Core-Sparse Monge Matrix Multiplication

Improved Algorithm and Applications

Paweł Gawrychowski, **Egor Gorbachev**, and Tomasz Kociumaka

University of Wrocław, Saarland University, and Max Planck Institute for Informatics

ESA 2025

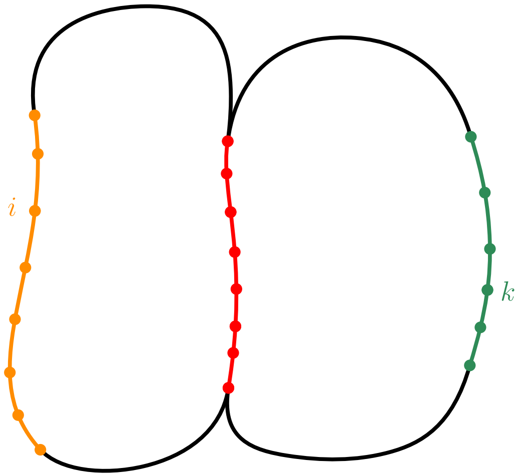
September 17th, 2025, Warsaw

Min-Plus Matrix Product

- Denoted by $C = A \otimes B$.
- Defined as $C_{i,k} = \min_j A_{i,j} + B_{j,k}$.

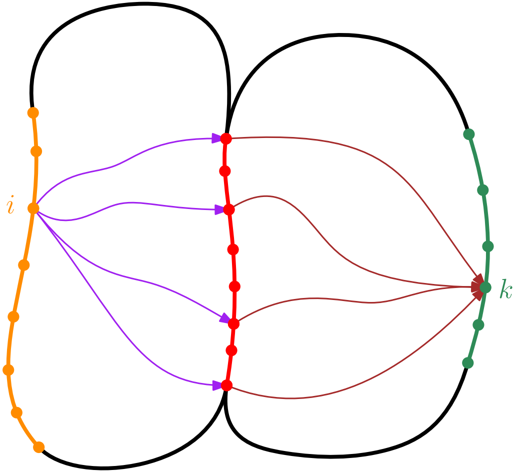
Min-Plus Matrix Product

- Denoted by $C = A \otimes B$.
- Defined as $C_{i,k} = \min_j A_{i,j} + B_{j,k}$.



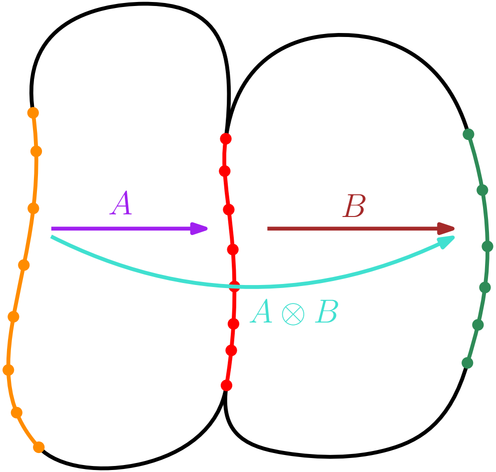
Min-Plus Matrix Product

- Denoted by $C = A \otimes B$.
- Defined as $C_{i,k} = \min_j A_{i,j} + B_{j,k}$.



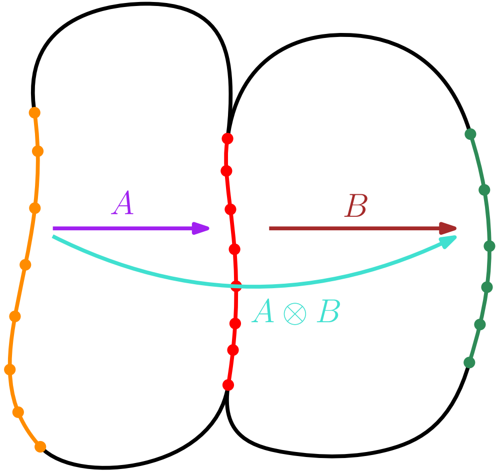
Min-Plus Matrix Product

- Denoted by $C = A \otimes B$.
- Defined as $C_{i,k} = \min_j A_{i,j} + B_{j,k}$.



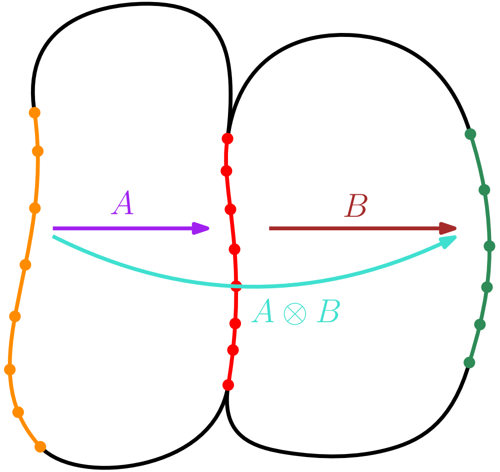
Min-Plus Matrix Product

- Denoted by $C = A \otimes B$.
- Defined as $C_{i,k} = \min_j A_{i,j} + B_{j,k}$.
- Finding the min-plus product of two $n \times n$ matrices is $\Omega(n^{3-o(1)})$-hard assuming the APSP hypothesis.



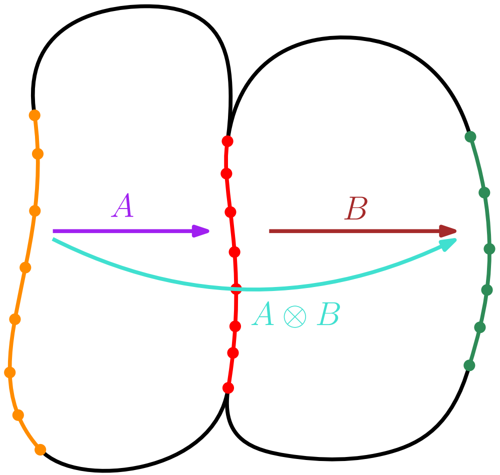
Min-Plus Matrix Product

- Denoted by $C = A \otimes B$.
- Defined as $C_{i,k} = \min_j A_{i,j} + B_{j,k}$.
- Finding the min-plus product of two $n \times n$ matrices is $\Omega(n^{3-o(1)})$-hard assuming the APSP hypothesis.
- Faster algorithms exist for many special cases:

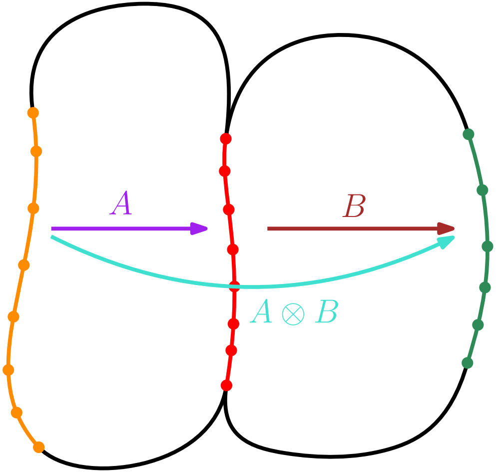


Min-Plus Matrix Product

- Denoted by $C = A \otimes B$.
- Defined as $C_{i,k} = \min_j A_{i,j} + B_{j,k}$.
- Finding the min-plus product of two $n \times n$ matrices is $\Omega(n^{3-o(1)})$-hard assuming the APSP hypothesis.
- Faster algorithms exist for many special cases:
  - Bounded-difference integer-valued matrices;

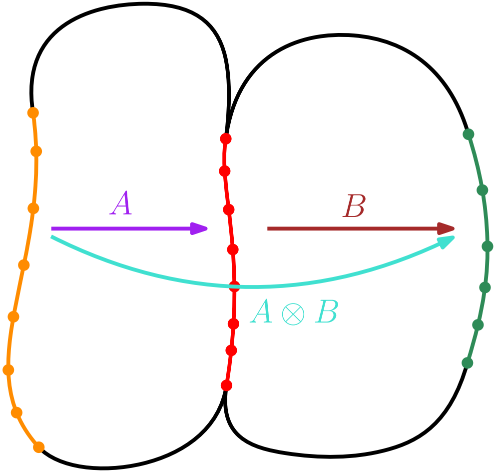


- Denoted by $C = A \otimes B$.
- Defined as $C_{i,k} = \min_j A_{i,j} + B_{j,k}$.
- Finding the min-plus product of two $n \times n$ matrices is $\Omega(n^{3-o(1)})$-hard assuming the APSP hypothesis.
- Faster algorithms exist for many special cases:
  - Bounded-difference integer-valued matrices;
  - Row-bounded-difference integer-valued matrices;



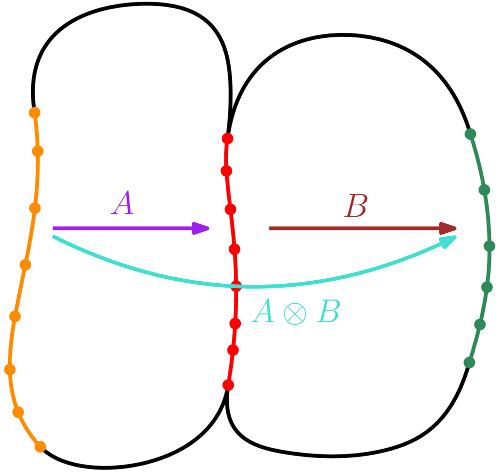
Min-Plus Matrix Product

- Denoted by $C = A \otimes B$.
- Defined as $C_{i,k} = \min_j A_{i,j} + B_{j,k}$.
- Finding the min-plus product of two $n \times n$ matrices is $\Omega(n^{3-o(1)})$-hard assuming the APSP hypothesis.
- Faster algorithms exist for many special cases:
  - Bounded-difference integer-valued matrices;
  - Row-bounded-difference integer-valued matrices;
  - Row-monotone integer-valued matrices with values in $\mathcal{O}(n)$;

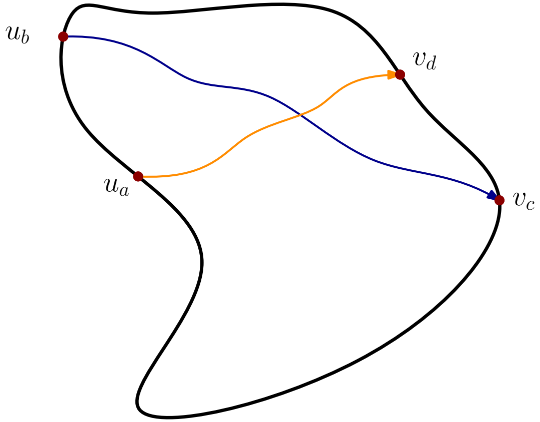


Min-Plus Matrix Product

- Denoted by $C = A \otimes B$.
- Defined as $C_{i,k} = \min_j A_{i,j} + B_{j,k}$.
- Finding the min-plus product of two $n \times n$ matrices is $\Omega(n^{3-o(1)})$-hard assuming the APSP hypothesis.
- Faster algorithms exist for many special cases:
  - Bounded-difference integer-valued matrices;
  - Row-bounded-difference integer-valued matrices;
  - Row-monotone integer-valued matrices with values in $\mathcal{O}(n)$;
  - Monge matrices.

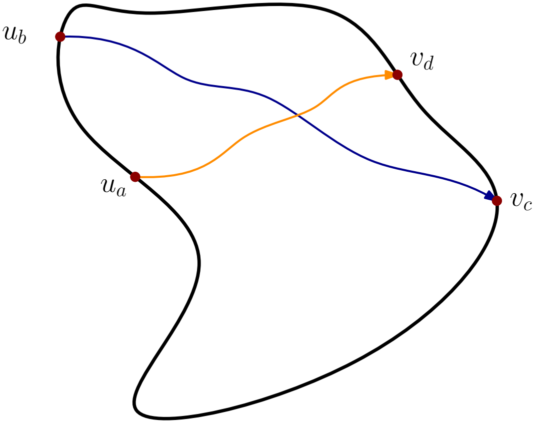


Monge Matrix



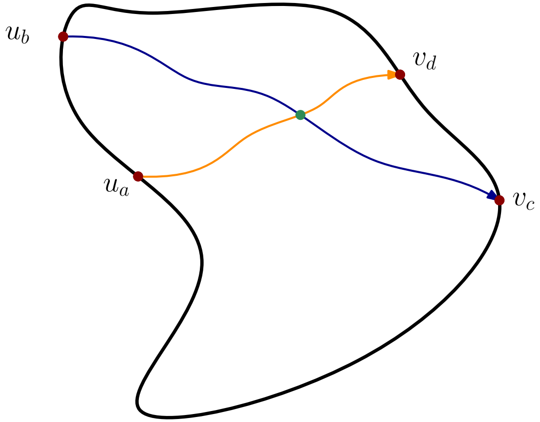
Monge Matrix

- Assume that the graph is planar.



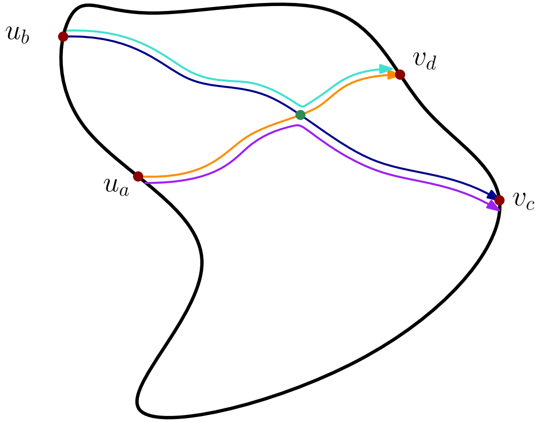
Monge Matrix

- Assume that the graph is planar.



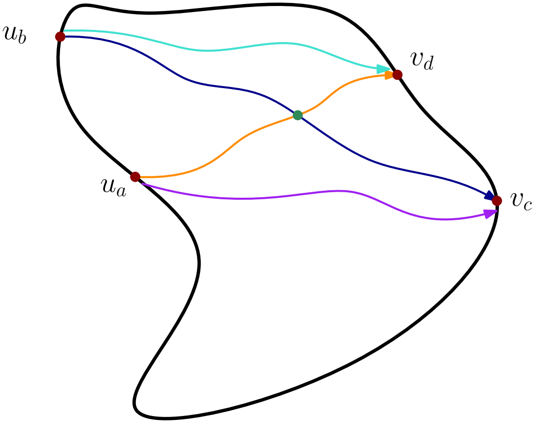
Monge Matrix

- Assume that the graph is planar.



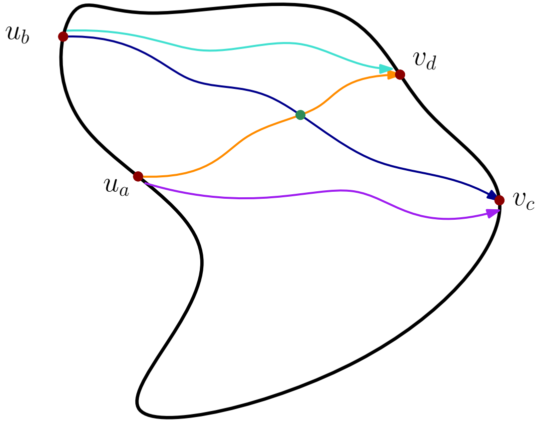
Monge Matrix

- Assume that the graph is planar.



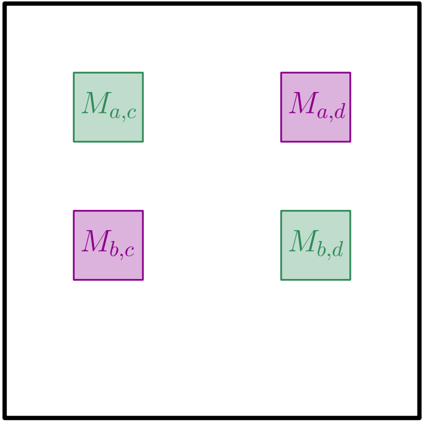
Monge Matrix

- Assume that the graph is planar.
- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.



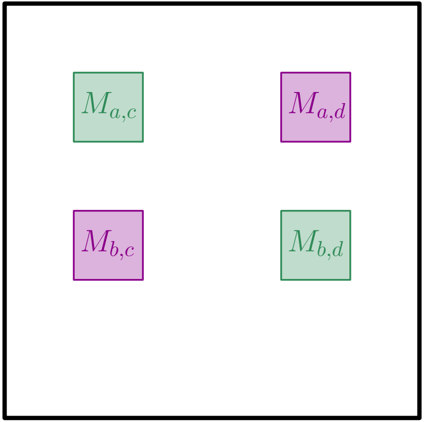
Monge Matrix

- Assume that the graph is planar.
- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.



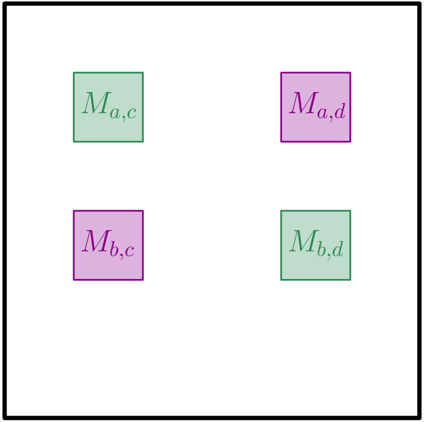
Monge Matrix

- Assume that the graph is planar.
- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- M is a Monge matrix if it satisfies the Monge property for all $a < b$ and $c < d$.



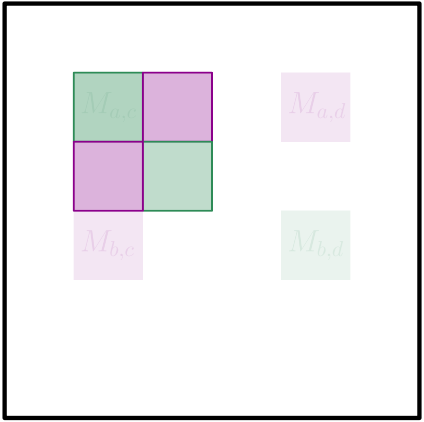
Monge Matrix

- Assume that the graph is planar.
- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- M is a Monge matrix if it satisfies the Monge property for all $a < b$ and $c < d$.
- Sufficient to check only for $b = a + 1$ and $d = c + 1$.



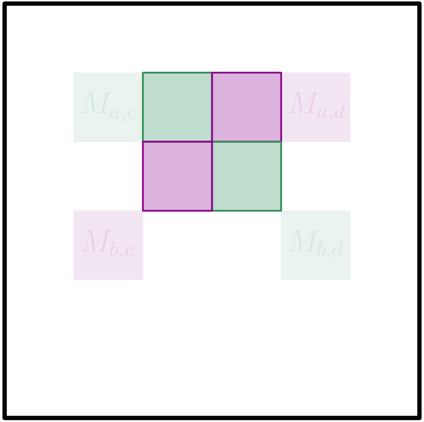
Monge Matrix

- Assume that the graph is planar.
- Monge property:
  $M_{a,c} + M_{b,d} \le M_{a,d} + M_{b,c}$.
- M is a Monge matrix if it satisfies the Monge property for all $a < b$ and $c < d$.
- Sufficient to check only for $b = a + 1$ and $d = c + 1$.



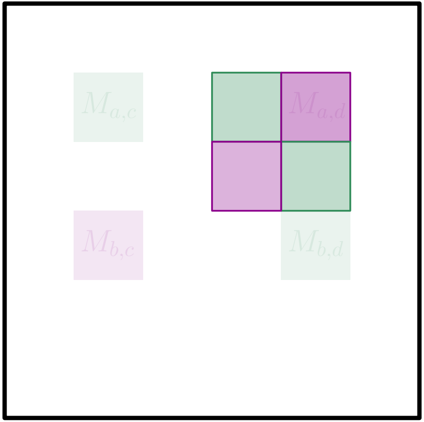
Monge Matrix

- Assume that the graph is planar.
- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- M is a Monge matrix if it satisfies the Monge property for all $a < b$ and $c < d$.
- Sufficient to check only for $b = a + 1$ and $d = c + 1$.



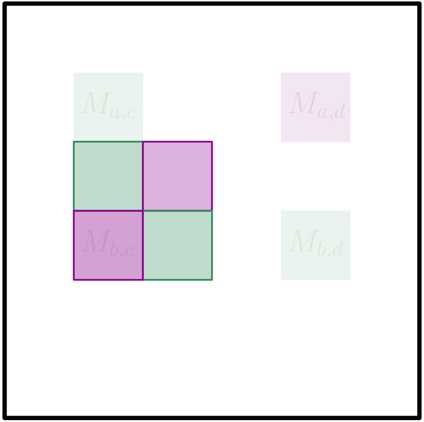
Monge Matrix

- Assume that the graph is planar.
- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- M is a Monge matrix if it satisfies the Monge property for all $a < b$ and $c < d$.
- Sufficient to check only for $b = a + 1$ and $d = c + 1$.



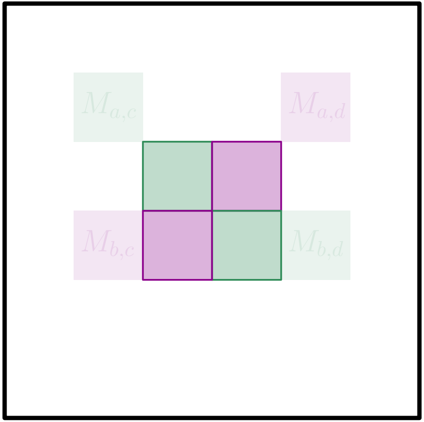
Monge Matrix

- Assume that the graph is planar.
- Monge property: $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- M is a Monge matrix if it satisfies the Monge property for all $a < b$ and $c < d$.
- Sufficient to check only for $b = a + 1$ and $d = c + 1$.



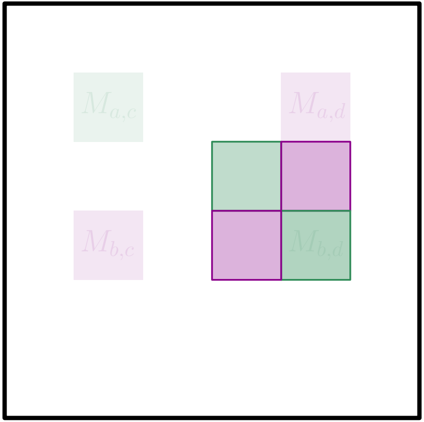
Monge Matrix

- Assume that the graph is planar.
- Monge property: $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- M is a Monge matrix if it satisfies the Monge property for all $a < b$ and $c < d$.
- Sufficient to check only for $b = a + 1$ and $d = c + 1$.



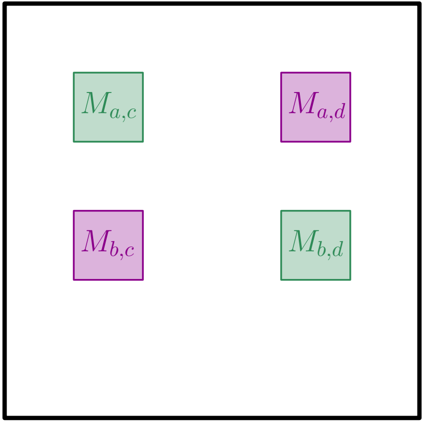
Monge Matrix

- Assume that the graph is planar.
- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- M is a Monge matrix if it satisfies the Monge property for all $a < b$ and $c < d$.
- Sufficient to check only for $b = a + 1$ and $d = c + 1$.



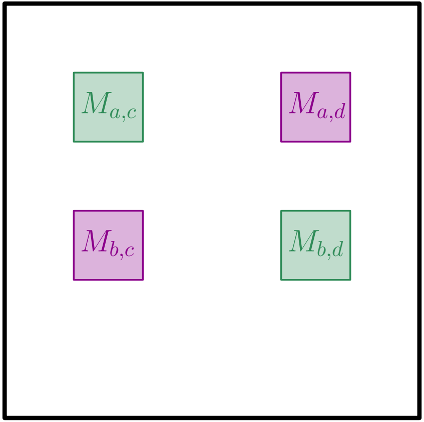
Monge Matrix

- Assume that the graph is planar.
- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- M is a Monge matrix if it satisfies the Monge property for all $a < b$ and $c < d$.
- Sufficient to check only for $b = a + 1$ and $d = c + 1$.



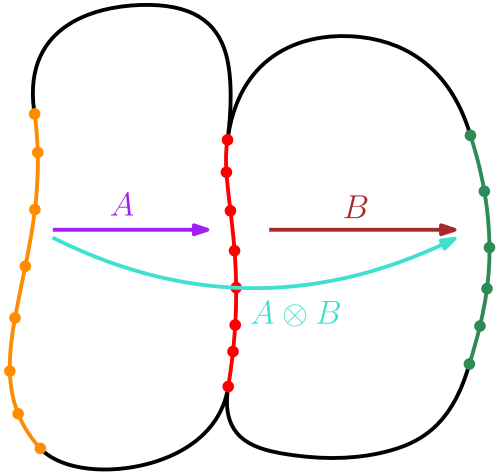
Monge Matrix

- Assume that the graph is planar.
- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- M is a Monge matrix if it satisfies the Monge property for all $a < b$ and $c < d$.
- Sufficient to check only for $b = a + 1$ and $d = c + 1$.
- Allowed for $\widetilde{\mathcal{O}}(n)$-time algorithm for SSSP on planar graphs [FR06].



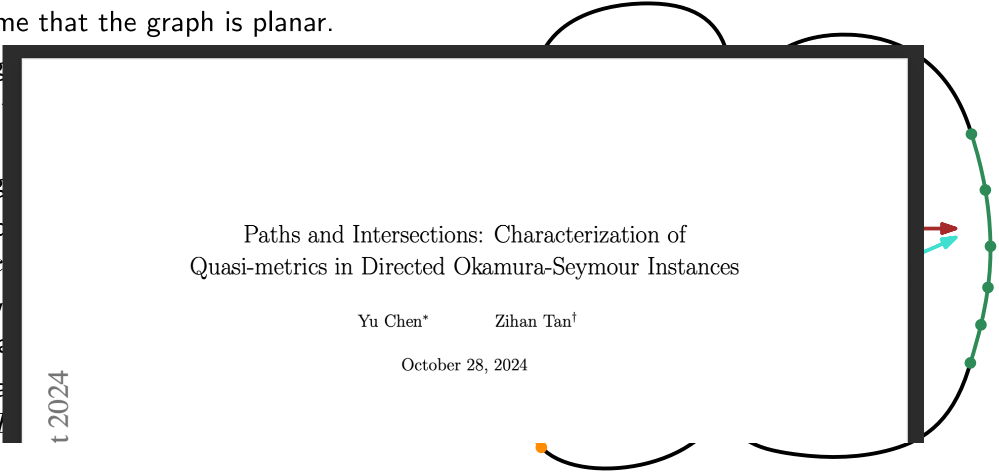
Monge Matrix

- Assume that the graph is planar.
- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- M is a Monge matrix if it satisfies the Monge property for all $a < b$ and $c < d$.
- Sufficient to check only for $b = a + 1$ and $d = c + 1$.
- Allowed for $\widetilde{\mathcal{O}}(n)$-time algorithm for SSSP on planar graphs [FR06].
- If A and B are Monge matrices, then $A \otimes B$ is also a Monge matrix.



Monge Matrix

- Assume that the graph is planar.
- Mong
  $M_{a,c}$
- M is
  Mong
- Suffic
  $d = c$
- Allow
  on pl
- If A a
  $A \otimes$


Paths and Intersections: Characterization of Quasi-metrics in Directed Okamura-Seymour Instances

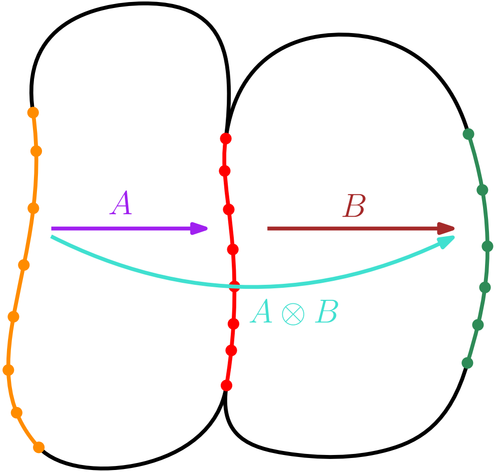
Yu Chen* Zihan Tan†

October 28, 2024

t 2024

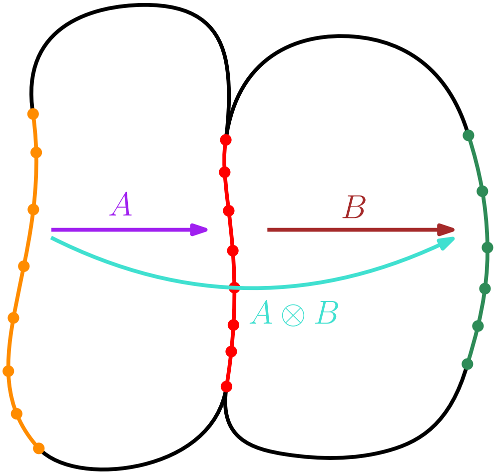
Monge Matrix

- Assume that the graph is planar.
- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- M is a Monge matrix if it satisfies the Monge property for all $a < b$ and $c < d$.
- Sufficient to check only for $b = a + 1$ and $d = c + 1$.
- Allowed for $\widetilde{\mathcal{O}}(n)$-time algorithm for SSSP on planar graphs [FR06].
- If A and B are Monge matrices, then $A \otimes B$ is also a Monge matrix.



Monge Matrix

- Assume that the graph is planar.
- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- M is a Monge matrix if it satisfies the Monge property for all $a < b$ and $c < d$.
- Sufficient to check only for $b = a + 1$ and $d = c + 1$.
- Allowed for $\widetilde{\mathcal{O}}(n)$-time algorithm for SSSP on planar graphs [FR06].
- If A and B are Monge matrices, then $A \otimes B$ is also a Monge matrix.
- SMAWK algorithm finds the min-plus product of two $n \times n$ Monge matrices in $\mathcal{O}(n^2)$ time.



Monge Matrix Core

- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.

Monge Matrix Core

- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- If all inequalities are equalities, we only need to store the first row and the first column.

2	8	4	9	7	7	10
7	13	9	14	12	12	15
6	12	8	13	11	11	14
8	14	10	15	13	13	16
5	11	7	12	10	10	13
10	16	12	17	15	15	18
12	18	14	19	17	17	20

Monge Matrix Core

- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- If all inequalities are equalities, we only need to store the first row and the first column.

2	8	4	9	7	7	10
7	13	9	14	12	12	15
6	12	8	13	11	11	14
8	14	10	15	13	13	16
5	11	7	12	10	10	13
10	16	12	17	15	15	18
12	18	14	19	17	17	20

Monge Matrix Core

- Monge property:
  $M_{a,c} + M_{b,d} \le M_{a,d} + M_{b,c}$.
- If all inequalities are equalities, we only need to store the first row and the first column.

2	8	4	9	7	7	10
7	13	9	14	12	12	15
6	12	8	13	11	11	14
8	14	10	15	13	13	16
5	11	7	12	10	10	13
10	16	12	17	15	15	18
12	18	14	19	17	17	20

Monge Matrix Core

- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- If all inequalities are equalities, we only need to store the first row and the first column.

2	8	4	9	7	7	10
7	13	9	14	12	12	15
6	12	8	13	11	11	14
8	14	10	15	13	13	16
5	11	7	12	10	10	13
10	16	12	17	15	15	18
12	18	14	19	17	17	20

Monge Matrix Core

- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- If all inequalities are equalities, we only need to store the first row and the first column.
- $17 = 10 + 9 - 2$.

2	8	4	9	7	7	10
7	13	9	14	12	12	15
6	12	8	13	11	11	14
8	14	10	15	13	13	16
5	11	7	12	10	10	13
10	16	12	17	15	15	18
12	18	14	19	17	17	20

Monge Matrix Core

- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- If all inequalities are equalities, we only need to store the first row and the first column.
- $17 = 10 + 9 - 2$.

2	8	4	9	7	7	10
1		3				
7	12	8	10	8	8	11
6	11	7	9	7	7	10
8	13	9	11	9	9	12
	1					
5	10	5	7	5	5	8
				2		
10	15	10	12	10	8	11
			4			
12	17	12	14	8	6	9

Monge Matrix Core

- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- If all inequalities are equalities, we only need to store the first row and the first column.
- $17 = 10 + 9 - 2$.
- $12 = 10 + 9 - 2 - 1 - 1 - 3$.

2	8	4	9	7	7	10
7	12	8	10	8	8	11
6	11	7	9	7	7	10
8	13	9	11	9	9	12
5	10	5	7	5	5	8
10	15	10	12	10	8	11
12	17	12	14	8	6	9

Monge Matrix Core

- Monge property: $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- If all inequalities are equalities, we only need to store the first row and the first column.
- $17 = 10 + 9 - 2$.
- $12 = 10 + 9 - 2 - 1 - 1 - 3$.

2		8		4		9		7		7		10
	1				3							
7		12		8		10		8		8		11
6		11		7		9		7		7		10
8		13		9		11		9		9		12
			1									
5		10		5		7		5		5		8
									2			
10		15		10		12		10		8		11
							4					
12		17		12		14		8		6		9

Monge Matrix Core

- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- If all inequalities are equalities, we only need to store the first row and the first column.
- $17 = 10 + 9 - 2$.
- $12 = 10 + 9 - 2 - 1 - 1 - 3$.

2		8		4		9		7		7		10
	1				3							
7		12		8		10		8		8		11
6		11		7		9		7		7		10
8		13		9		11		9		9		12
			1									
5		10		5		7		5		5		8
									2			
10		15		10		12		10		8		11
							4					
12		17		12		14		8		6		9

Monge Matrix Core

- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- If all inequalities are equalities, we only need to store the first row and the first column.
- $17 = 10 + 9 - 2$.
- $12 = 10 + 9 - 2 - 1 - 1 - 3$.
- $9 = 12 + 10 - 2 - 1 - 1 - 3 - 4 - 2$.

2	8	4	9	7	7	10
7	12	8	10	8	8	11
6	11	7	9	7	7	10
8	13	9	11	9	9	12
5	10	5	7	5	5	8
10	15	10	12	10	8	11
12	17	12	14	8	6	9

Monge Matrix Core

- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- If all inequalities are equalities, we only need to store the first row and the first column.
- $17 = 10 + 9 - 2$.
- $12 = 10 + 9 - 2 - 1 - 1 - 3$.
- $9 = 12 + 10 - 2 - 1 - 1 - 3 - 4 - 2$.
- First row, first column, and core elements determine the whole matrix.

2		8		4		9		7		7		10
	1				3							
7		12		8		10		8		8		11
6		11		7		9		7		7		10
8		13		9		11		9		9		12
			1									
5		10		5		7		5		5		8
									2			
10		15		10		12		10		8		11
							4					
12		17		12		14		8		6		9

Monge Matrix Core

- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- If all inequalities are equalities, we only need to store the first row and the first column.
- $17 = 10 + 9 - 2$.
- $12 = 10 + 9 - 2 - 1 - 1 - 3$.
- $9 = 12 + 10 - 2 - 1 - 1 - 3 - 4 - 2$.
- First row, first column, and core elements determine the whole matrix.
- Using an orthogonal range sum data structure, we can efficiently query any element of the matrix.

2		8		4		9		7		7		10
	1				3							
7		12		8		10		8		8		11
6		11		7		9		7		7		10
8		13		9		11		9		9		12
			1									
5		10		5		7		5		5		8
									2			
10		15		10		12		10		8		11
							4					
12		17		12		14		8		6		9

Monge Matrix Core Size

- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- First row, first column, and core elements determine the whole matrix.

2	8	4	9	7	7	10
7	12	8	10	8	8	11
6	11	7	9	7	7	10
8	13	9	11	9	9	12
5	10	5	7	5	5	8
10	15	10	12	10	8	11
12	17	12	14	8	6	9

Monge Matrix Core Size

- Monge property: $M_{a,c} + M_{b,d} \le M_{a,d} + M_{b,c}$.
- First row, first column, and core elements determine the whole matrix.
- Let $\delta(M)$ be the number of elements in the core of M. We use $\widetilde{\mathcal{O}}(n + \delta(M))$ space to encode M.

2		8		4		9		7		7		10
	1				3							
7		12		8		10		8		8		11
6		11		7		9		7		7		10
8		13		9		11		9		9		12
			1									
5		10		5		7		5		5		8
									2			
10		15		10		12		10		8		11
							4					
12		17		12		14		8		6		9

Monge Matrix Core Size

- Monge property:
  $M_{a,c} + M_{b,d} \leq M_{a,d} + M_{b,c}$.
- First row, first column, and core elements determine the whole matrix.
- Let $\delta(M)$ be the number of elements in the core of M. We use $\widetilde{\mathcal{O}}(n + \delta(M))$ space to encode M.
- $M_{0,n-1} + M_{n-1,0} - M_{0,0} - M_{n-1,n-1}$ is the sum of all core values of M.

2		8		4		9		7		7		10
	1				3							
7		12		8		10		8		8		11
6		11		7		9		7		7		10
8		13		9		11		9		9		12
			1									
5		10		5		7		5		5		8
									2			
10		15		10		12		10		8		11
							4					
12		17		12		14		8		6		9

Monge Matrix Core Size

- Monge property: $M_{a,c} + M_{b,d} \le M_{a,d} + M_{b,c}$.
- First row, first column, and core elements determine the whole matrix.
- Let $\delta(M)$ be the number of elements in the core of M. We use $\widetilde{\mathcal{O}}(n + \delta(M))$ space to encode M.
- $M_{0,n-1} + M_{n-1,0} - M_{0,0} - M_{n-1,n-1}$ is the sum of all core values of M.
- If M is integer-valued, then $\delta(M) \le M_{0,n-1} + M_{n-1,0} - M_{0,0} - M_{n-1,n-1}$.

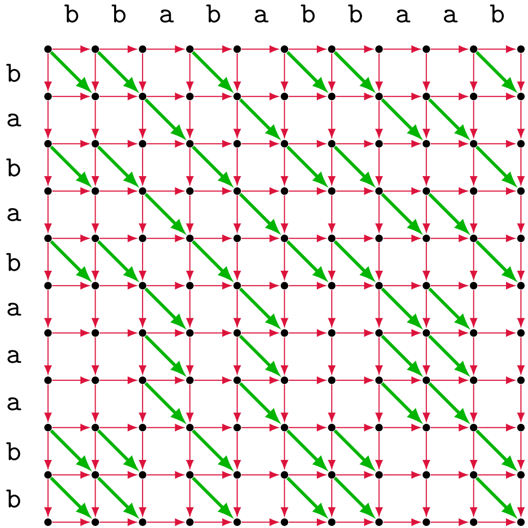
2		8		4		9		7		7		10
	1				3							
7		12		8		10		8		8		11
6		11		7		9		7		7		10
8		13		9		11		9		9		12
			1									
5		10		5		7		5		5		8
									2			
10		15		10		12		10		8		11
							4					
12		17		12		14		8		6		9

Monge Matrix Core Size

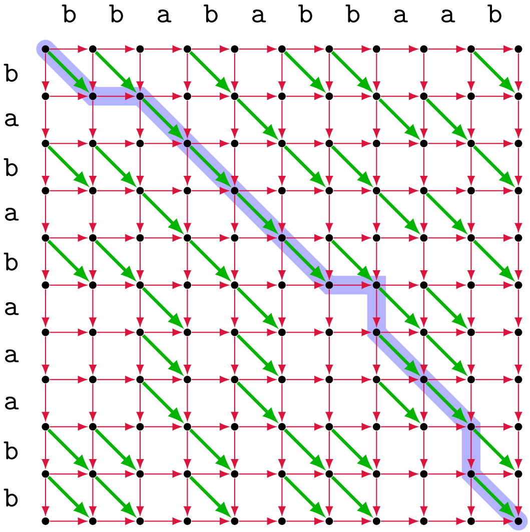
- Monge property: $M_{a,c} + M_{b,d} \le M_{a,d} + M_{b,c}$.
- First row, first column, and core elements determine the whole matrix.
- Let $\delta(M)$ be the number of elements in the core of M. We use $\widetilde{\mathcal{O}}(n + \delta(M))$ space to encode M.
- $M_{0,n-1} + M_{n-1,0} - M_{0,0} - M_{n-1,n-1}$ is the sum of all core values of M.
- If M is integer-valued, then $\delta(M) \le M_{0,n-1} + M_{n-1,0} - M_{0,0} - M_{n-1,n-1}$.
- If M is integer-valued and its elements are in $\mathcal{O}(n)$, we have $\delta(M) = \mathcal{O}(n)$.

2		8		4		9		7		7		10
	1				3							
7		12		8		10		8		8		11
6		11		7		9		7		7		10
8		13		9		11		9		9		12
			1									
5		10		5		7		5		5		8
									2			
10		15		10		12		10		8		11
							4					
12		17		12		14		8		6		9

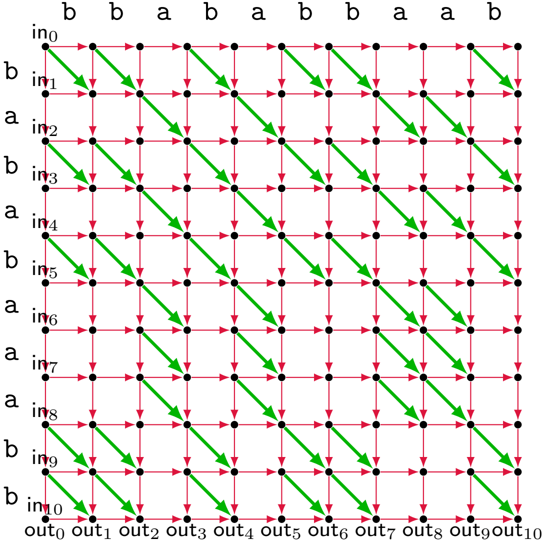
Semi-Local LCS



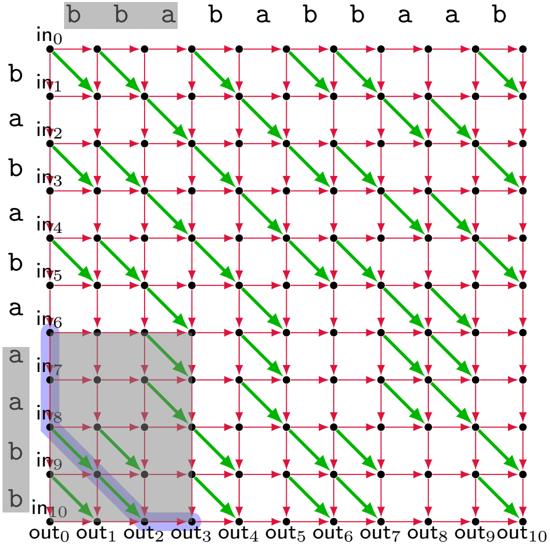
Semi-Local LCS



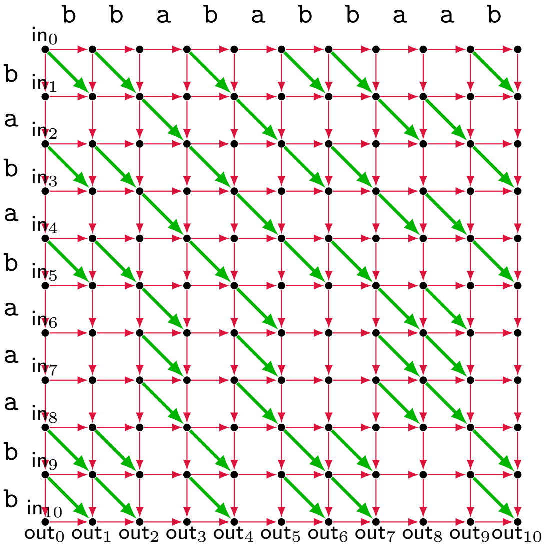
Semi-Local LCS



Semi-Local LCS



Semi-Local LCS



0		1		2		3		4		5		6		7		7		7		8
									1											
0		1		2		3		4		4		5		6		6		6		7
0		1		2		3		4		4		5		6		6		6		7
					1															
0		1		2		2		3		3		4		5		5		5		6
																			1	
0		1		2		2		3		3		4		5		5		5		5
							1													
0		1		2		2		2		2		3		4		4		4		4
0		1		2		2		2		2		3		4		4		4		4
													1							
0		1		2		2		2		2		3		3		3		3		3
											1									
0		1		2		2		2		2		2		2		2		2		2
			1																	
0		1		1		1		1		1		1		1		1		1		1
	1																			
0		0		0		0		0		0		0		0		0		0		0

Min-Plus Multiplication of Simple Subunit-Monge Matrices

Simple Subunit-Monge Matrix

0	1	2	3	4	5	6	7	7	7	8
0	1	2	3	4	4	5	6	6	6	7
0	1	2	3	4	4	5	6	6	6	7
0	1	2	2	3	3	4	5	5	5	6
0	1	2	2	3	3	4	5	5	5	5
0	1	2	2	2	2	3	4	4	4	4
0	1	2	2	2	2	3	4	4	4	4
0	1	2	2	2	2	3	3	3	3	3
0	1	2	2	2	2	2	2	2	2	2
0	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0

Min-Plus Multiplication of Simple Subunit-Monge Matrices

Fact [Tiskin08]

Given the cores of two simple subunit-Monge matrices A and B, their min-plus product $A \otimes B$ is also simple subunit-Monge, and its core can be computed in $\mathcal{O}(n \log n)$ time.

Simple Subunit-Monge Matrix

0	1	2	3	4	5	6	7	7	7	8
0	1	2	3	4	4	5	6	6	6	7
0	1	2	3	4	4	5	6	6	6	7
0	1	2	2	3	3	4	5	5	5	6
0	1	2	2	3	3	4	5	5	5	5
0	1	2	2	2	2	3	4	4	4	4
0	1	2	2	2	2	3	4	4	4	4
0	1	2	2	2	2	3	3	3	3	3
0	1	2	2	2	2	2	2	2	2	2
0	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0

Fact [Tiskin08]

Given the cores of two simple subunit-Monge matrices A and B, their min-plus product $A \otimes B$ is also simple subunit-Monge, and its core can be computed in $\mathcal{O}(n \log n)$ time.

Core-Sparse Monge Matrix

2	8	4	9	7	7	10	11	12	17	23
7	12	8	10	8	8	11	12	13	18	24
6	11	7	9	7	7	10	11	12	17	22
8	13	9	11	9	9	12	13	14	19	24
5	10	5	7	5	5	8	9	9	14	19
10	15	10	12	10	8	11	12	12	17	22
12	17	12	14	8	6	9	10	10	15	20
10	15	10	12	6	2	5	6	5	10	15
15	20	15	16	10	6	9	10	9	14	19
13	18	13	14	8	4	7	8	7	9	14
18	20	15	16	10	6	9	10	9	11	16

Core entries (red, printed between adjacent rows and columns):

1		3							
									1
	1						1		
				2					
			4						
				2			1		
		1							
								3	
3									

Min-Plus Multiplication of Core-Sparse Monge Matrices

Fact [Tiskin08]

Given the cores of two simple subunit-Monge matrices A and B, their min-plus product $A \otimes B$ is also simple subunit-Monge, and its core can be computed in $\mathcal{O}(n \log n)$ time.

Fact [Russo10]

Given the condensed representations of two Monge matrices A and B, the condensed representation of their min-plus product $A \otimes B$ can be computed in $\mathcal{O}((n + \delta(A) + \delta(B) + \delta(A \otimes B)) \log^3 n)$ time.

Core-Sparse Monge Matrix

2		8		4		9		7		7		10		11		12		17		23
	1				3															
7		12		8		10		8		8		11		12		13		18		24
																			1	
6		11		7		9		7		7		10		11		12		17		22
8		13		9		11		9		9		12		13		14		19		24
			1												1					
5		10		5		7		5		5		8		9		9		14		19
									2											
10		15		10		12		10		8		11		12		12		17		22
							4													
12		17		12		14		8		6		9		10		10		15		20
									2						1					
10		15		10		12		6		2		5		6		5		10		15
					1															
15		20		15		16		10		6		9		10		9		14		19
																	3			
13		18		13		14		8		4		7		8		7		9		14
	3																			
18		20		15		16		10		6		9		10		9		11		16

Our Results

- Prior Work:
  - [Tiskin08] $\mathcal{O}(n \log n)$-time algorithm for min-plus multiplication of simple subunit-Monge matrices.
  - [Russo10] $\mathcal{O}((n + \delta(A) + \delta(B) + \delta(A \otimes B)) \log^3 n)$-time algorithm for min-plus multiplication of arbitrary Monge matrices.

Our Results

- Prior Work:
  - [Tiskin08] $\mathcal{O}(n \log n)$-time algorithm for min-plus multiplication of simple subunit-Monge matrices.
  - [Russo10] $\mathcal{O}((n + \delta(A) + \delta(B) + \delta(A \otimes B)) \log^3 n)$-time algorithm for min-plus multiplication of arbitrary Monge matrices.
- Our Results:

Our Results

- Prior Work:
  - [Tiskin08] $\mathcal{O}(n \log n)$-time algorithm for min-plus multiplication of simple subunit-Monge matrices.
  - [Russo10] $\mathcal{O}((n + \delta(A) + \delta(B) + \delta(A \otimes B)) \log^3 n)$-time algorithm for min-plus multiplication of arbitrary Monge matrices.
- Our Results:

Theorem 1 [this work]

If A and B are Monge matrices then $\delta(A \otimes B) \leq 2 \cdot (\delta(A) + \delta(B))$.

Our Results

- Prior Work:
  - [Tiskin08] $\mathcal{O}(n \log n)$-time algorithm for min-plus multiplication of simple subunit-Monge matrices.
  - [Russo10] $\mathcal{O}((n + \delta(A) + \delta(B) + \delta(A \otimes B)) \log^3 n)$-time algorithm for min-plus multiplication of arbitrary Monge matrices.
- Our Results:

Theorem 1 [this work]

If A and B are Monge matrices then $\delta(A \otimes B) \leq 2 \cdot (\delta(A) + \delta(B))$.

Theorem 2 [this work]

Given the condensed representations of two Monge matrices A and B, the condensed representation of their min-plus product $A \otimes B$ can be computed in $\mathcal{O}(n + (\delta(A) + \delta(B)) \log n)$ time.

Applications

Applications

- [GK25]: Integer-weight edit distance.

Applications

- [GK25]: Integer-weight edit distance.
  ↪ Using our algorithm saves two log factors compared to the application of [Russo10].

Applications

- [GK25]: Integer-weight edit distance.
  $\hookrightarrow$ Using our algorithm saves two log factors compared to the application of [Russo10].
- Range LIS queries:

Applications

- [GK25]: Integer-weight edit distance.
  - $\hookrightarrow$ Using our algorithm saves two log factors compared to the application of [Russo10].
- Range LIS queries:

Reference	*Preprocessing*	*Length Queries*
[Tiskin08]	$\mathcal{O}(n \log^2 n)$	$\mathcal{O}(\log n)$

Applications

- [GK25]: Integer-weight edit distance.
  - $\hookrightarrow$ Using our algorithm saves two log factors compared to the application of [Russo10].
- Range LIS queries:

Reference	*Preprocessing*	*Length Queries*	*Reporting Queries*
[Tiskin08]	$\mathcal{O}(n \log^2 n)$	$\mathcal{O}(\log n)$	——

Applications

- [GK25]: Integer-weight edit distance.
  - ↪ Using our algorithm saves two log factors compared to the application of [Russo10].
- Range LIS queries:

Reference	*Preprocessing*	*Length Queries*	*Reporting Queries*
[Tiskin08]	$\mathcal{O}(n \log^2 n)$	$\mathcal{O}(\log n)$	——
[KS24]	$\mathcal{O}(n^{3/2} \log^3 n)$		$\mathcal{O}(n^{1/2} \log^3 n + \lvert ANS \rvert)$

Applications

- [GK25]: Integer-weight edit distance.
  $\hookrightarrow$ Using our algorithm saves two log factors compared to the application of [Russo10].
- Range LIS queries:

Reference	*Preprocessing*	*Length Queries*	*Reporting Queries*
[Tiskin08]	$\mathcal{O}(n \log^2 n)$	$\mathcal{O}(\log n)$	——
[KS24]	$\mathcal{O}(n^{3/2} \log^3 n)$		$\mathcal{O}(n^{1/2} \log^3 n + \|ANS\|)$
[this work]	$\mathcal{O}(n \log^2 n)$		$\mathcal{O}(\|ANS\| \log n)$
[this work]	$\mathcal{O}(n \log^3 n)$		$\mathcal{O}(\|ANS\|)$

Summary and Open Problems

- Results:

Summary and Open Problems

- Results:
  - If A and B are Monge matrices then $\delta(A \otimes B) \leq 2 \cdot (\delta(A) + \delta(B))$.

Summary and Open Problems

- Results:
  - If A and B are Monge matrices then $\delta(A \otimes B) \leq 2 \cdot (\delta(A) + \delta(B))$.
  - Min-plus product of Monge matrices A and B can be computed in $\mathcal{O}(n + (\delta(A) + \delta(B)) \log n)$ time.

Summary and Open Problems

- Results:
  - If A and B are Monge matrices then $\delta(A \otimes B) \leq 2 \cdot (\delta(A) + \delta(B))$.
  - Min-plus product of Monge matrices A and B can be computed in $\mathcal{O}(n + (\delta(A) + \delta(B)) \log n)$ time.
  - Range LIS reporting queries can be answered in $\mathcal{O}(|ANS|)$ time after $\mathcal{O}(n \log^3 n)$-time preprocessing.

Summary and Open Problems

- Results:
  - If A and B are Monge matrices then $\delta(A \otimes B) \leq 2 \cdot (\delta(A) + \delta(B))$.
  - Min-plus product of Monge matrices A and B can be computed in $\mathcal{O}(n + (\delta(A) + \delta(B)) \log n)$ time.
  - Range LIS reporting queries can be answered in $\mathcal{O}(|ANS|)$ time after $\mathcal{O}(n \log^3 n)$-time preprocessing.
- Open Problems:

Summary and Open Problems

- Results:
  - If A and B are Monge matrices then $\delta(A \otimes B) \le 2 \cdot (\delta(A) + \delta(B))$.
  - Min-plus product of Monge matrices A and B can be computed in $\mathcal{O}(n + (\delta(A) + \delta(B)) \log n)$ time.
  - Range LIS reporting queries can be answered in $\mathcal{O}(|ANS|)$ time after $\mathcal{O}(n \log^3 n)$-time preprocessing.
- Open Problems:
  - If we can prove $\delta(A \otimes B) \le c \cdot (\delta(A) + \delta(B)) + \widetilde{\mathcal{O}}(n)$ for some $c \in [1, 2)$, we automatically get a solution for *weighted* range LIS with $\widetilde{\mathcal{O}}(n^{1+\log_2 c})$ preprocessing time, $\widetilde{\mathcal{O}}(1)$ query time, and $\widetilde{\mathcal{O}}(|ANS|)$ reporting time.

Summary and Open Problems

- Results:
  - If A and B are Monge matrices then $\delta(A \otimes B) \leq 2 \cdot (\delta(A) + \delta(B))$.
  - Min-plus product of Monge matrices A and B can be computed in $\mathcal{O}(n + (\delta(A) + \delta(B)) \log n)$ time.
  - Range LIS reporting queries can be answered in $\mathcal{O}(|ANS|)$ time after $\mathcal{O}(n \log^3 n)$-time preprocessing.
- Open Problems:
  - If we can prove $\delta(A \otimes B) \leq c \cdot (\delta(A) + \delta(B)) + \widetilde{\mathcal{O}}(n)$ for some $c \in [1, 2)$, we automatically get a solution for *weighted* range LIS with $\widetilde{\mathcal{O}}(n^{1+\log_2 c})$ preprocessing time, $\widetilde{\mathcal{O}}(1)$ query time, and $\widetilde{\mathcal{O}}(|ANS|)$ reporting time.
  - Improve range LIS reporting preprocessing time to $\mathcal{O}(n \log^2 n)$.

Summary and Open Problems

- Results:
  - If A and B are Monge matrices then $\delta(A \otimes B) \leq 2 \cdot (\delta(A) + \delta(B))$.
  - Min-plus product of Monge matrices A and B can be computed in $\mathcal{O}(n + (\delta(A) + \delta(B)) \log n)$ time.
  - Range LIS reporting queries can be answered in $\mathcal{O}(|ANS|)$ time after $\mathcal{O}(n \log^3 n)$-time preprocessing.
- Open Problems:
  - If we can prove $\delta(A \otimes B) \leq c \cdot (\delta(A) + \delta(B)) + \widetilde{\mathcal{O}}(n)$ for some $c \in [1, 2)$, we automatically get a solution for *weighted* range LIS with $\widetilde{\mathcal{O}}(n^{1+\log_2 c})$ preprocessing time, $\widetilde{\mathcal{O}}(1)$ query time, and $\widetilde{\mathcal{O}}(|ANS|)$ reporting time.
  - Improve range LIS reporting preprocessing time to $\mathcal{O}(n \log^2 n)$.
  - Find more applications of core-sparse Monge matrix multiplication.

Summary and Open Problems

- Results:
  - If A and B are Monge matrices then $\delta(A \otimes B) \leq 2 \cdot (\delta(A) + \delta(B))$.
  - Min-plus product of Monge matrices A and B can be computed in $\mathcal{O}(n + (\delta(A) + \delta(B)) \log n)$ time.
  - Range LIS reporting queries can be answered in $\mathcal{O}(|ANS|)$ time after $\mathcal{O}(n \log^3 n)$-time preprocessing.
- Open Problems:
  - If we can prove $\delta(A \otimes B) \leq c \cdot (\delta(A) + \delta(B)) + \widetilde{\mathcal{O}}(n)$ for some $c \in [1, 2)$, we automatically get a solution for *weighted* range LIS with $\widetilde{\mathcal{O}}(n^{1+\log_2 c})$ preprocessing time, $\widetilde{\mathcal{O}}(1)$ query time, and $\widetilde{\mathcal{O}}(|ANS|)$ reporting time.
  - Improve range LIS reporting preprocessing time to $\mathcal{O}(n \log^2 n)$.
  - Find more applications of core-sparse Monge matrix multiplication.

Thank you!