Graph Coloring Problems in the Two-Party Communication Model

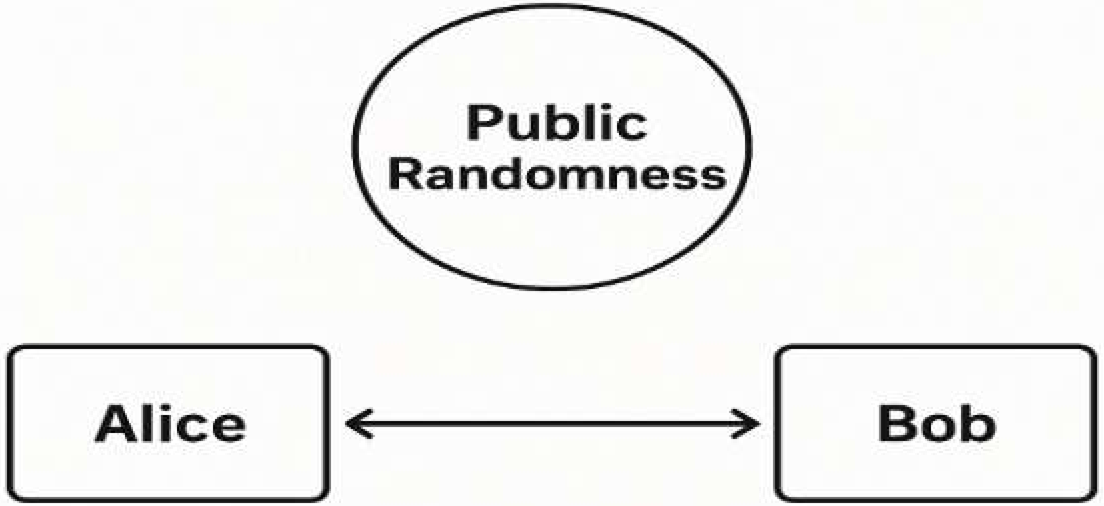
Yi-Jun Chang[2] Gopinath Mishra[1] Nguyen Thuan Hung[2] Farrel DSalim[2]

[1]Institute of Mathematical Sciences (IMSc), Chennai, India

[2]National University of Singapore, Singapore

Workshop on Foundations of Distributed and Parallel Graph Algorithms
Venice, Italy
20 May 2026

Two-party communication model



parameters:

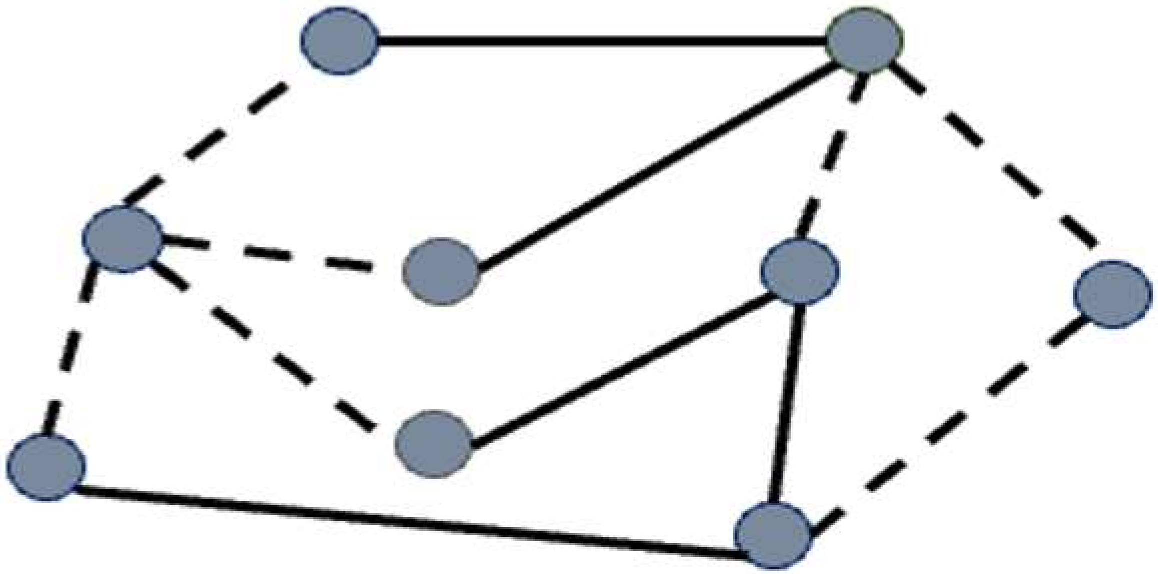
1. Communication complexity: # bits communicated
2. Round complexity: Total number of rounds

Private randomness

- Any public-coin protocol can be simulated using private coins.
- So, public and private randomness are equivalent up to a small additive overhead.
- For this talk, only think of public randomness.

The model for the graph problems

- The vertex set V is known to both Alice and Bob.
- Edges are partitioned among them.



Complexity of some graph problems

Graph Problem	Upper Bound	Lower Bound	Rounds	References
Connectivity	$O(n \log n)$	$\Omega(n)$	1	[HMT'88]
Minimum Cut	$O(n \log n)$ (randomized)	$\Omega(n \log\log n)$	2	[AD'21]
Maximum Matching	$O(n \log^2 n)$ (deterministic)	$\Omega(n)$	$O(n)$	[BVMEN'22, BFS'86; Raz'92]

$O(n)$ vs. almost $O(n)$ communication cost!

Problems and key take away

- $(\Delta + 1)$-Vertex Coloring
- $(2\Delta - 1)$-Edge Coloring
- Maximal Independent Set
- Maximal Matching

n:#vertices
Δ:the maximum degree

Problems and key take away

- $(\Delta + 1)$-Vertex Coloring
- $(2\Delta - 1)$-Edge Coloring
- Maximal Independent Set
- Maximal Matching

n:#vertices
Δ:the maximum degree

The communication complexities of all the problems is $\Theta(n)$ bits with some caveats on round complexities.

We give communication optimal round efficient algorithms!

Problems and key take away

Problem	Communication Complexity	Round complexity	Reference
$(\Delta+1)$-Vertex Coloring	$\Theta(n)$	$O(\log\log n \cdot \log\Delta)$	[FM’25] & [CMNS’25a]
$(2\Delta-1)$-Edge Coloring	$\Theta(n)$	$O(1)$	[CMNS’25a]
Maximal Independent Set	$\Theta(n)$	$O((\log n)^{1/2})$	[CMNS’25b]
Maximal Matching	$\Theta(n)$	1	[CMNS’25b]

Problems and key take away

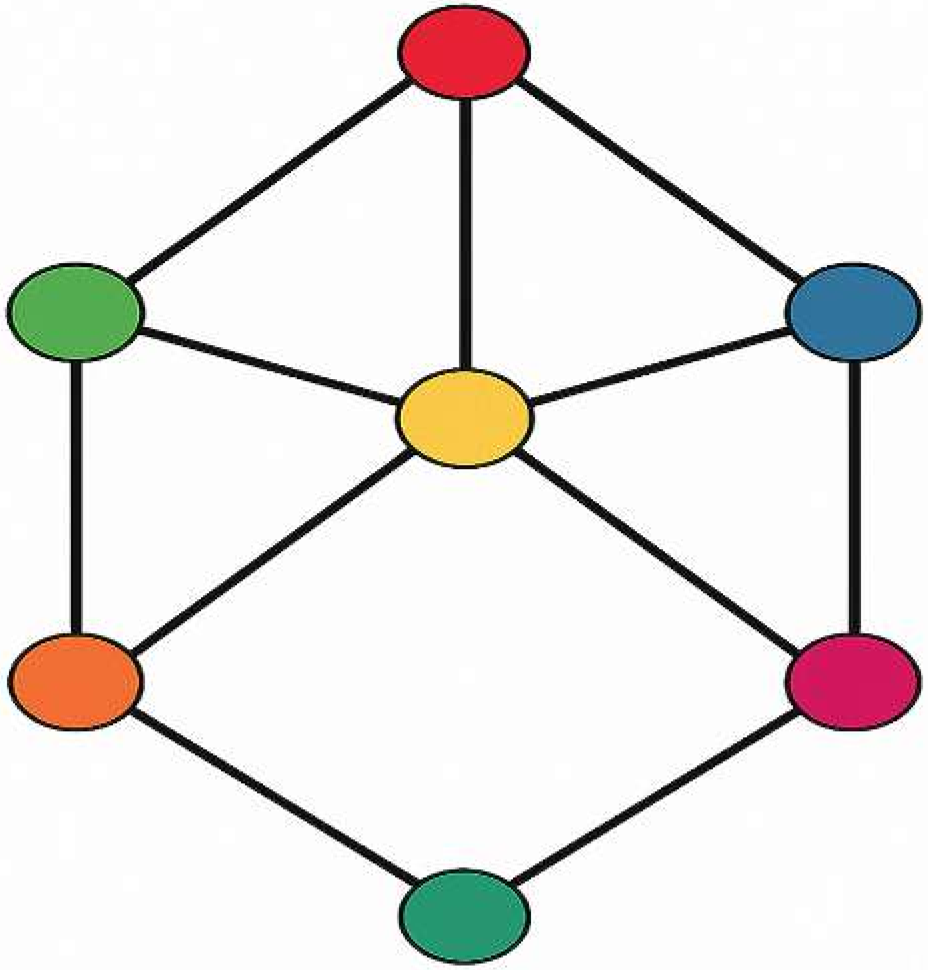
Problem	Communication Complexity	Round complexity	Reference
$(\Delta+1)$-Vertex Coloring	$\Theta(n)$	$O(\log\log n \cdot \log \Delta)$	[FM'25] & [CMNS'25a]
$(2\Delta-1)$-Edge Coloring	$\Theta(n)$	$O(1)$	[CMNS'25a]
Maximal Independent Set	$\Theta(n)$	$O((\log n)^{1/2})$	[CMNS'25b]
Maximal Matching	$\Theta(n)$	1	[CMNS'25b]

All the algorithms are communication optimal irrespective of the number of rounds.

$(\Delta + 1)$-Vertex Coloring

$(\Delta + 1)$-Vertex Coloring

- Each node is to use colors from $\{1, \ldots, \Delta + 1\}$.
- No two adjacent vertices can use the same color.
- Objective of each party is to report the colors of all vertices.



Trivial algorithm: $O(\min\{n\Delta \log n, n^2\})$ bits and 1 round.

Our algorithm for $(\Delta + 1)$-Vertex Coloring

- $O(n)$ bits of communication and $O(\log\log n \cdot \log\Delta)$ rounds.

- This improves the previous results by [FM'25] with the same communication complexity of but of $\Omega(n)$ rounds.

Simulating the greedy algorithm

- Pick vertices one by one in some arbitrary order.
- For each vertex, find an available color.
- For a vertex v and color c, whether v can be colored with color c can be verified with 2 bits of communication.

Communication cost= $O(n\Delta)$ bits

Number of rounds = $O(n\Delta)$

Can we find an available color efficiently?

Using binary search!

- The total number of already used colors is at most Δ.
- Divide the color set $\{1, \dots, \Delta + 1\}$ into two equal sets L and R.
- Alice and Bob communicate the number of used colors in L and R.
- For at least one of L and R, the total number of used colors must be less than $\Delta/2$
- Then recurse.

Communication cost= $O(\log^2 \Delta)$.

Number of rounds = $O(\log \Delta)$.

Implication for $(\Delta + 1)$-Vertex Coloring

A deterministic algorithm

Communication cost= $O(\mathrm{n} \cdot \log^2 \Delta)$.

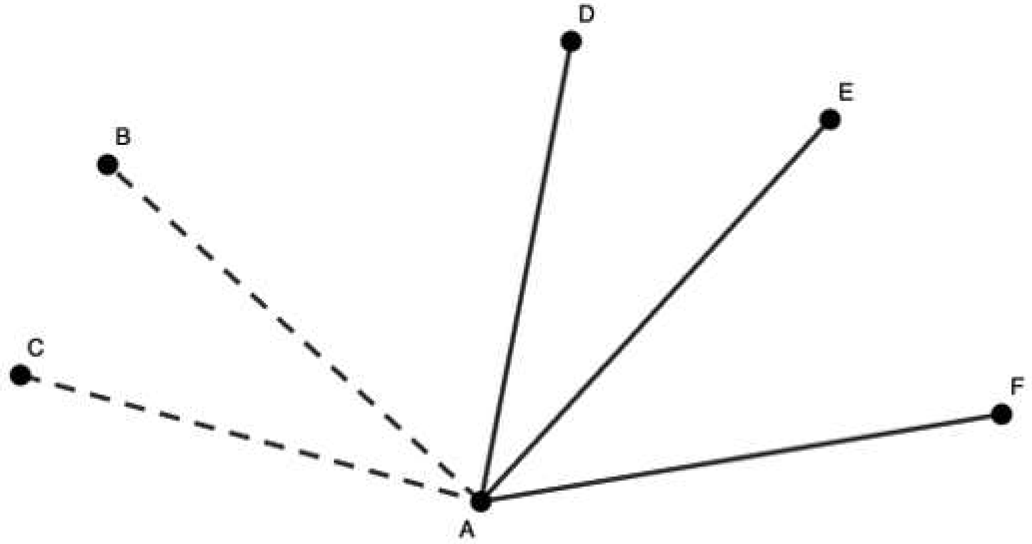
Number of rounds = $O(\mathrm{n} \cdot \log \Delta)$.

Can we find an available color efficiently?

Can we find an available color efficiently?

[FM'25] k_v is the number of available colors, one can be found/sampled with (in expectation):

- $O(\log^2(\Delta/k_v))$ bits
- $O(\log(\Delta/k_v))$ rounds



The final algorithm by [FM'25]

- Alice and Bob use public randomness to order the vertices uniformly at random.

- Each vertex v finds an available color using $O(\log^2 \Delta / k_v)$ bits of communication and in $O(\log \Delta / k_v)$ rounds: both in expectation!

- k_v is uniformly distributed over $\{\Delta + 1, \dots, \Delta + 1 - d(v)\}$.

- The expected communication cost to color v is

$$\frac{1}{d(v)+1} \sum_{k_v = \Delta+1-d(v)}^{\Delta+1} \log^2(\Delta / k_v) = O(1).$$

The final algorithm by [FM'25]

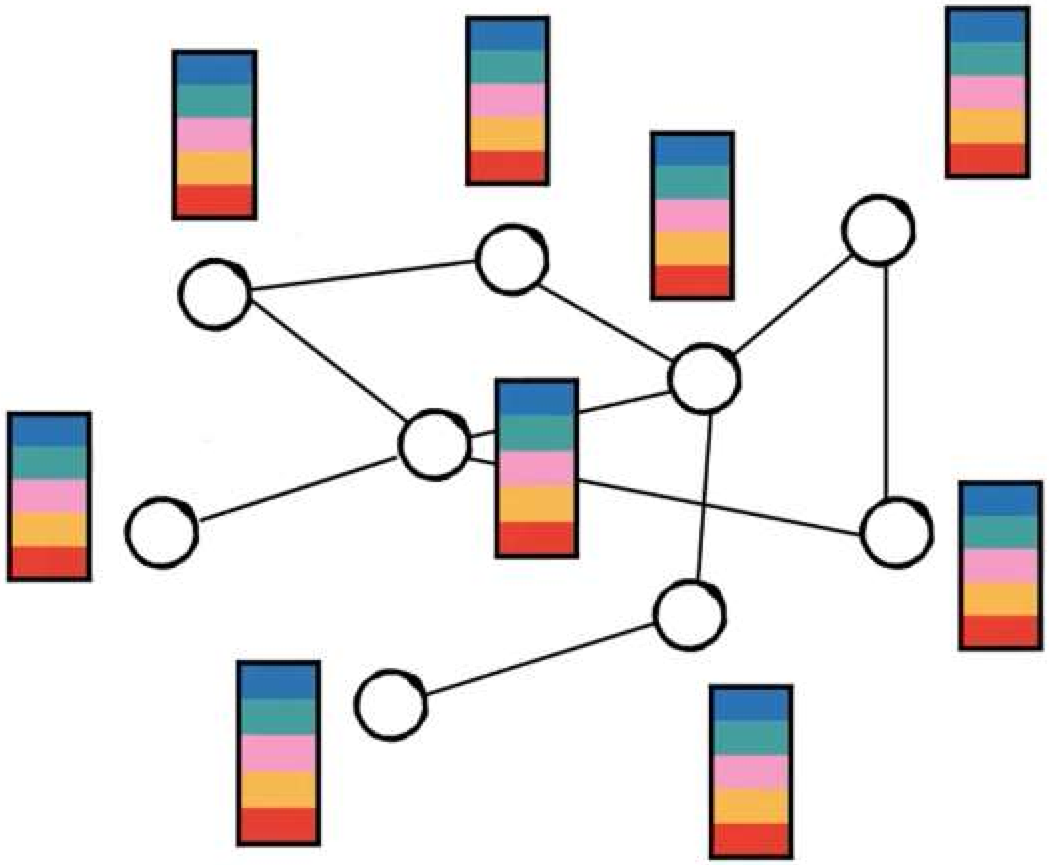
- Expected communication complexity is $O(n)$.
- Expected number of rounds is $O(n)$.
- Communication optimal, but round inefficient.

We improve the worst case round complexity to $O(\log\log n \cdot \log \Delta)$!

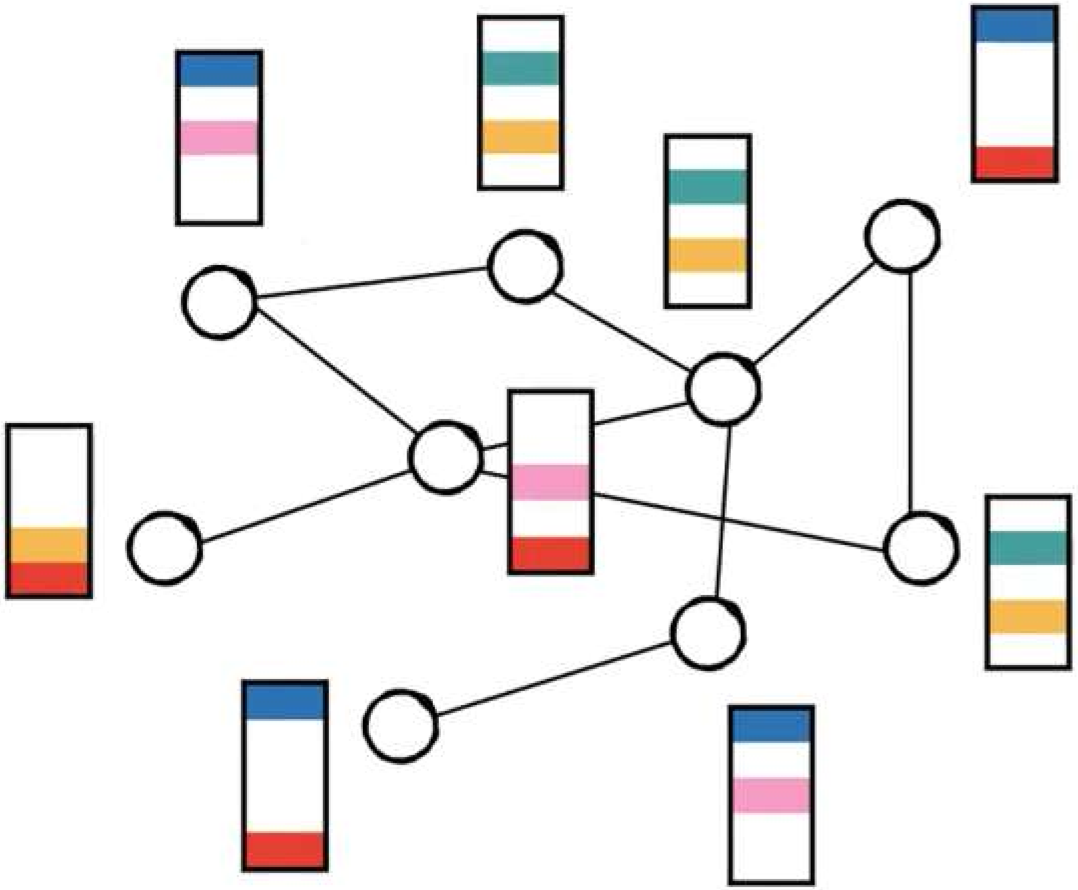
Ingredient One: Palette sparsification

- Introduced by [ACK'19] for streaming graph coloring.
- Allows for sparsifying the graph dramatically while still allowing for a valid $\Delta + 1$ coloring.

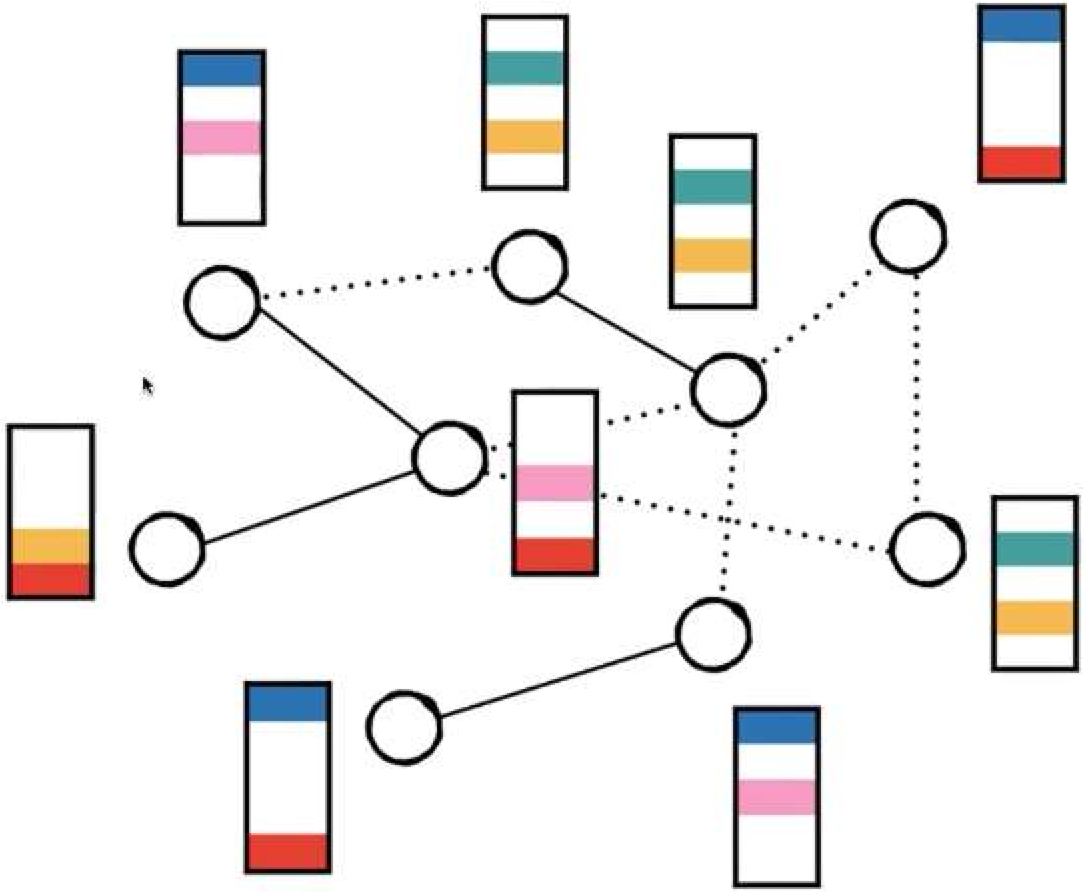
Ingredient One: Palette sparsification



Ingredient One: Palette sparsification



Ingredient One: Palette sparsification



Ingredient One: Palette sparsification

Palette sparsification theorem [HKNT'22][ACK'19]

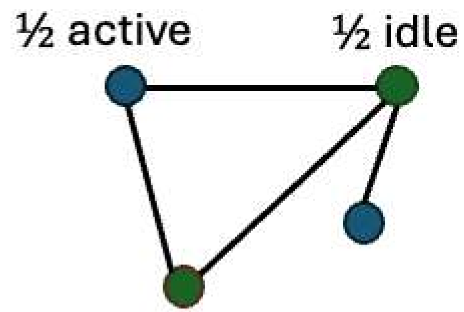
- Each vertex independently and uniformly sampling $O(\log\ n)$ colors from its palette.
- Remove edge (u, v) if the sampling lists are disjoint.
- Remaining $O(n \cdot \log^2 n)$ edges

- To sample colors, Alice and Bob can get them for free from public randomness.
- Implies an one-round algorithm with $O(n \log^3 n)$ bits of communication

Ingredient Two: Random Color Trial

Random Color Trial

- Every (uncolored) vertex participates with probability 1/2
  - Sample a color uniformly at random from available palette using [FM'25].
- Vertex v is colored with the sampled color if none of the neighbors have selected it



[FM'25] k_A is the number of available colors, one can be sampled with

- $O(\log^2(\Delta/k_A))$ bits in expectation:
- $O(\log(\Delta))$ rounds in the worst case

Pr(v is colored)$\geq \frac{1}{24}$.

In expectation, constant fraction of vertices Will be colored in an iteration.

Our algorithm for $(\Delta + 1)$-Vertex Coloring

- Run random Color Trial for $O(\log \log n)$ iterations.

- Use palette sparsification theorem on remaining nodes.

Our algorithm for $(\Delta + 1)$-Vertex Coloring

- Run random Color Trial for $O(\log\log n)$ iterations.

Expected # remaining nodes = $O(n/\log^3 n)$

- Use palette sparsification theorem on remaining nodes.

Our algorithm for $(\Delta + 1)$-Vertex Coloring

- Run random Color Trial for $O(\log \log n)$ iterations.

Expected # remaining nodes = $O(n/\log^3 n)$

- Use palette sparsification theorem on remaining nodes.

1 round is enough to color the remaining nodes

Our algorithm for $(\Delta + 1)$-Vertex Coloring

- Run random Color Trial for $O(\log \log n)$ iterations.

Expected # remaining nodes = $O(n / \log^3 n)$

- Use palette sparsification theorem on remaining nodes.

1 round is enough to color the remaining nodes

Total number of rounds = $O(\log \log n \cdot \log \Delta)$ in the worst case.

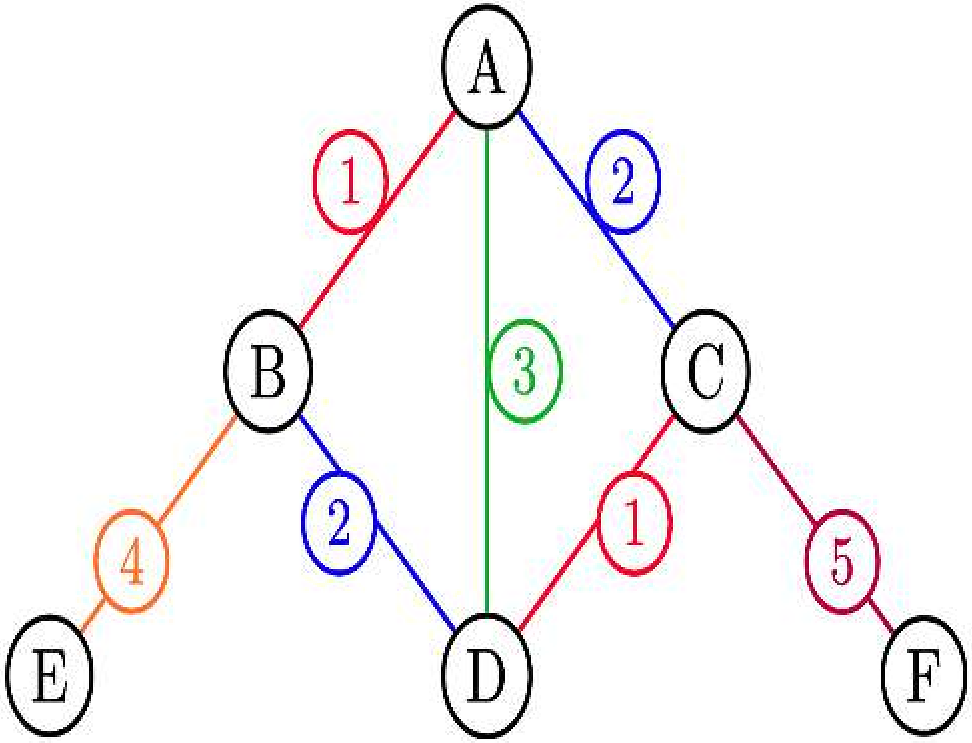
Communication complexity: main challenge?

- In [FM'25], a vertex gets colored in a step based on the random permutation.
- In our case, a vertex can be colored in any iteration and in an iteration multiple vertices may be colored requiring different communication cost and rounds.

$(2\Delta - 1)$-Edge Coloring

$(2\Delta - 1)$-Edge Coloring

- Each edge is to use colors from $\{1, \ldots, 2\Delta - 1\}$.
- No two adjacent edges can use the same color.
- Objective of each party is to report the color of its own edges.



Trivial algorithm: $O(\min\{n\Delta \log n, n^2\})$ bits and 1 round.

Our result on $(2\Delta - 1)$-Edge Coloring

- Deterministic protocol.
- $O(n)$ bits of communication and $O(1)$ rounds.
- We give first edge coloring protocol in the two party communication model.

Our result on $(2\Delta - 1)$-Edge Coloring

- Deterministic protocol.
- $O(n)$ bits of communication and $O(1)$ rounds.
- We give first edge coloring protocol in the two party communication model.
- In this talk, we discuss a randomized $O(\log \log n)$ round protocol.

Simulating the greedy algorithm

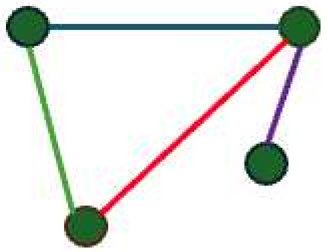
- Alice can color the color of all of its own edges.
- Then, for each vertex, she can send a Δ-bit string denoting the list of available colors.
- Bob resume the algorithm with the information.

Communication cost= $O(n\Delta)$ bits
Number of rounds = 1

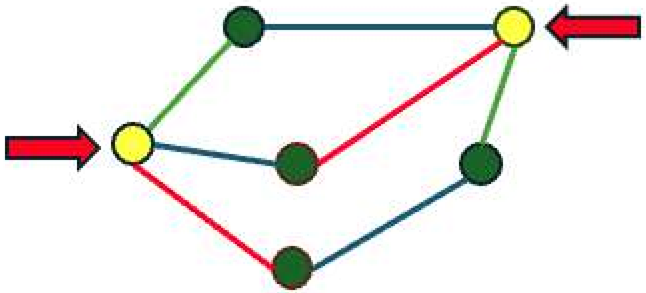
-

Background

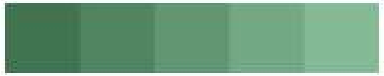
Vizing's Theorem [Viz64]: A simple graph requires either Δ or $(\Delta + 1)$ colors for proper edge coloring.



Fournier's Extension [Fou73]: If the maximum-degree vertices form an independent set, then only Δ colors are needed.



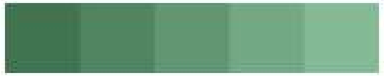
(2Δ)-Edge Coloring Protocol



Alice's palette of size Δ

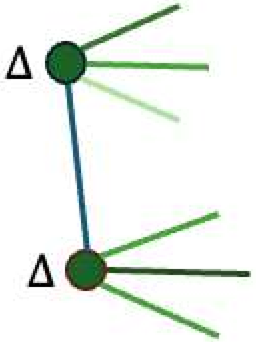
Bob's palette of size Δ

(2Δ)-Edge Coloring Protocol



Alice's palette of size Δ

Bob's palette of size Δ

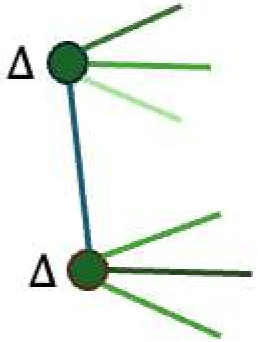


Consider an edge with Alice whose two vertices have Δ neighbors in Alice. Use any color from Bob.

(2Δ)-Edge Coloring Protocol

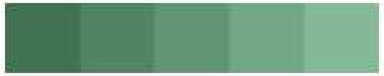
Alice's palette of size Δ

Bob's palette of size Δ



In the remaining subgraph of Alice and Bob, the vertices of degree Δ form independent set.

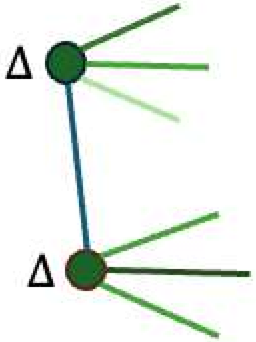
(2Δ)-Edge Coloring Protocol



Alice’s palette of size Δ

Bob’s palette of size Δ

Fournier’s Extension [Fou73]: If the maximum -degree vertices form an independent set, then only Δ colors are needed.



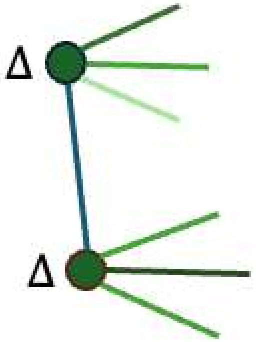
In the remaining subgraph of Alice and Bob, the vertices of degree Δ form independent set. Hence, edge with both end points degree Δ can use any color from the other party.

(2Δ)-Edge Coloring Protocol

Alice's palette of size Δ

Bob's palette of size Δ

Fournier's Extension [Fou73]: If the maximum -degree vertices form an independent set, then only Δ colors are needed.

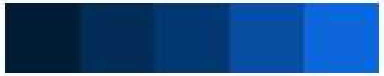


In the remaining subgraph of Alice and Bob, the vertices of degree Δ form independent set. Hence, edge with both end points degree Δ can use any color from the other party.

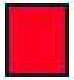
Zero communication required for (2Δ)-edge coloring.

$(2\Delta - 1)$-Edge Coloring Protocol

Alice's palette of size $\Delta - 1$



Bob's palette of size $\Delta - 1$



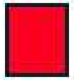
Special color

Fournier's Extension [Fou73]: If the maximum-degree = $\Delta - 1$ vertices form an **independent set**, then only $\Delta - 1$ colors are needed.

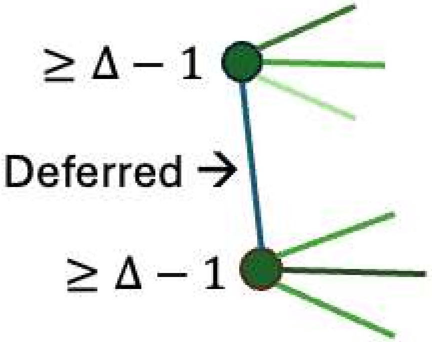
Deferred edges

Alice's palette of size $\Delta - 1$

Bob's palette of size $\Delta - 1$



Special color

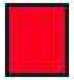


Fournier's Extension [Fou73]: If the maximum-degree = $\Delta - 1$ vertices form an **independent set**, then only $\Delta - 1$ colors are needed.

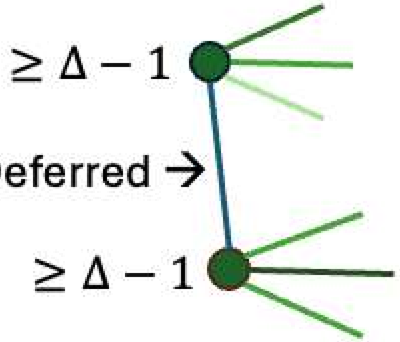
$(2\Delta - 1)$-Edge Coloring Protocol

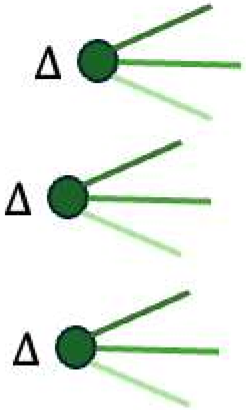
Alice's palette of size $\Delta - 1$

Bob's palette of size $\Delta - 1$



Special color

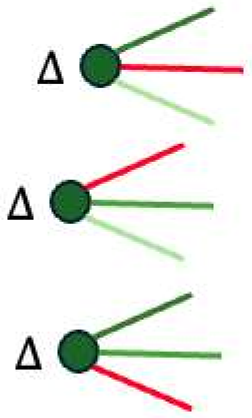




Fournier's Extension [Fou73]: If the maximum-degree = $\Delta - 1$ vertices form an **independent set**, then only $\Delta - 1$ colors are needed.

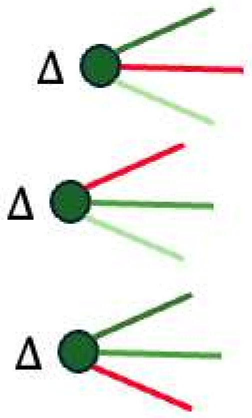
Δ-degree vertices are "independent"

Δ-Matching



Δ-degree vertices are originally "independent"

Δ-Matching



Δ-degree vertices are originally "independent"

A *Matching* covers these vertices. We call it Δ-Matching

$(2\Delta - 1)$-Edge Coloring Protocol

We started with deferring the coloring of the edges with both the vertices having degree at least $\Delta - 1$.

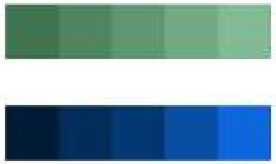
Now we have deferred the coloring of the Δ-matching edges. So, **Fournier's Extension** holds.

So, Alice and Bob can color the remaining graph with the set of own $\Delta - 1$ colors.

Fournier's Extension [Fou73]: If the maximum-degree = $\Delta - 1$ vertices form an **independent set**, then only $\Delta - 1$ colors are needed.

We are left with only $O(n)$ edges. Not Done yet!

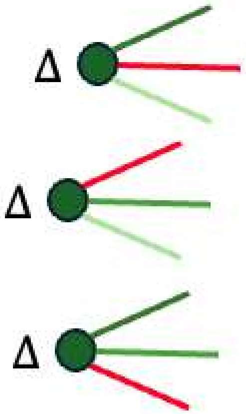
Use of special color



Alice's palette of size $\Delta - 1$

Bob's palette of size $\Delta - 1$

Special color

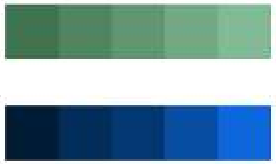


Δ-degree vertices are originally "independent"

A Δ-Matching covers these vertices

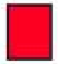
Color the matching by special color

Use of special color

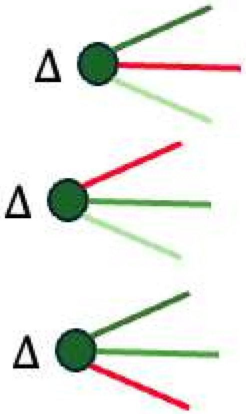


Alice's palette of size $\Delta - 1$

Bob's palette of size $\Delta - 1$



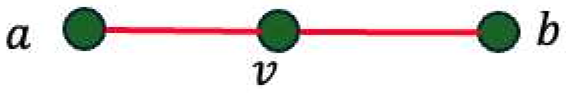
Special color



Δ-degree vertices are originally "independent"

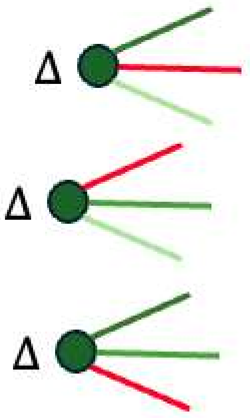
A Δ-Matching covers these vertices

Color the matching by special color



An matching edge of Alice and another of Bob may be adjacent.

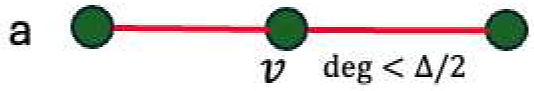
An observation



Δ-degree vertices are originally "independent"

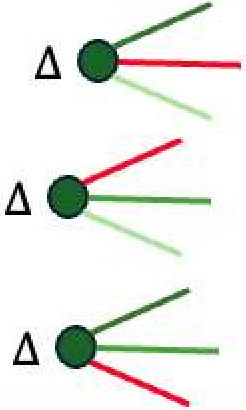
A Δ-Matching covers these vertices

Color the matching by special color



An matching edge of Alice and another of Bob may be adjacent.

Special color and random sampling



Color vb with special color and av with a color from Bob using random sampling.

Δ-degree vertices are originally "independent"

A Δ-Matching covers these vertices.

Color the matching by special color

An matching edge of Alice and another of Bob may be adjacent.

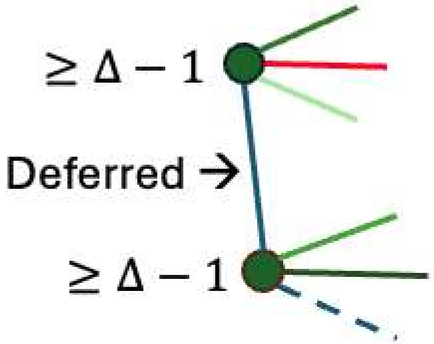
Coloring the Δ-matching edges

- Alice and communicate $O(n)$ bits regarding the match vertices.
- They color the Δ-matching edges that need to be colored using special color.
- For $O(\log\log n)$ rounds they try to color the other Δ-matching edges.
- In each round, the probability that a desired edges gets colored is at east a constant.
- After that, in expectation only $O(n/\log n)$ matching edges left, which can be communicate explicitly using $O(n)$ expected bits.

Coloring the deferred edges

- The subgraph induced by the deferred edges has max degree 2.

- We can run essentially $O(n\,\Delta)$ bit and $O(1)$ round algorithm discussed earlier.



$(2\Delta - 1)$-Edge Coloring Protocol

- Each party defer coloring of some edges incident to high degree vertices to apply

Fournier's Extension [Fou73]: If the maximum-degree = $\Delta - 1$ vertices form an **independent set**, then only $\Delta - 1$ colors are needed.

- Then color the Δ-matching edges using expected O(n) bits of communication and O(log log n) rounds: Only randomization step.

- Color the deferred subgraph using $O(n)$ bits of communication and $O(1)$ rounds.

$(2\Delta - 1)$-Edge Coloring Protocol

- Each party defer coloring of some edges incident to high degree vertices to apply

Fournier's Extension [Fou73]: If the maximum-degree = $\Delta - 1$ vertices form an **independent set**, then only $\Delta - 1$ colors are needed.

- Then color the Δ-matching edges using O(n) bits of communication and O(1) rounds deterministically.

- Color the deferred subgraph using $O(n)$ bits of communication and $O(1)$ rounds.

Maximal Independent Set

Maximal Independent Set

Algorithm	Communication Complexity	Rounds Complexity	Random Greedy?
Prior Works			
Greedy MIS	$O(n)$	$O(n)$	Yes
Luby's Algorithm	$O(n \log n)$	$O(\log n)$	No
Parallel Random Greedy MIS	$O(n \log n)$	$O(\log n)$	Yes
Batched processing	$O(n \cdot \text{poly}(\log n))$	$O(\log \log \Delta)$	Yes
New Results			
Lower Bound	$\Omega(n)$	Any	Any
Main MIS Result	$O(n)$	$O(\sqrt{\log n})$	No
Random Greedy MIS	$O(n)$	$O(\log n)$	Yes
Round-Efficient MIS-1	$O(n \log n)$	$O(\log \log \Delta)$	Yes
Round-Efficient MIS-2	$O(n \log \log n)$	$O(\log \log n)$	No

Summary and open problems

Problem	Communication Complexity	Round complexity	Reference
$(\Delta+1)$-Vertex Coloring	$\Theta(n)$	$O(\log\log n \cdot \log \Delta)$	[FM'25] & [CMNS'25a]
$(2\Delta - 1)$-Edge Coloring	$\Theta(n)$	$O(1)$	[CMNS'25a]
Maximal Independent Set	$\Theta(n)$	$O((\log n)^{1/2})$	[CMNS'25b]
Maximal Matching	$\Theta(n)$	1	[CMNS'25b]

Summary and open problems

Problem	Communication Complexity	Round complexity	Reference
(Δ+1)-Vertex Coloring	$\Theta(n)$	$O(\log\log n \cdot \log \Delta)$	[FM'25] & [CMNS'25a]
$(2\Delta - 1)$-Edge Coloring	$\Theta(n)$	$O(1)$	[CMNS'25a]
Maximal Independent Set	$\Theta(n)$	$O((\log n)^{1/2})$	[CMNS'25b]
Maximal Matching	$\Theta(n)$	1	[CMNS'25b]

Open problems:

1. Improve the round complexities or prove lower bounds.

2. Consider other problems in this model, for example, Δ-veretex-coloring