\delta)$ time gives $\tilde{\mathcal{O}}(n^\delta)$-time algorithm for counting non-trivial 4-patterns.

$\implies$ an $\mathcal{O}(n^{1.48})$-time algorithm for $\#_{1324}(\pi)$.

Where are we?

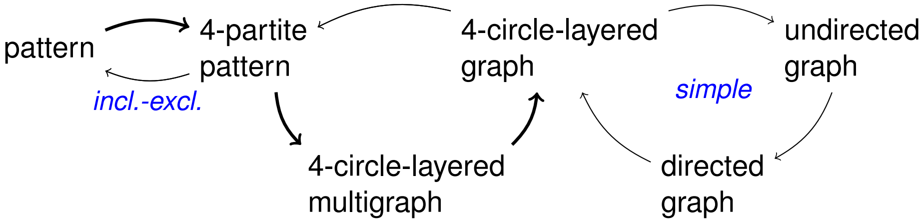


Theorem

Counting 4-cycles in simple graphs in $\mathcal{O}(m^{\delta})$ time gives $\tilde{\mathcal{O}}(n^{\delta})$-time algorithm for counting non-trivial 4-patterns.

$\implies$ an $\mathcal{O}(n^{1.48})$-time algorithm for $\#_{1324}(\pi)$.

Where are we?

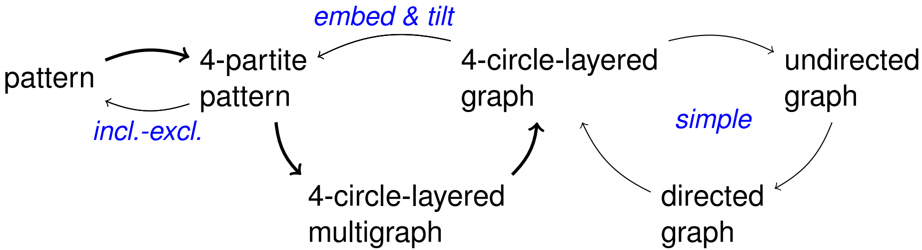


Theorem

Counting 4-cycles in simple graphs in $\mathcal{O}(m^\delta)$ time gives $\tilde{\mathcal{O}}(n^\delta)$-time algorithm for counting non-trivial 4-patterns.

$\implies$ an $\mathcal{O}(n^{1.48})$-time algorithm for $\#_{1324}(\pi)$.

Where are we?

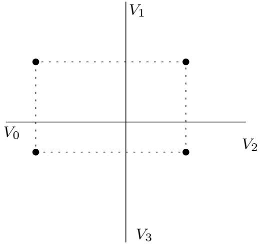


Theorem

Counting 4-cycles in simple graphs in $\mathcal{O}(m^{\delta})$ time gives $\tilde{\mathcal{O}}(n^{\delta})$-time algorithm for counting non-trivial 4-patterns.

$\implies$ an $\mathcal{O}(n^{1.48})$-time algorithm for $\#_{1324}(\pi)$.

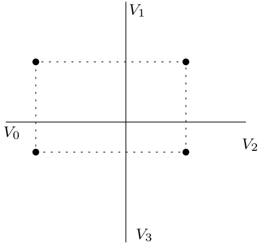
From graph to permutation



1. Embed 4-circle-layered graph in the plane
   $\implies$ every 4-cycle corresponds to a rectangle
2. Tilt each quadrant separately
   $\implies$ every 4-cycle corresponds to an occurrence of 1324_4
3. Make sure coordinates are distinct and subtract extra occurrences of 1324_4

$\implies$ $\mathcal{O}(1)$ instances of counting $\#_{1324}(\pi)$!

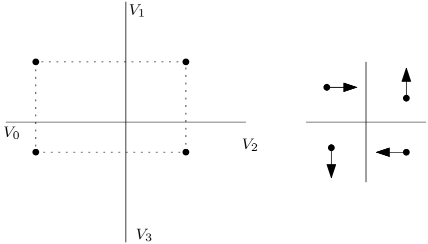
From graph to permutation



1. Embed 4-circle-layered graph in the plane
   $\implies$ every 4-cycle corresponds to a rectangle
2. Tilt each quadrant separately
   $\implies$ every 4-cycle corresponds to an occurrence of 1324_4
3. Make sure coordinates are distinct and subtract extra occurrences of 1324_4

$\implies \mathcal{O}(1)$ instances of counting $\#_{1324}(\pi)$!

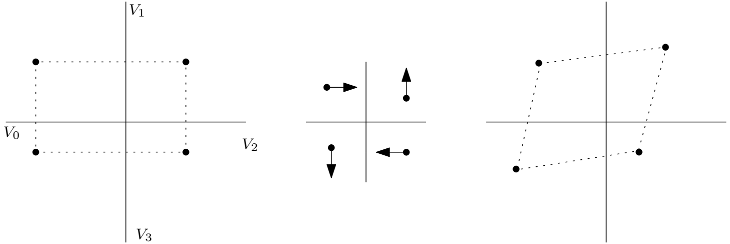
From graph to permutation



1. Embed 4-circle-layered graph in the plane
   $\implies$ every 4-cycle corresponds to a rectangle
2. Tilt each quadrant separately
   $\implies$ every 4-cycle corresponds to an occurrence of 1324_4
3. Make sure coordinates are distinct and subtract extra occurrences of 1324_4

$\implies \mathcal{O}(1)$ instances of counting $\#_{1324}(\pi)$!

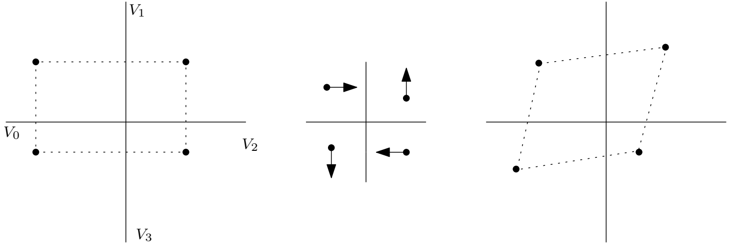
From graph to permutation



1. Embed 4-circle-layered graph in the plane
   $\implies$ every 4-cycle corresponds to a rectangle
2. Tilt each quadrant separately
   $\implies$ every 4-cycle corresponds to an occurrence of 1324_4
3. Make sure coordinates are distinct and subtract extra occurrences of 1324_4

$\implies$ $\mathcal{O}(1)$ instances of counting $\#_{1324}(\pi)$!

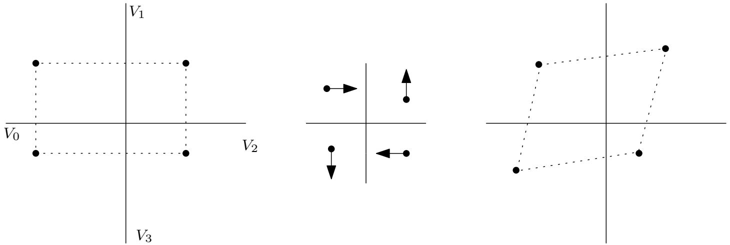
From graph to permutation



1. Embed 4-circle-layered graph in the plane
   $\implies$ every 4-cycle corresponds to a rectangle
2. Tilt each quadrant separately
   $\implies$ every 4-cycle corresponds to an occurrence of 1324_4
3. Make sure coordinates are distinct and subtract extra occurrences of 1324_4

$\implies \mathcal{O}(1)$ instances of counting $\#_{1324}(\pi)$!

From graph to permutation



1. Embed 4-circle-layered graph in the plane
   $\implies$ every 4-cycle corresponds to a rectangle
2. Tilt each quadrant separately
   $\implies$ every 4-cycle corresponds to an occurrence of 1324_4
3. Make sure coordinates are distinct and subtract extra occurrences of 1324_4

$\implies$ $\mathcal{O}(1)$ instances of counting $\#_{1324}(\pi)$!

Summary

Theorem

An $\mathcal{O}(n^\delta)$-time algorithm for counting 4-patterns gives $\mathcal{O}(m^\delta)$-time algorithm for counting 4-cycles in simple graphs.

Recall:

Conjecture [Dahlgaard et al., STOC'17]

For every $\varepsilon > 0$ no algorithm detects 4-cycles in $\mathcal{O}(m^{4/3-\varepsilon})$ time.

$\Longrightarrow$ probably no $\mathcal{O}(n^{4/3-\varepsilon})$-time algorithm for $\#_{1324}(\pi)$.

Summary

Theorem

An $\mathcal{O}(n^\delta)$-time algorithm for counting 4-patterns gives $\mathcal{O}(m^\delta)$-time algorithm for counting 4-cycles in simple graphs.

Recall:

Conjecture [Dahlgaard et al., STOC'17]

For every $\varepsilon > 0$ no algorithm detects 4-cycles in $\mathcal{O}(m^{4/3-\varepsilon})$ time.

$\implies$ probably no $\mathcal{O}(n^{4/3-\varepsilon})$-time algorithm for $\#_{1324}(\pi)$.

Summary

Theorem

An $\mathcal{O}(n^\delta)$-time algorithm for counting 4-patterns gives $\mathcal{O}(m^\delta)$-time algorithm for counting 4-cycles in simple graphs.

Recall:

Conjecture [Dahlgaard et al., STOC'17]

For every $\varepsilon > 0$ no algorithm detects 4-cycles in $\mathcal{O}(m^{4/3-\varepsilon})$ time.

$\Longrightarrow$ probably no $\mathcal{O}(n^{4/3-\varepsilon})$-time algorithm for $\#_{1324}(\pi)$.

Open questions

Berendsohn et al. left the following questions open:

1. Can we beat $n^{k/4+o(k)}$ for counting k-patterns?
2. Is there a better (conditional?) lower bound than $f(k)n^{o(k/\log k)}$?

Thank you!

Open questions

Berendsohn et al. left the following questions open:

1. Can we beat $n^{k/4+o(k)}$ for counting k-patterns?
2. Is there a better (conditional?) lower bound than $f(k)n^{o(k/\log k)}$?

Thank you!

Open questions

Berendsohn et al. left the following questions open:

1. Can we beat $n^{k/4+o(k)}$ for counting k-patterns?
2. Is there a better (conditional?) lower bound than $f(k)n^{o(k/\log k)}$?

Thank you!