One-Face Shortest Disjoint Paths with a Deviation Terminal

Yusuke Kobayashi, **Tatsuya Terao**
Kyoto University

ISAAC 2022 @Seoul Dec 20, 2022

Outline

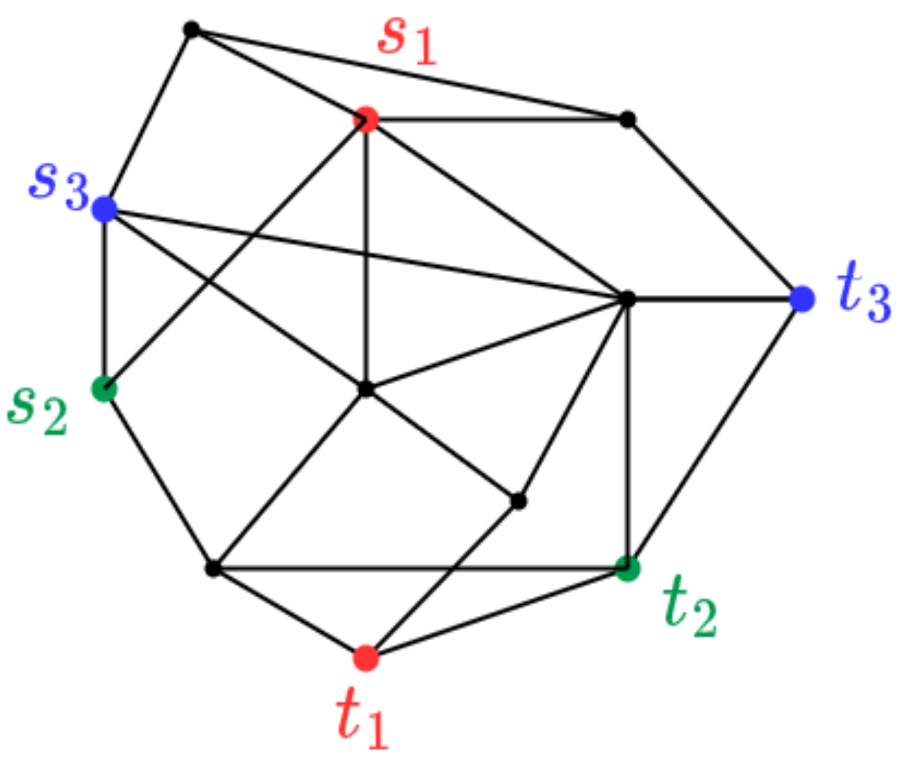
- **Preliminaries**
  - Disjoint Paths
  - Shortest Disjoint Paths
  - One-Face Shortest Disjoint Paths
- **Result**
  - One-Face Shortest Disjoint Paths with a Deviation Terminal
- **Idea**
  - Bijection between $(A+B)$-Paths and Pairing of Terminals
- **Conclusion**

Outline

- **Preliminaries**
  - Disjoint Paths
  - Shortest Disjoint Paths
  - One-Face Shortest Disjoint Paths
- **Result**
  - One-Face Shortest Disjoint Paths with a Deviation Terminal
- **Idea**
  - Bijection between $(A+B)$-Paths and Pairing of Terminals
- **Conclusion**

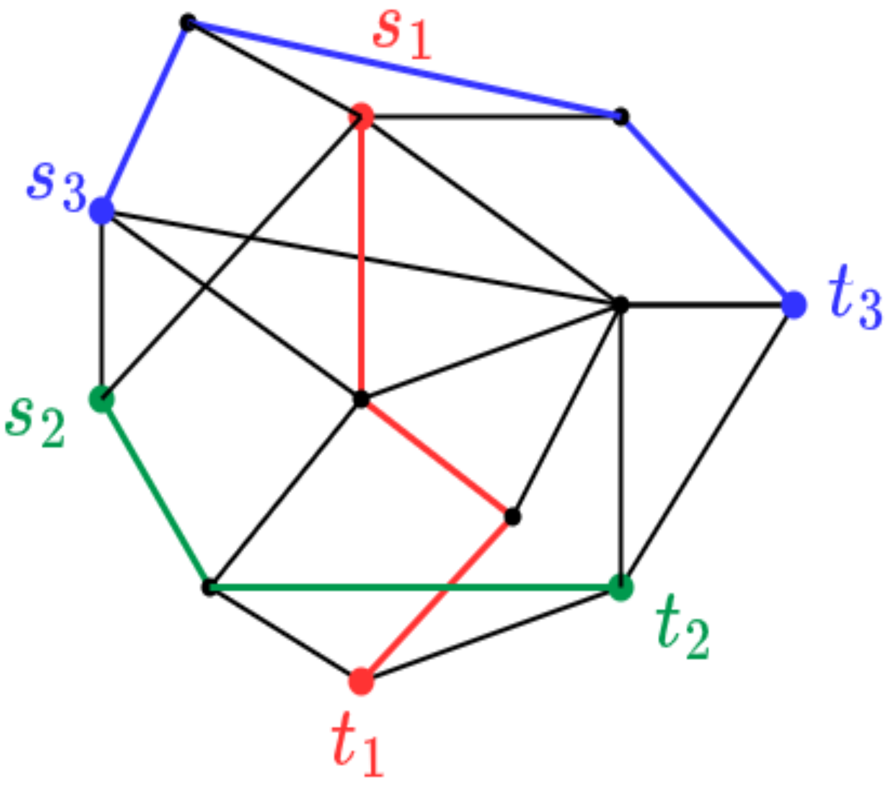
Disjoint Paths Problem

<u>Input</u> : **vertex pairs** $(s_1, t_1), \ldots, (s_k, t_k)$



Disjoint Paths Problem

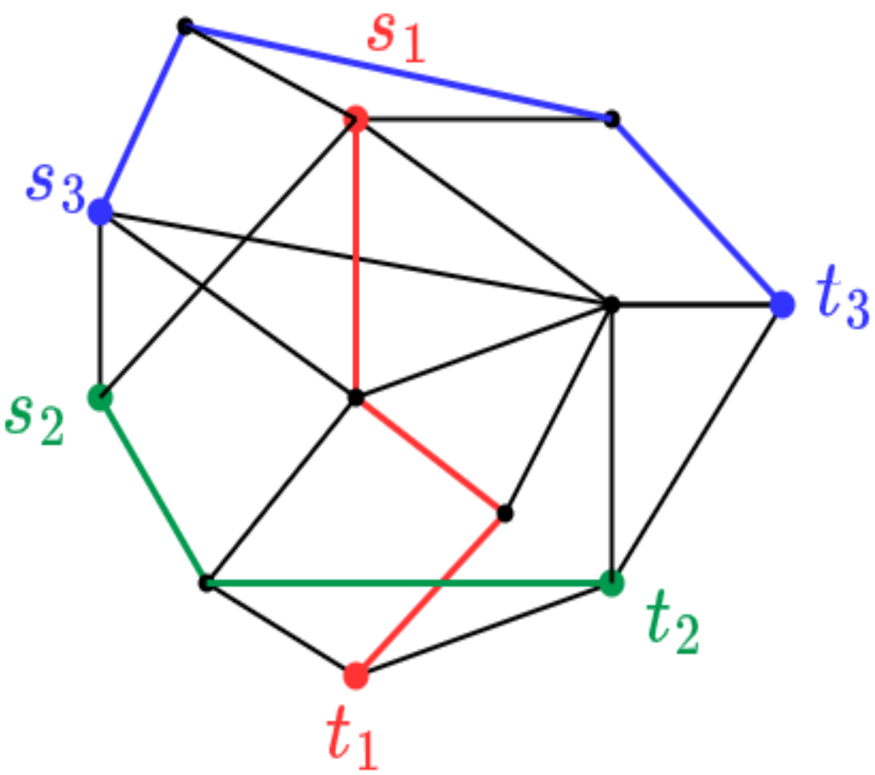
Input : vertex pairs $(s_1, t_1), \ldots, (s_k, t_k)$
Find : vertex-disjoint paths $P_1, \ldots, P_k \, (P_i : s_i \to t_i)$



- **Many Applications**
  ex : VLSI-design, network routing (1980s)

Disjoint Paths Problem

Input : vertex pairs $(s_1, t_1), \ldots, (s_k, t_k)$
Find : vertex-disjoint paths $P_1, \ldots, P_k \; (P_i : s_i \to t_i)$



- **Many Applications**
  ex : VLSI-design, network routing (1980s)

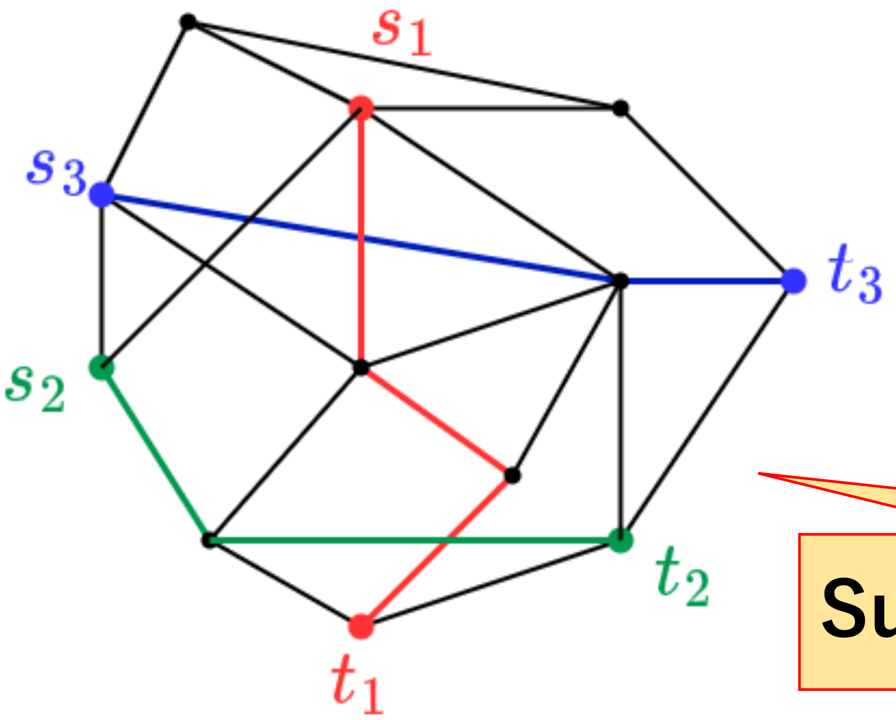
	Directed	Undirected
k : fixed	NP-hard (Fortune et al. 1980)	Polytime (Robertson & Seymour 1995)
k : general	NP-hard (Karp 1975)	NP-hard (Karp 1975)

Shortest Disjoint Paths Problem

Input : vertex pairs $(s_1, t_1), \ldots, (s_k, t_k)$

Find : vertex-disjoint paths $P_1, \ldots, P_k \ (P_i : s_i \to t_i)$

minimizing sum of their length



- natural optimization version
- k : fixed , undirected

polynomial solvability is widely open

Sum of their length: 3 + 2 + 2 = 7

Polynomially solvable cases (1)

- $k = 2$: **Randomized Polytime algorithm**

  (Björklund & Husfeldt 2014)

  Permanent modulo 4

- $k = 2$, **cubic, planar : Deterministic Polytime algorithm**

  (Björklund & Husfeldt 2018)

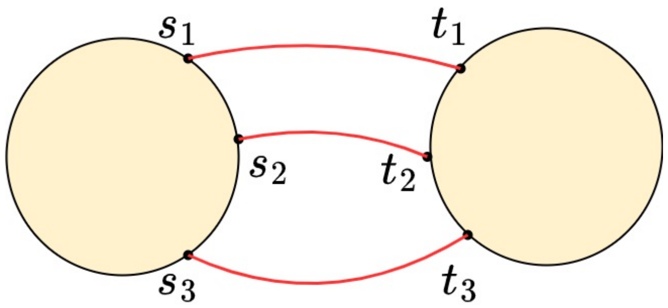
  Pfaffian

Idea : algebraic approach via polynomial matrix

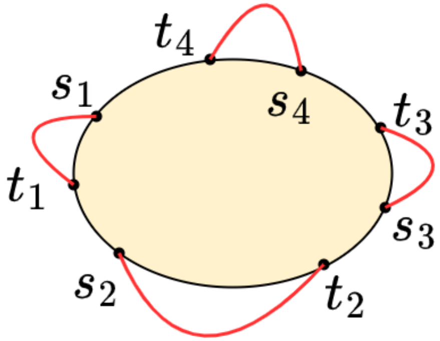
Polynomially solvable cases (2)

planar graph, terminals satisfy certain conditions

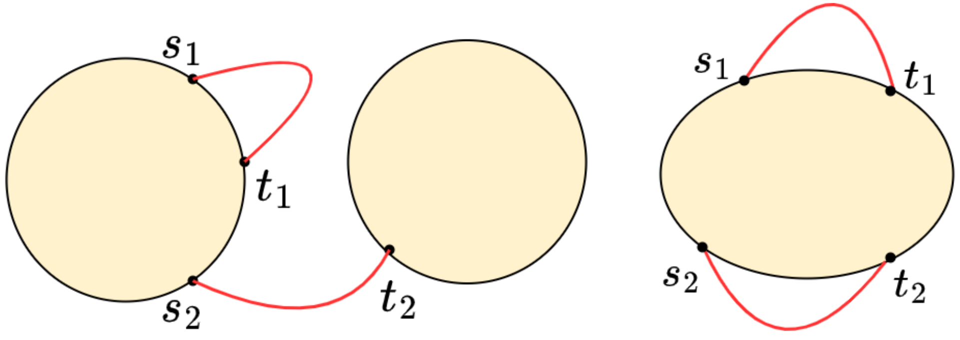
- $s_1, \ldots, s_k$ **are on one face and** $t_1, \ldots, t_k$ **are on another face**
  (Colin de Verdière & Schrijver, 2011)



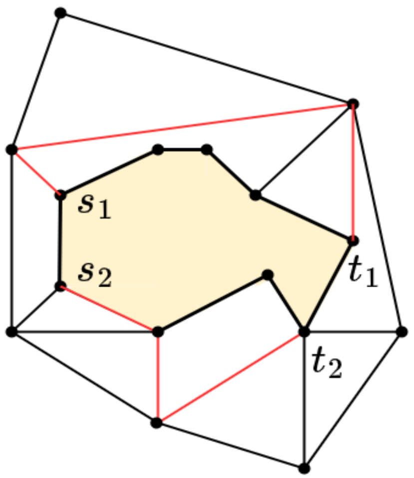
- $s_1, t_1, \ldots, s_k, t_k$ **are on one face in this order**
  (Borradile et al. 2015)



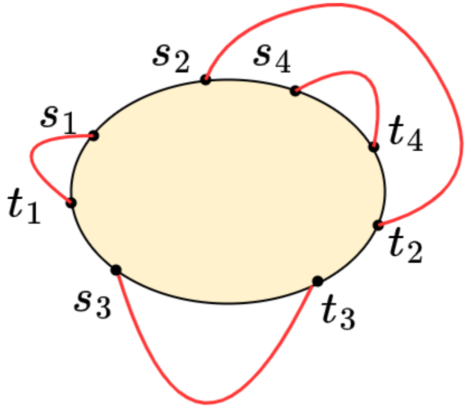
- $k = 2$ **, terminals are on at most 2 faces**
  (Kobayashi & Sommer, 2010)



Interpretation



- **all the terminals are on one face**
  (Datta et al. 2018)



One-Face Shortest Disjoint Paths Problem

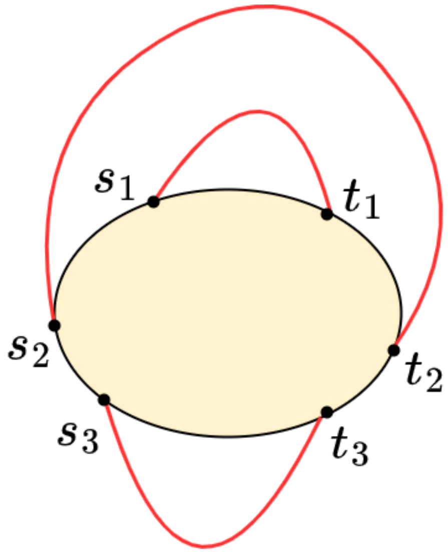
(Datta et al. 2018)

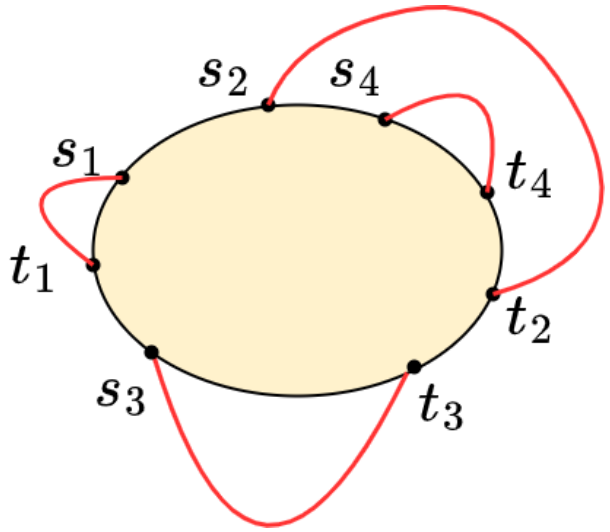
<u>Input</u> : planar graph, vertex pairs $(s_1, t_1), \ldots, (s_k, t_k)$

all the terminals are on the same face

<u>Find</u> : vertex-disjoint paths $P_1, \ldots, P_k \ (P_i : s_i \to t_i)$

minimizing sum of their length





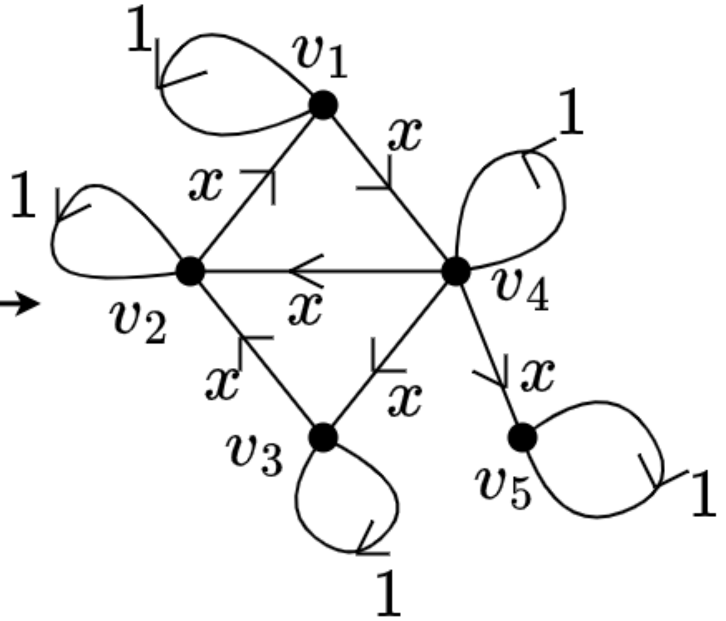
Algorithm for One-Face Shortest Disjoint Paths

(Datta et al. 2018)

Obs.

An expansion term of determinant of adjacency matrix corresponds to a cycle cover of directed graph

$$\det A[x] = \det \begin{bmatrix} 1 & & & x & \\ x & 1 & & & \\ & x & 1 & & \\ & x & x & 1 & x \\ & & & & 1 \end{bmatrix} \leftrightarrow$$



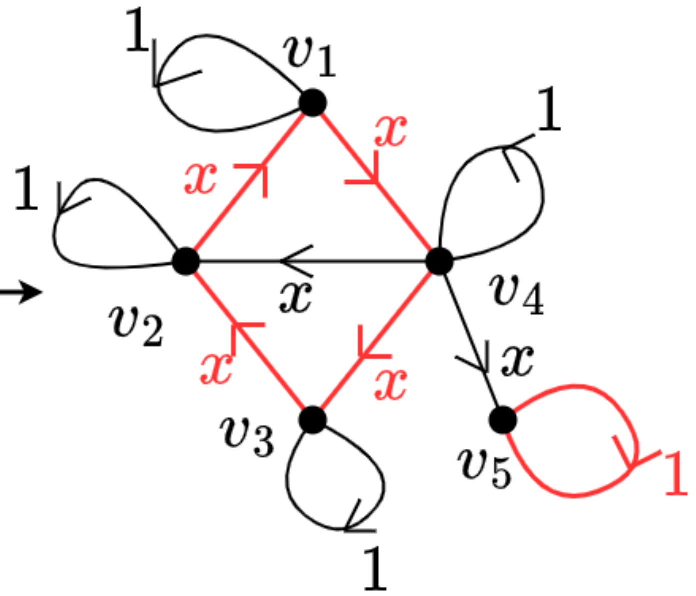
Algorithm for One-Face Shortest Disjoint Paths

(Datta et al. 2018)

Obs.

An expansion term of determinant of adjacency matrix corresponds to a cycle cover of directed graph

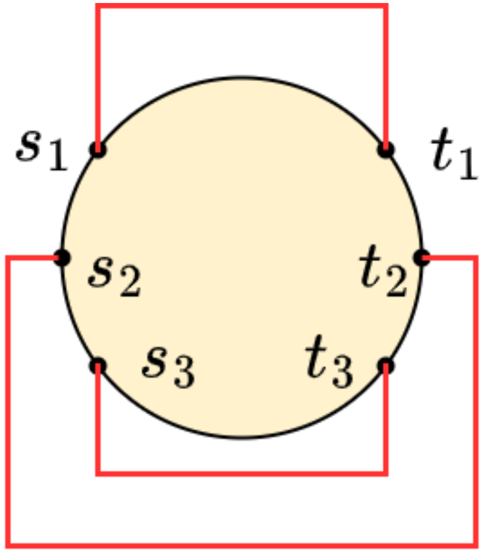
$$\det A[x] = \det \begin{bmatrix} 1 & & & x & \\ x & 1 & & & \\ & x & 1 & & \\ & x & x & 1 & x \\ & & & & 1 \end{bmatrix} \leftrightarrow$$



$$\det A[x] = -x^4 + \cdots$$

Algorithm for One-Face Shortest Disjoint Paths

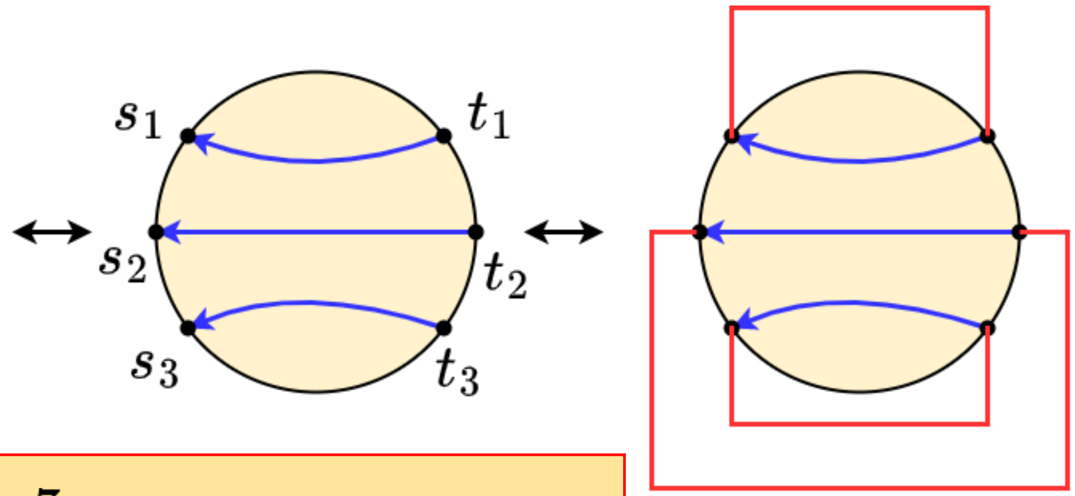
(**Datta et al. 2018**)



Algorithm for One-Face Shortest Disjoint Paths

(Datta et al. 2018)

$\det A_1[x]$



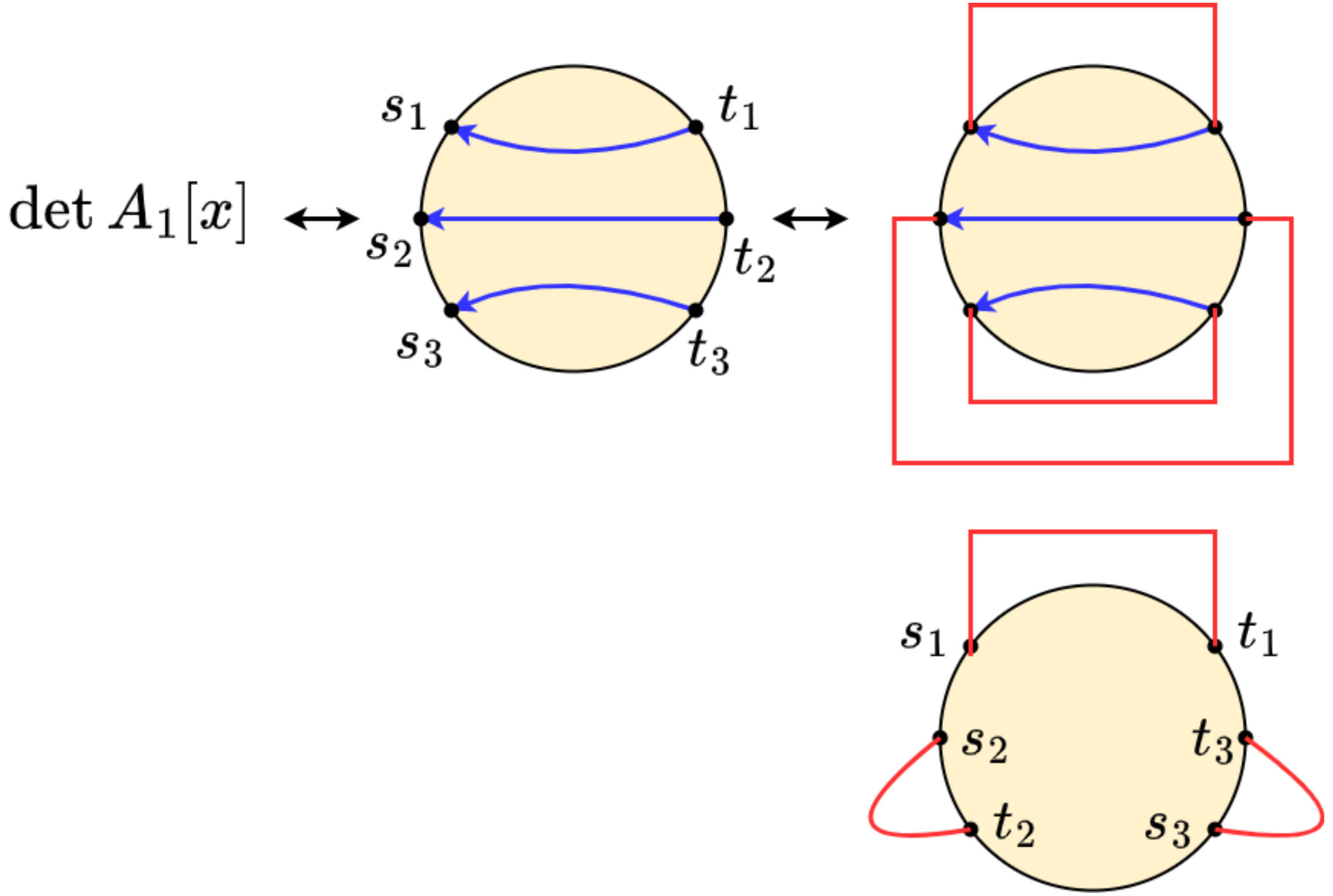
Ex. If $\det = 3x^5 + 4x^7 + \cdots$

- #(disjoint paths of total length 5) = 3
- #(disjoint paths of total length 7) = 4

Determinant ⇔ cycle covers

Algorithm for One-Face Shortest Disjoint Paths

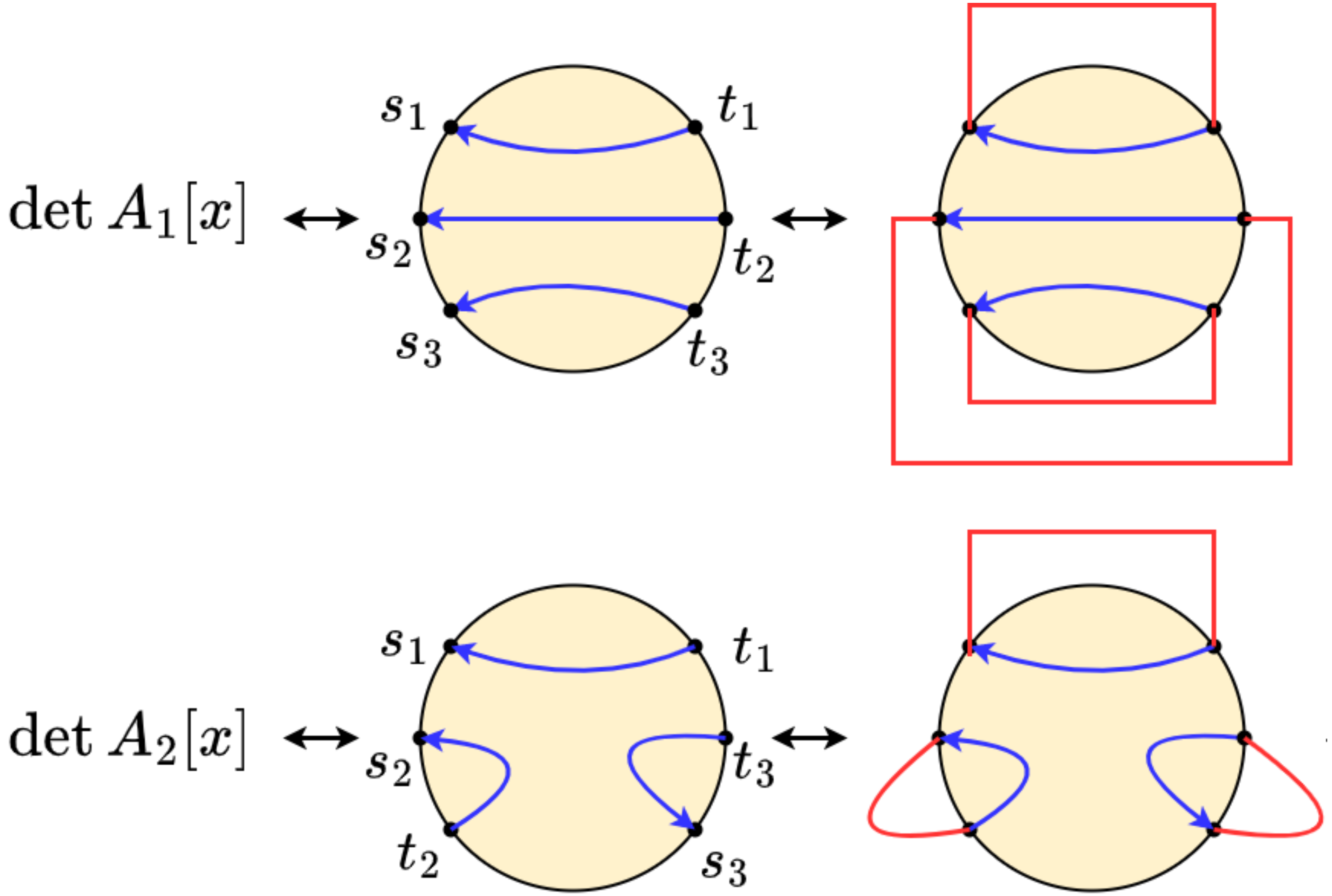
(Datta et al. 2018)



Determinant ⇔ cycle covers

Algorithm for One-Face Shortest Disjoint Paths

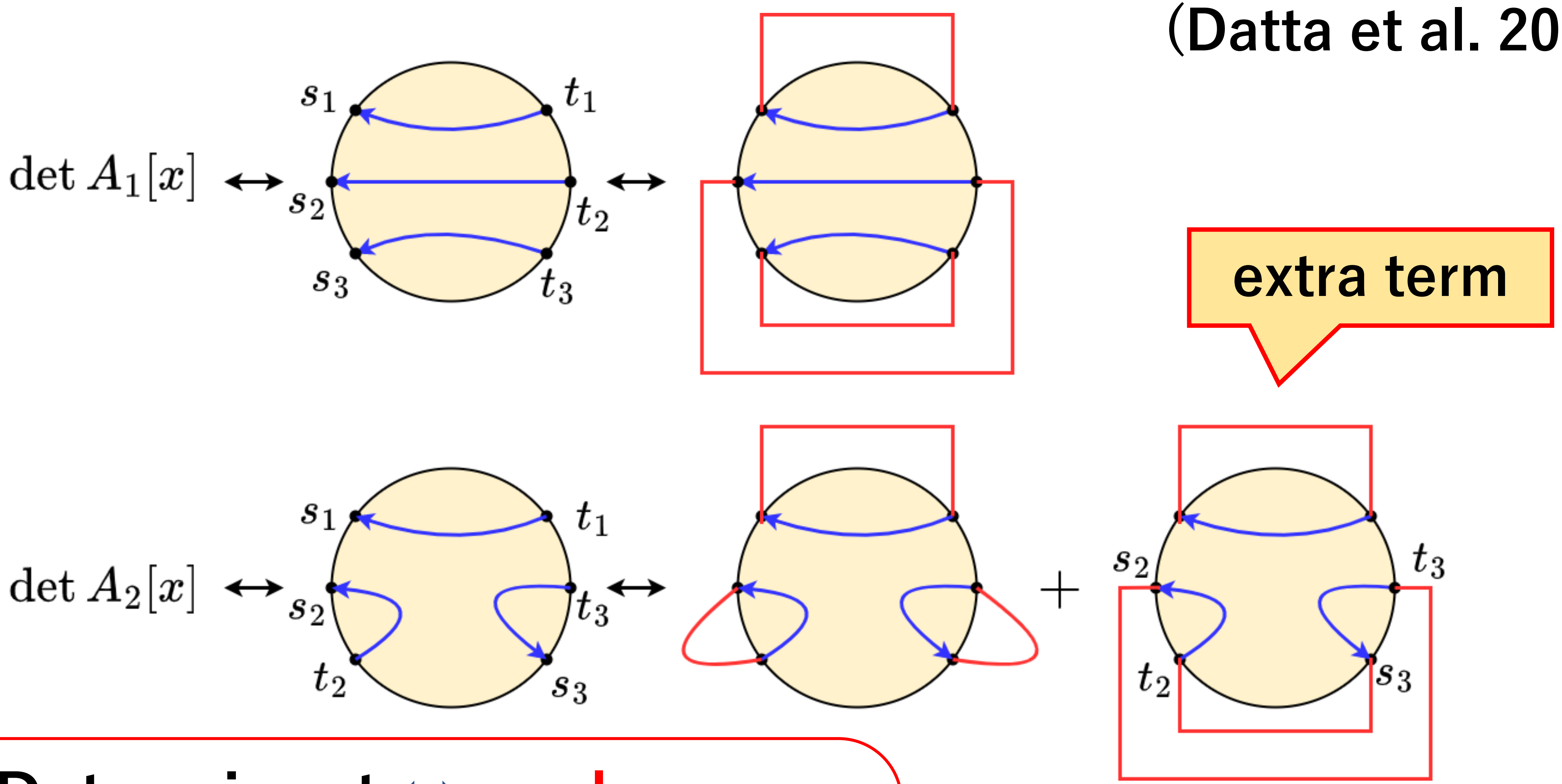
(Datta et al. 2018)



Determinant ⇔ cycle covers

Algorithm for One-Face Shortest Disjoint Paths

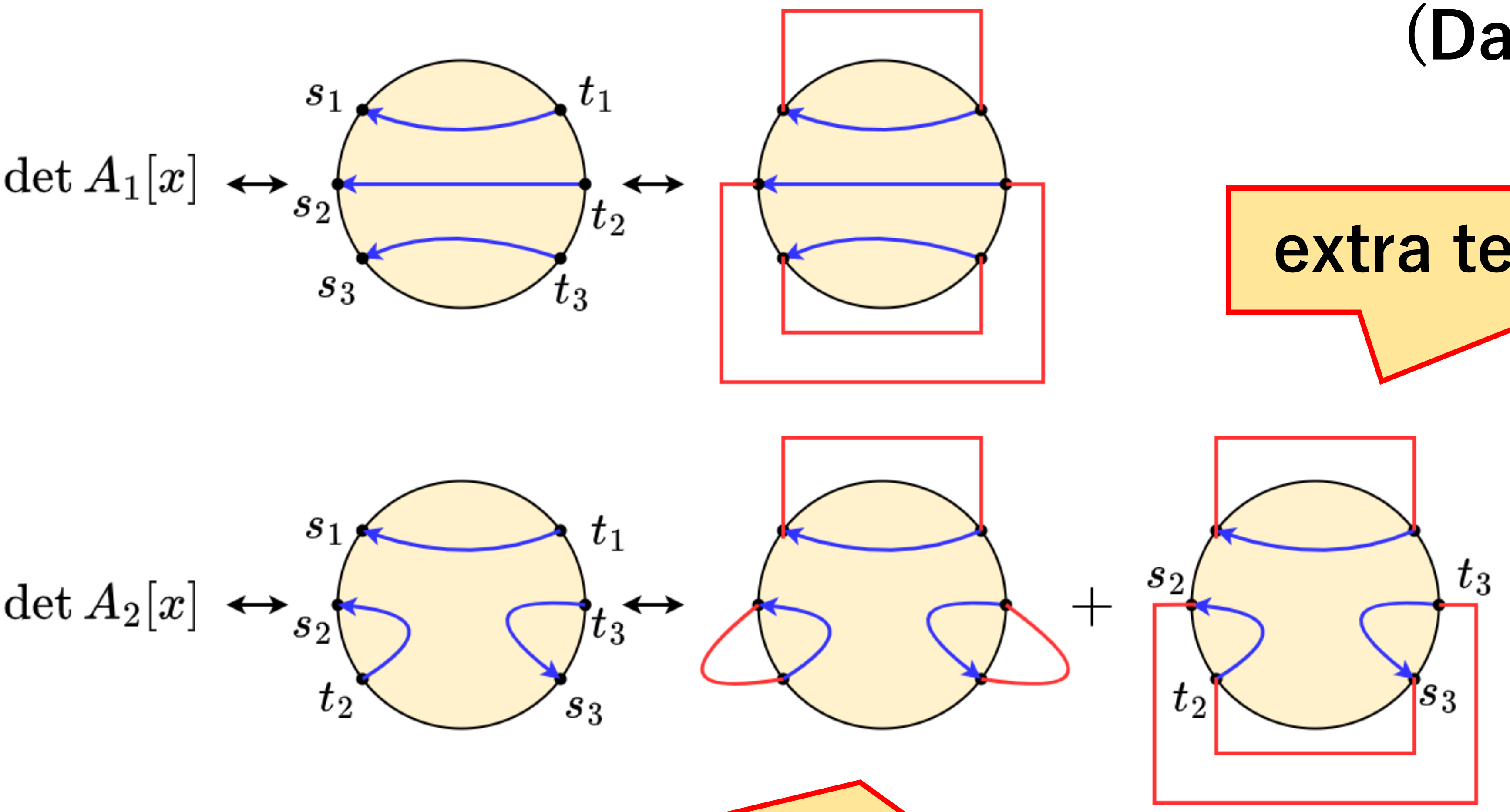
(Datta et al. 2018)



Determinant ↔ cycle covers

Algorithm for One-Face Shortest Disjoint Paths

(Datta et al. 2018)



extra term = $\det A_1[x]$

The lowest degree term of $\det A_2[x] - \det A_1[x]$ corresponds to shortest disjoint paths

Outline

- Preliminaries
  - Disjoint Paths
  - Shortest Disjoint Paths
  - One-Face Shortest Disjoint Paths
- **Result**
  - **One-Face Shortest Disjoint Paths with a Deviation Terminal**
- **Idea**
  - **Bijection between $(A+B)$-Paths and Pairing of Terminals**
- Conclusion

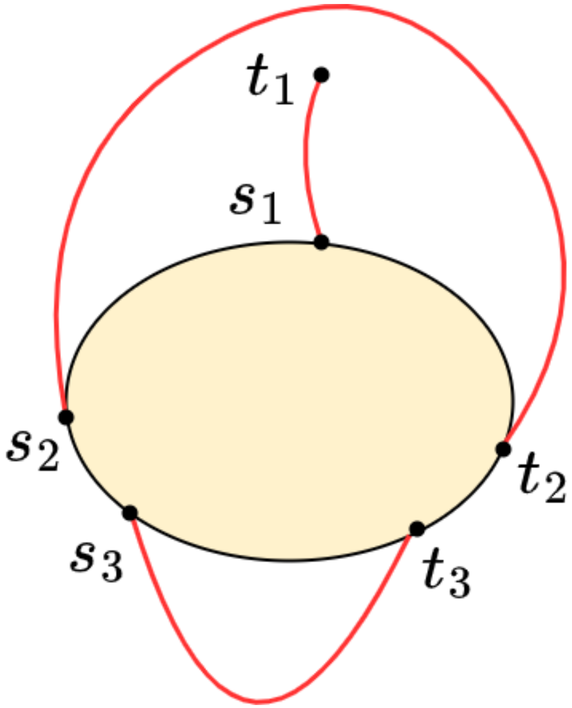
One-Face Shortest Disjoint Paths with a Deviation Terminal

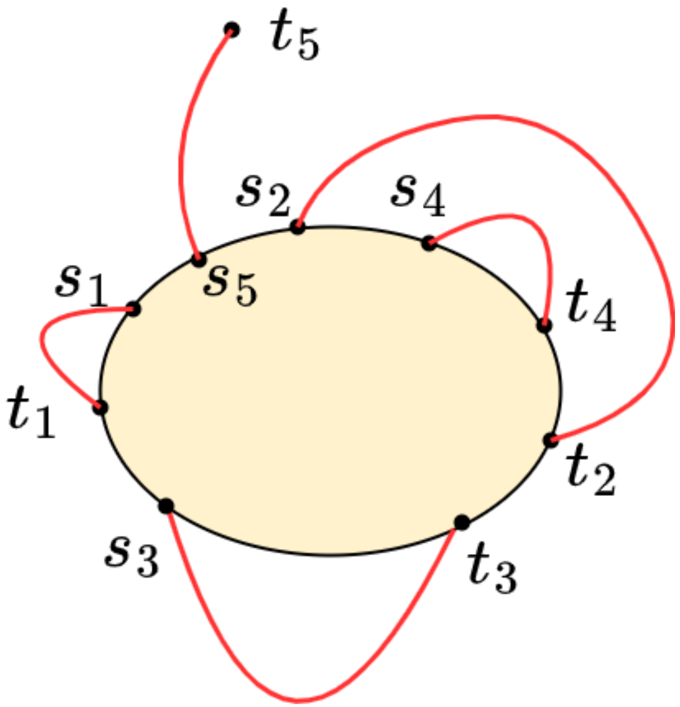
Input : planar graph, vertex pairs $(s_1, t_1), \ldots, (s_k, t_k)$

all the terminals except one are on the same face

Find : vertex-disjoint paths $P_1, \ldots, P_k \ (P_i : s_i \to t_i)$

minimizing sum of their length





Our Contribution

Thm.

k : fixed

One-Face Shortest Disjoint Paths Problem with a Deviation Terminal can be solved by **Randomized Poly.-time Algorithm**

Significance

- extend the case by Datta et al. 2018

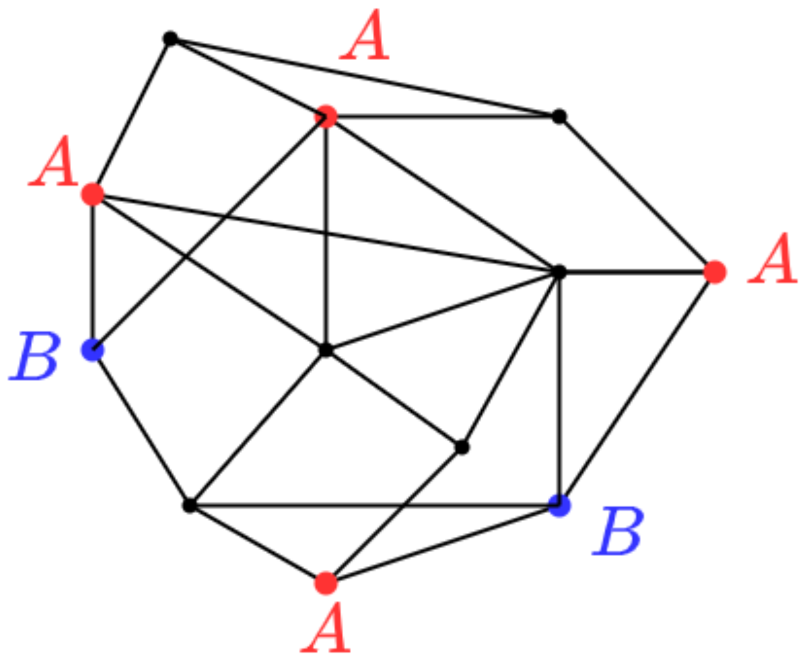
Technique

- One-Face Shortest Disjoint Paths [Datta et al. 2018]
- Shortest Disjoint $(A+B)$-Paths [Hirai & Namba 2018]
- $(A+B)$-Paths ⟺ Pairing of Terminals
  (Insight on combinatorial properties : Related to Catalan Number)

Shortest Disjoint $(A + B)$-Paths Problem

(Hirai & Namba 2018)

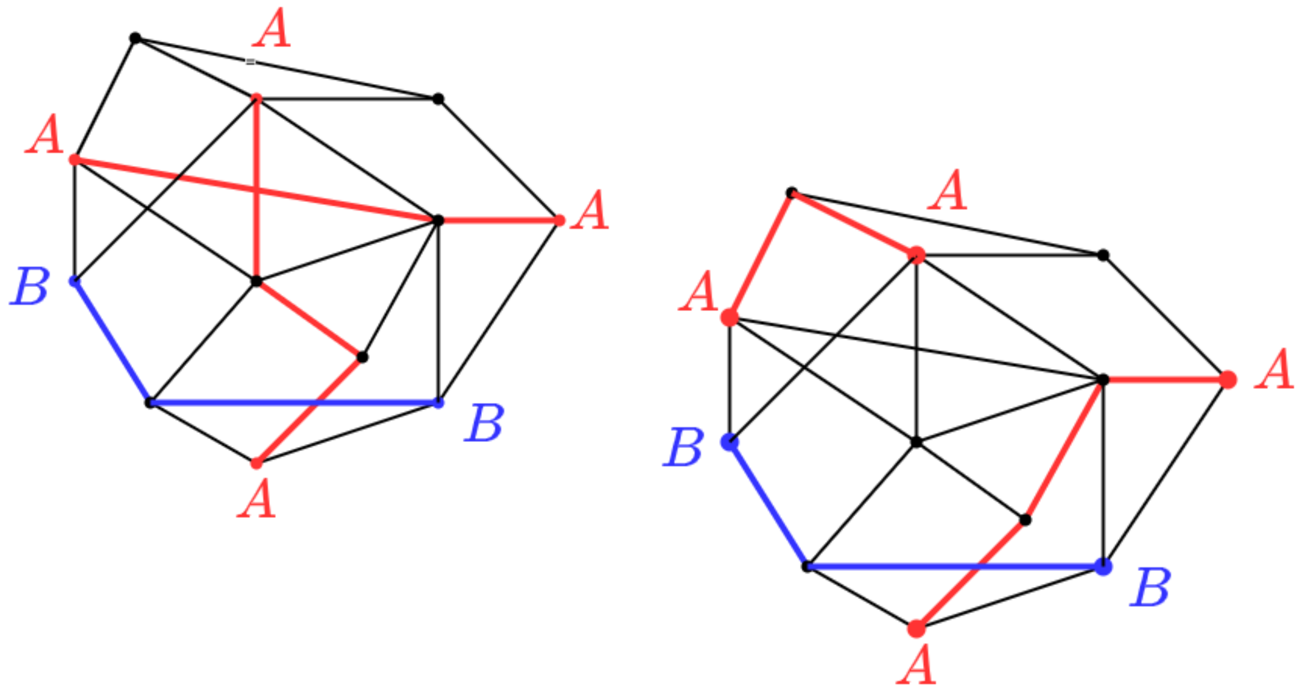
<u>Input</u> : disjoint terminal sets $\boldsymbol{A}, \boldsymbol{B} \subseteq \boldsymbol{V}$ of even size



Shortest Disjoint $(A + B)$-Paths Problem

(Hirai & Namba 2018)

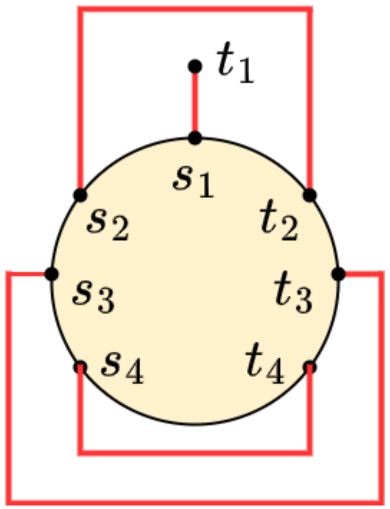
<u>Input</u> : disjoint terminal sets $A, B \subseteq V$ of even size
<u>Find</u> : $\tau = |A|/2 + |B|/2$ vertex-disjoint paths
with endpoints both in A or both in B
minimizing sum of their length



<u>Thm.</u>
A Polynomial corresponding to all disjoint $(A + B)$-paths can be computed in poly.-time.

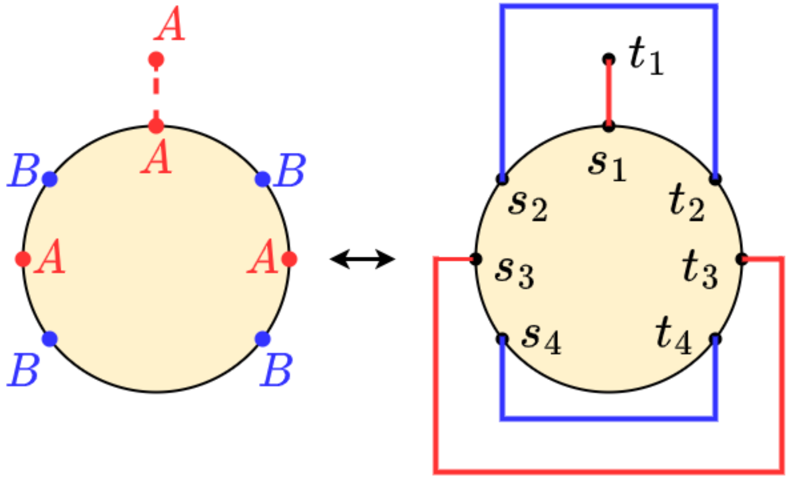
Hafnian modulo $2^{\tau+1}$

Our Algorithm

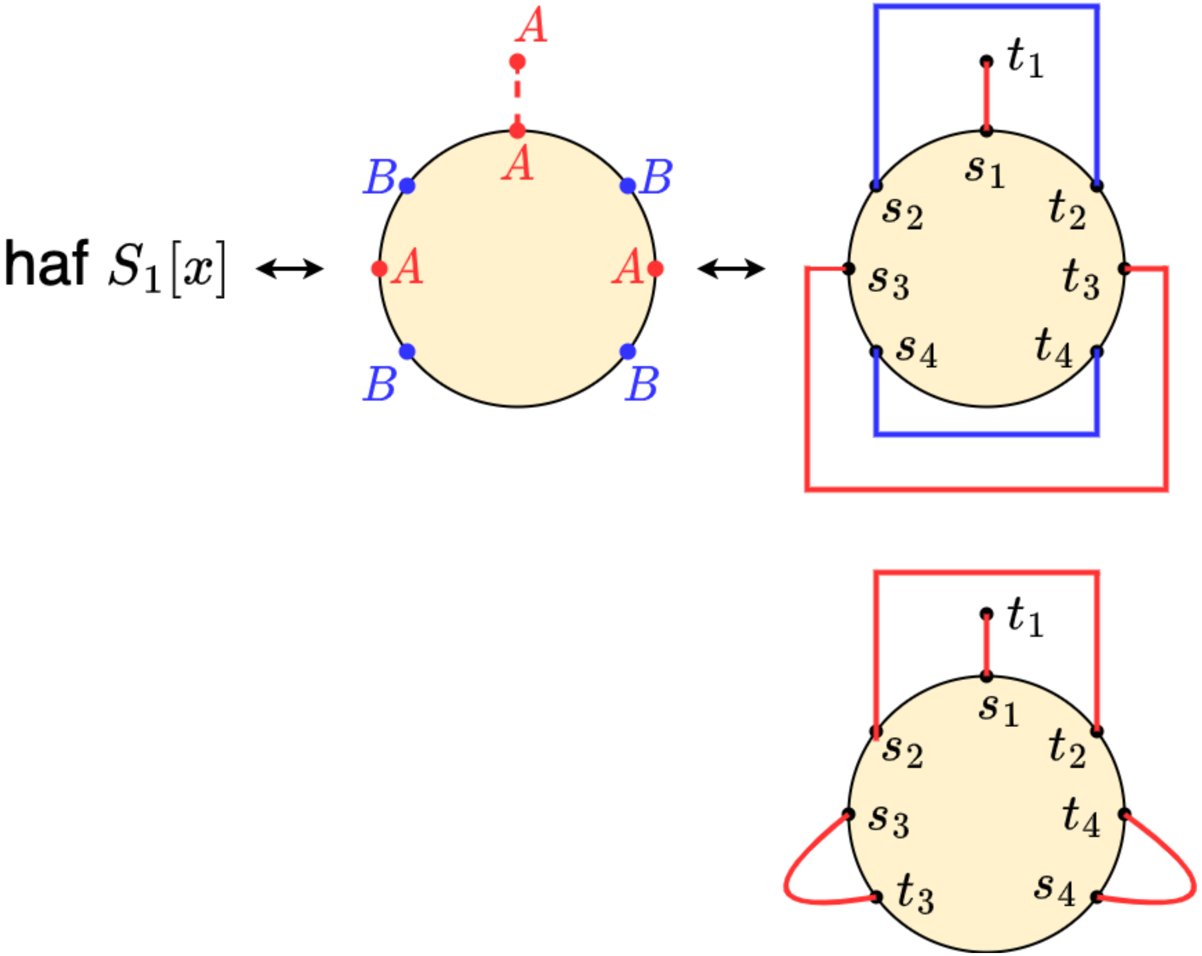


Our Algorithm

haf $S_1[x]$ $\leftrightarrow$



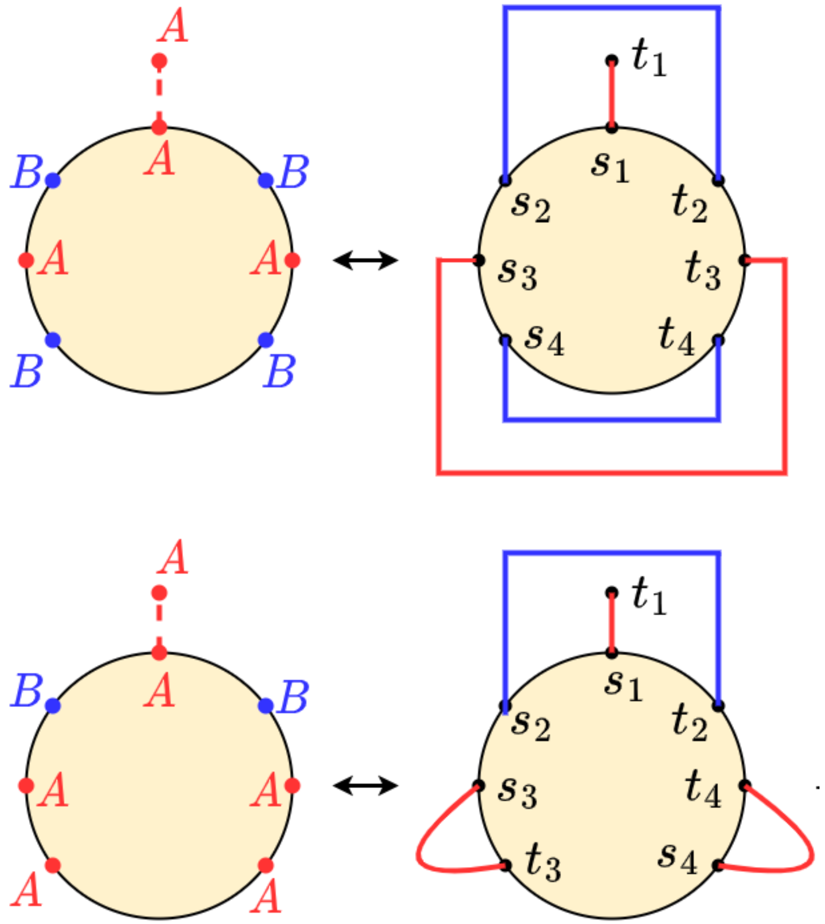
Our Algorithm



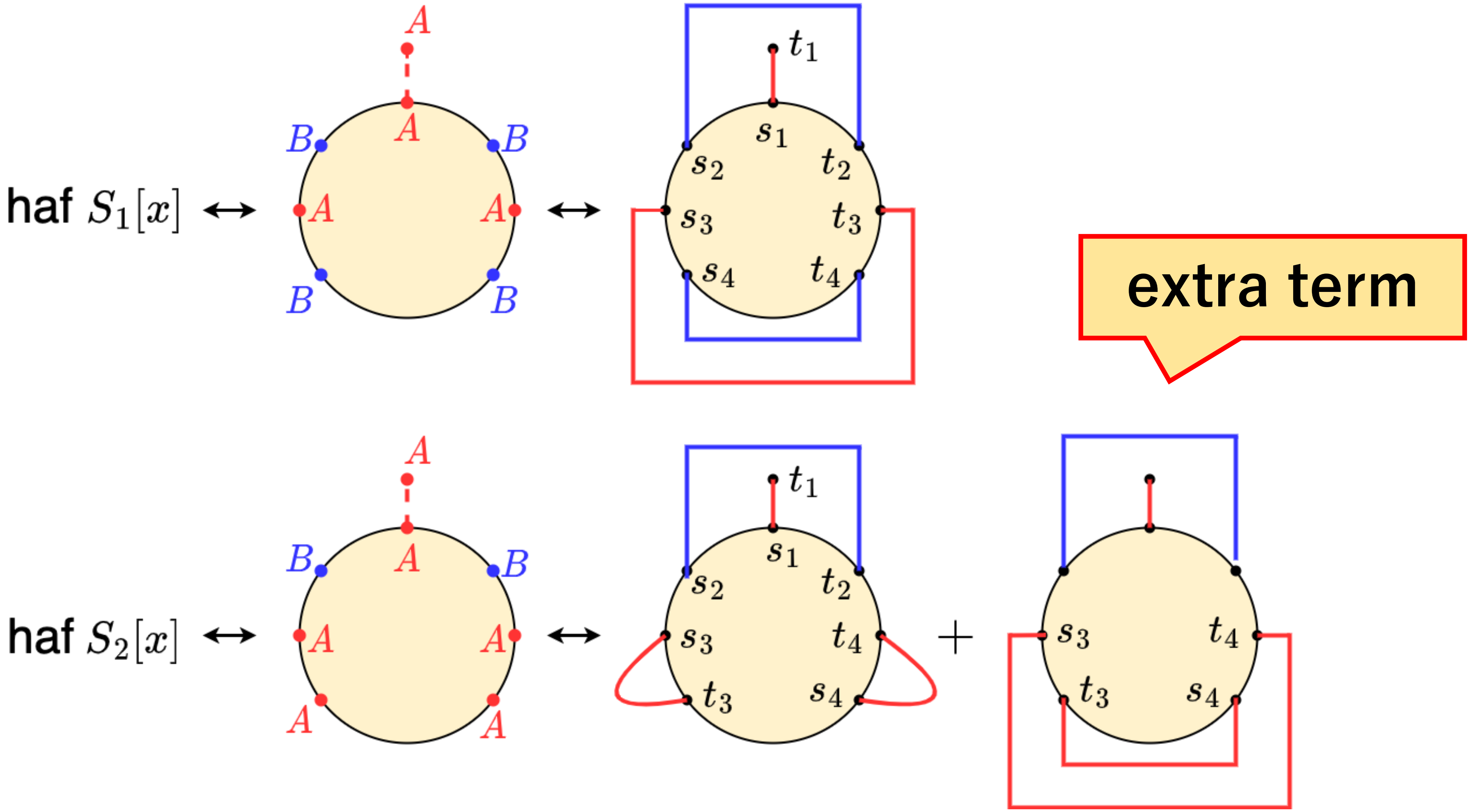
Our Algorithm

haf $S_1[x] \leftrightarrow$

haf $S_2[x] \leftrightarrow$

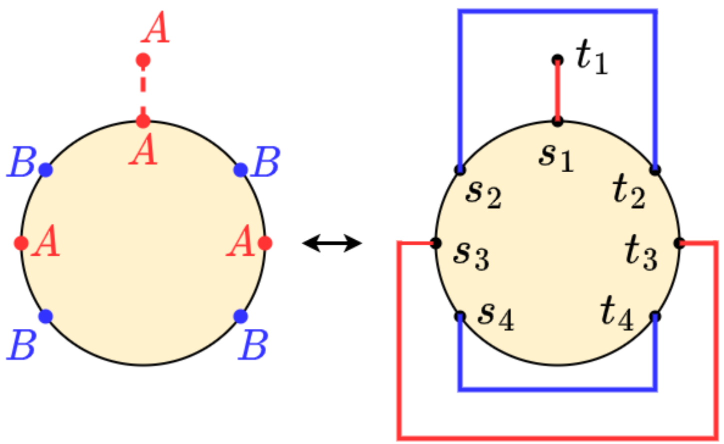


Our Algorithm



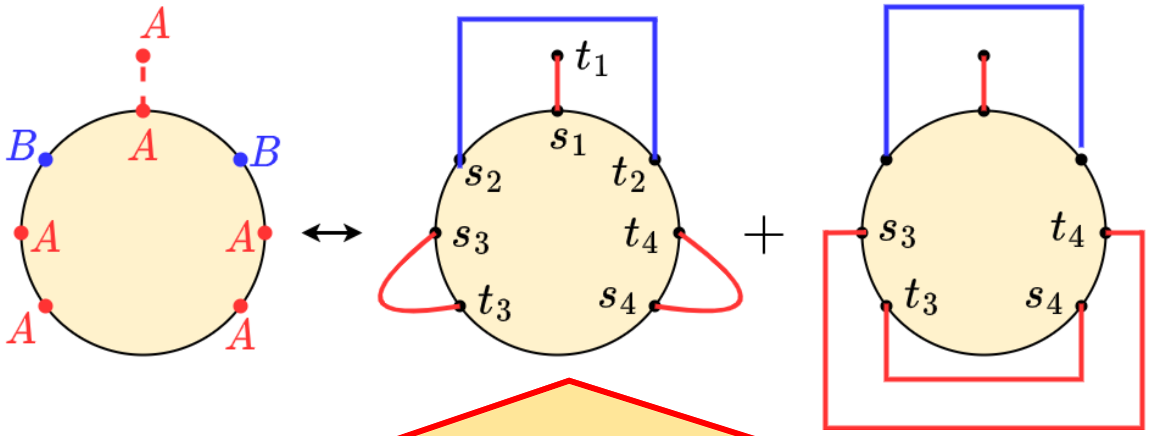
Our Algorithm

haf $S_1[x]$ ↔



extra term = haf $S_1[x]$

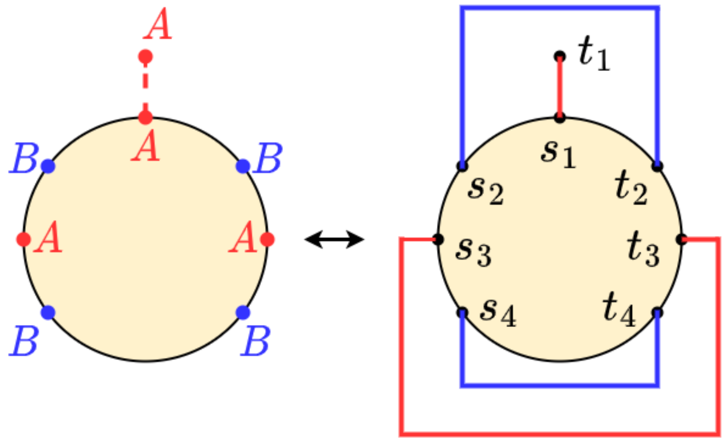
haf $S_2[x]$ ↔



The lowest degree term of haf $S_2[x]$ − haf $S_1[x]$ corresponds to shortest disjoint paths

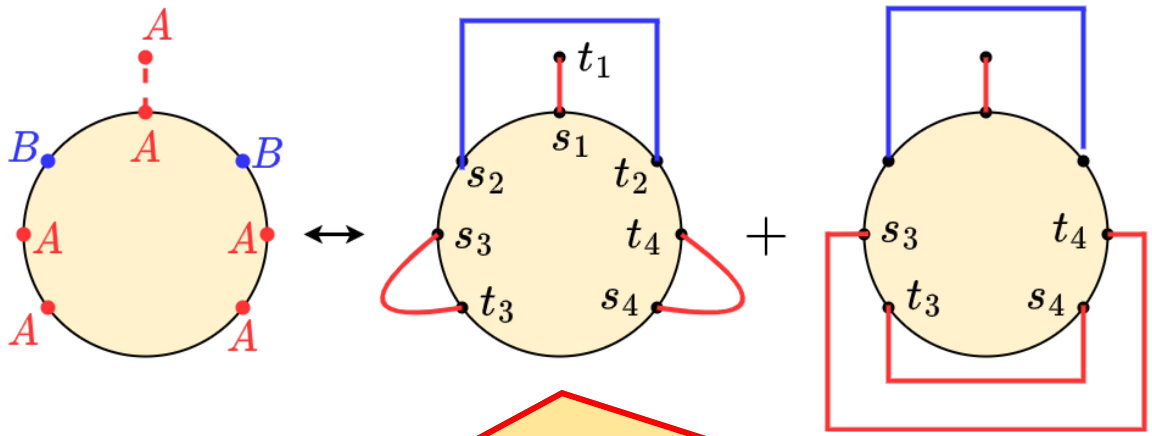
Our Algorithm

haf $S_1[x]$ ↔



extra term = haf $S_1[x]$

haf $S_2[x]$ ↔

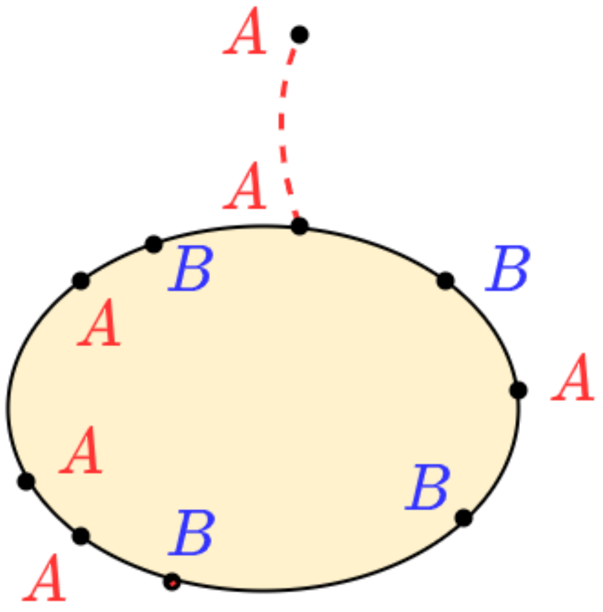


- Hafnian is hard to compute
- But computable modulo 2^{k+1}
- perturb the length of edges (**Randomized**)

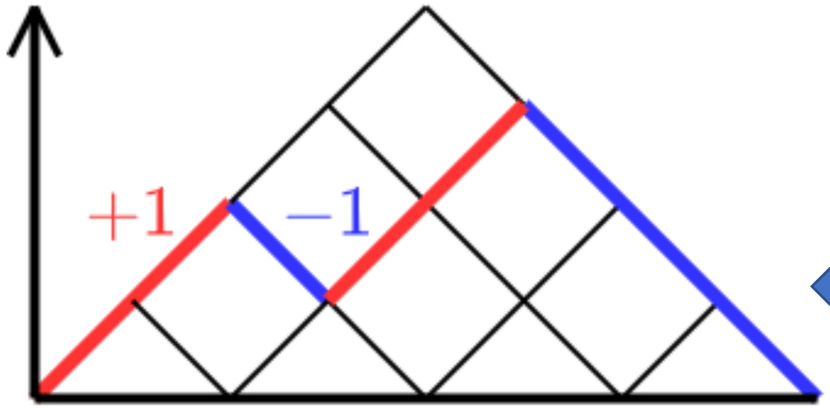
☞ a unique solution

The lowest degree term of haf $S_2[x]$ − haf $S_1[x]$ corresponds to shortest disjoint paths

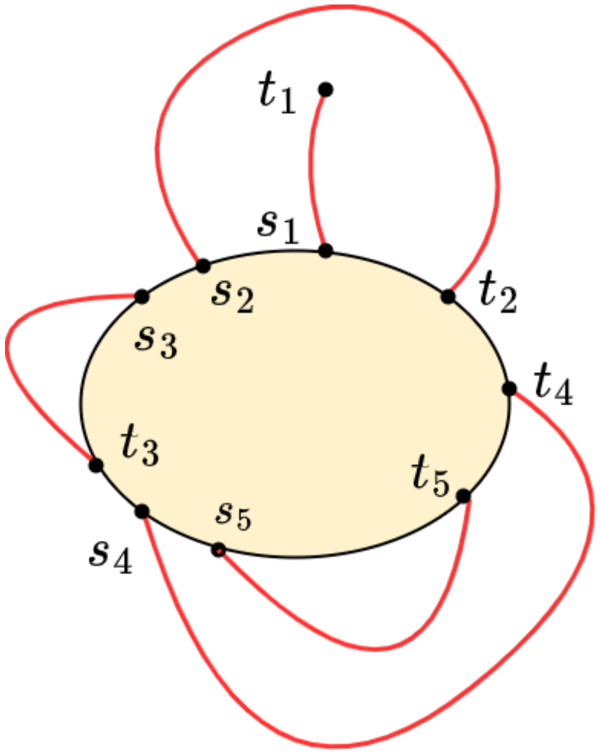
Bijection between $(\mathbf{A}, \mathbf{B})$-Partitions and Pairing of Terminals



Dyck Path

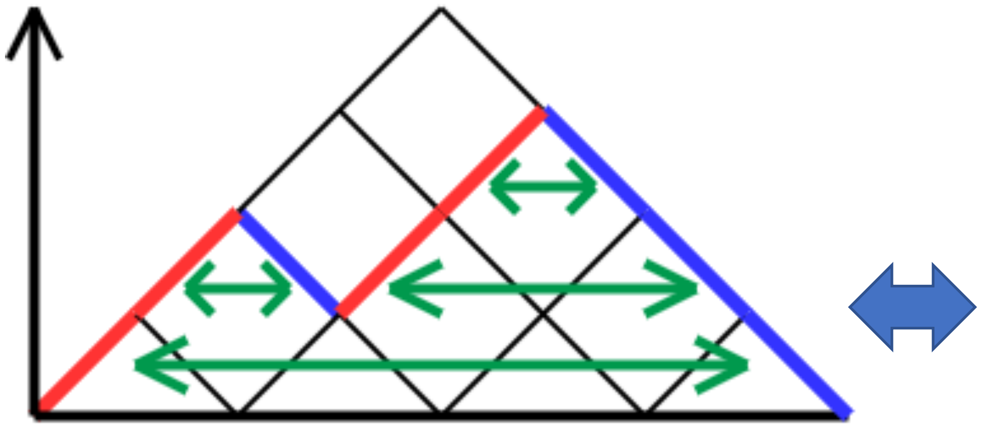


(+1, +1, -1, +1, +1, -1, -1, -1)



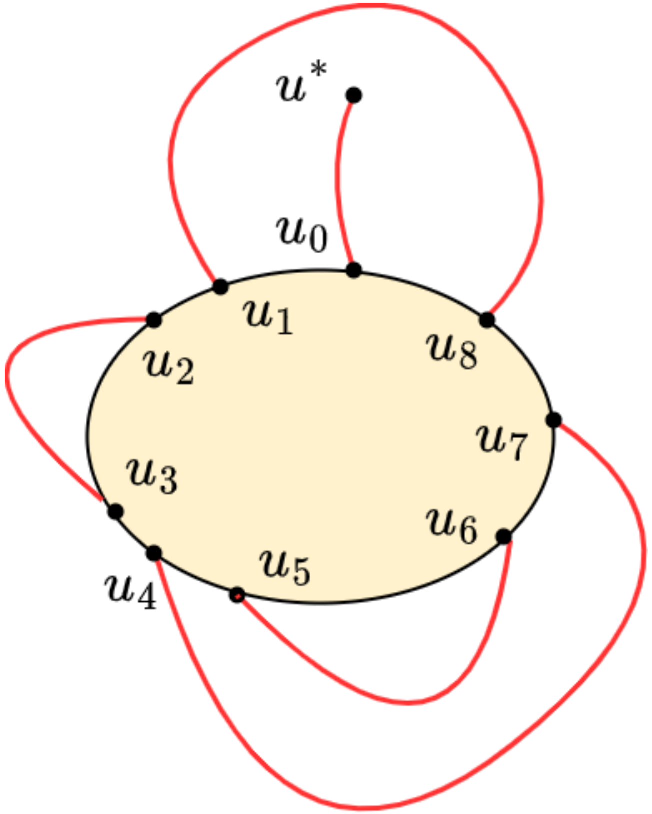
ballot sequences

Bijection between $(\mathbf{A}, \mathbf{B})$-Partitions and Pairing of Terminals

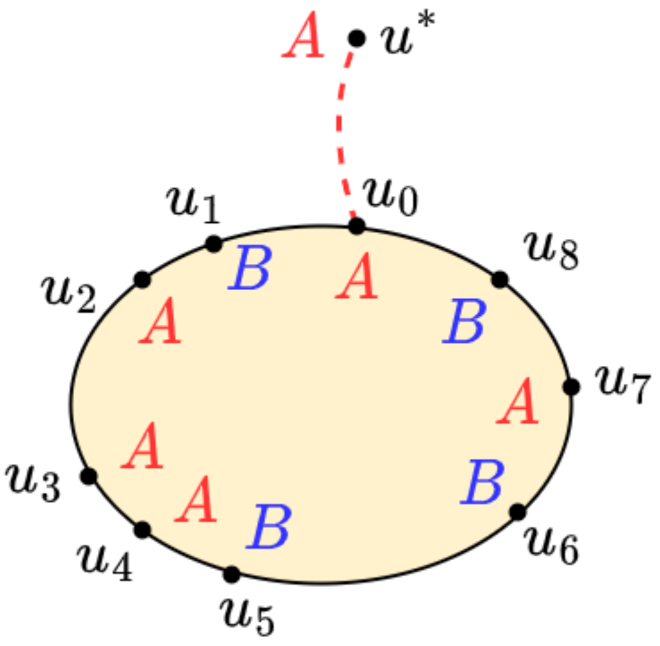


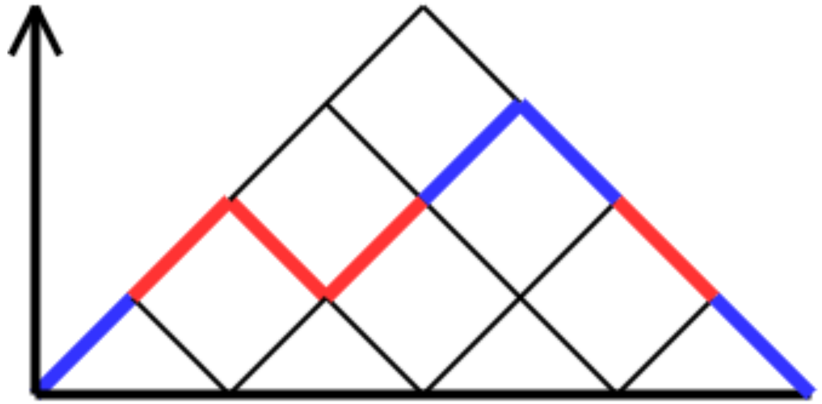
(+1, +1, -1, +1, +1, -1, -1, -1)

(( ) (( )))



Bijection between $(\mathbf{A}, \mathbf{B})$-Partitions and Pairing of Terminals

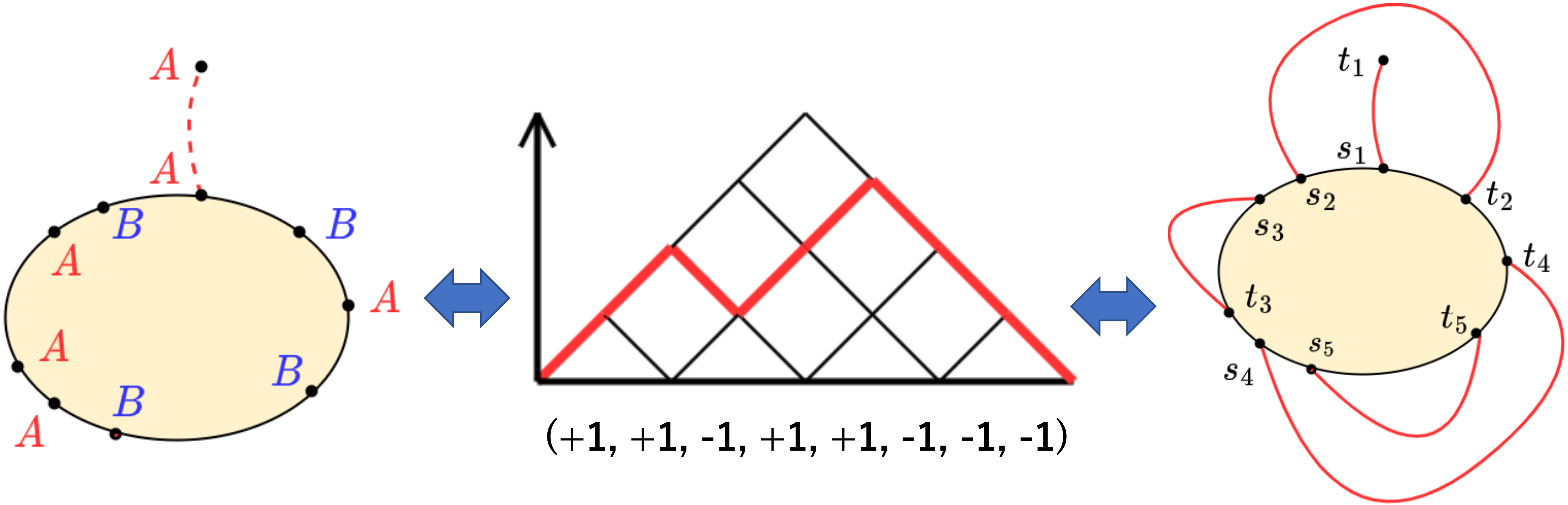




$\boldsymbol{f(i)} = (+1, +1, -1, +1, +1, -1, -1, -1)$

$$u_i \in \begin{cases} A & \text{if } (f(i) = +1 \text{ and } i \text{ is even}) \text{ or} \\ & (f(i) = -1 \text{ and } i \text{ is odd}), \\ B & \text{otherwise.} \end{cases}$$

Bijection between $(\mathbf{A}, \mathbf{B})$-Partitions and Pairing of Terminals



Outline

- Preliminaries
  - Disjoint Paths
  - Shortest Disjoint Paths
  - One-Face Shortest Disjoint Paths
- Result
  - One-Face Shortest Disjoint Paths with a Deviation Terminal
- Idea
  - Bijection between $(A+B)$-Paths and Pairing of Terminals
- **Conclusion**

Conclusion

- introduce **One-Face Shortest Disjoint Paths with a Deviation Terminal**
- present **Randomized Poly.-time algorithm**

- Combination of **One-Face Shortest Disjoint Paths** and **Disjoint $(\boldsymbol{A}+\boldsymbol{B})$-Paths**
- **Combinatorial Insight** on $(\boldsymbol{A}+\boldsymbol{B})$**-Paths** and **Pairing of Terminals**

Q. Deterministic Poly.-time algorithm
Q. All the terminals except two or more are on the same face
Q. The terminals are on two faces