Paper URL

Improved Algorithms for Overlapping and Robust Clustering of Edge-Colored Hypergraphs: An LP-Based Combinatorial Approach

Changyeol Lee[1*], Yongho Shin[2*], and Hyung-Chan An[1]

[1] Yonsei University, South Korea

[2] University of Wrocław, Poland

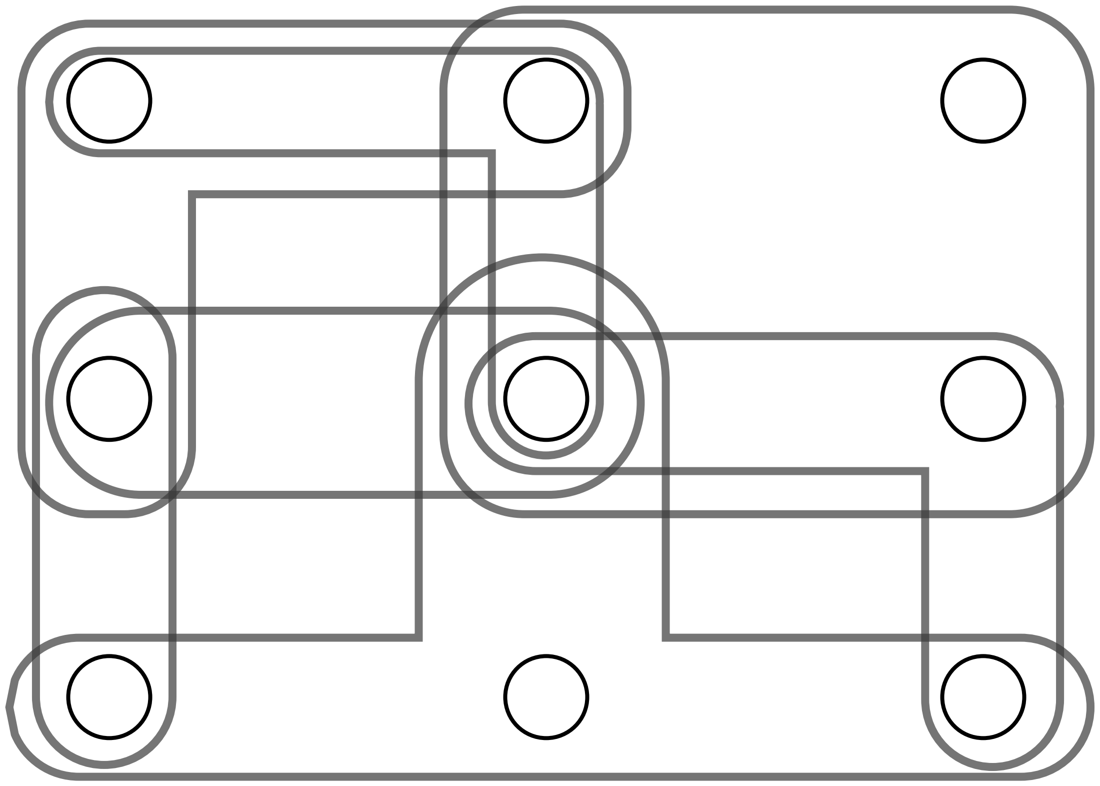
[*] Equal contributions

NEURAL INFORMATION PROCESSING SYSTEMS

NeurIPS 2025: Dec 2-7, 2025

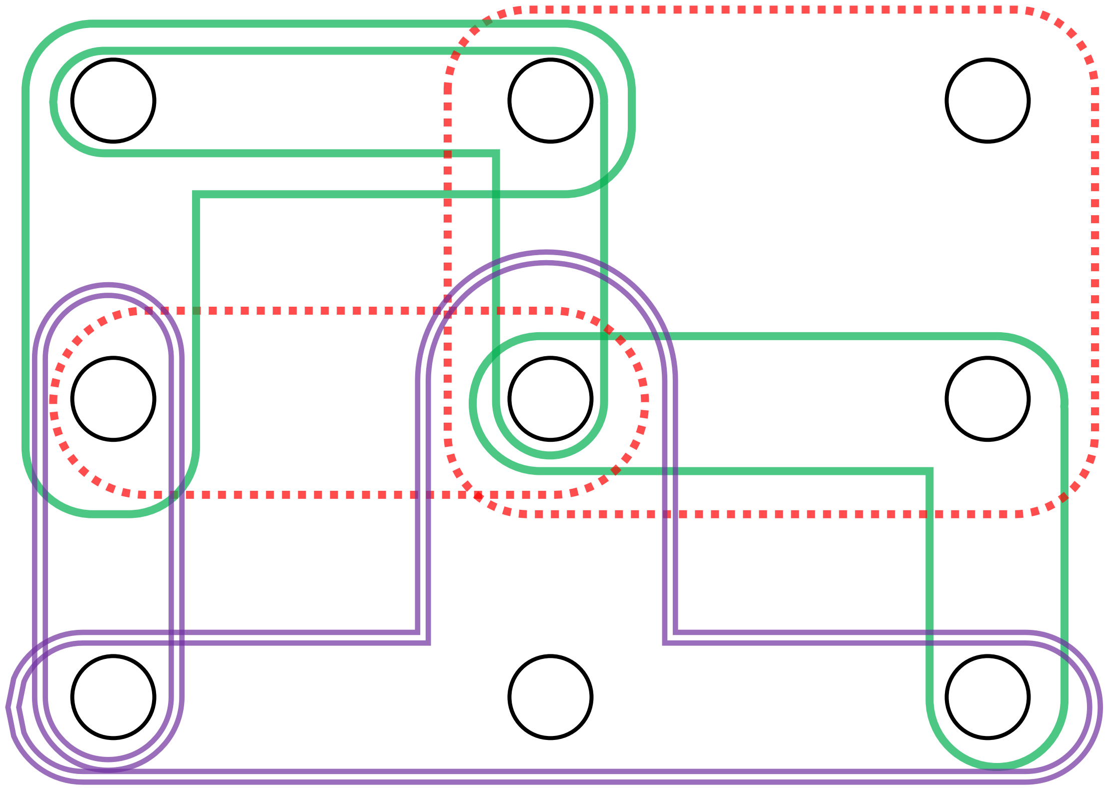
Edge-colored clustering (ECC)

A hypergraph is given



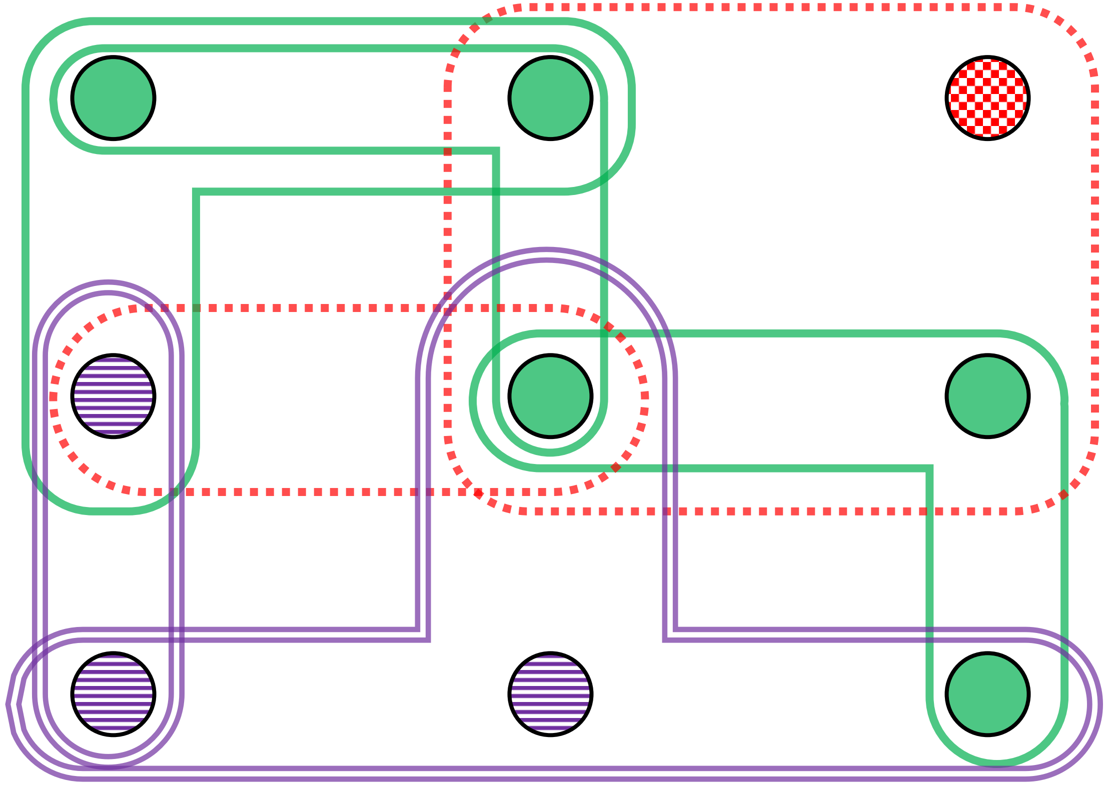
Edge-colored clustering (ECC)

- An edge-colored hypergraph is given



Edge-colored clustering (ECC)

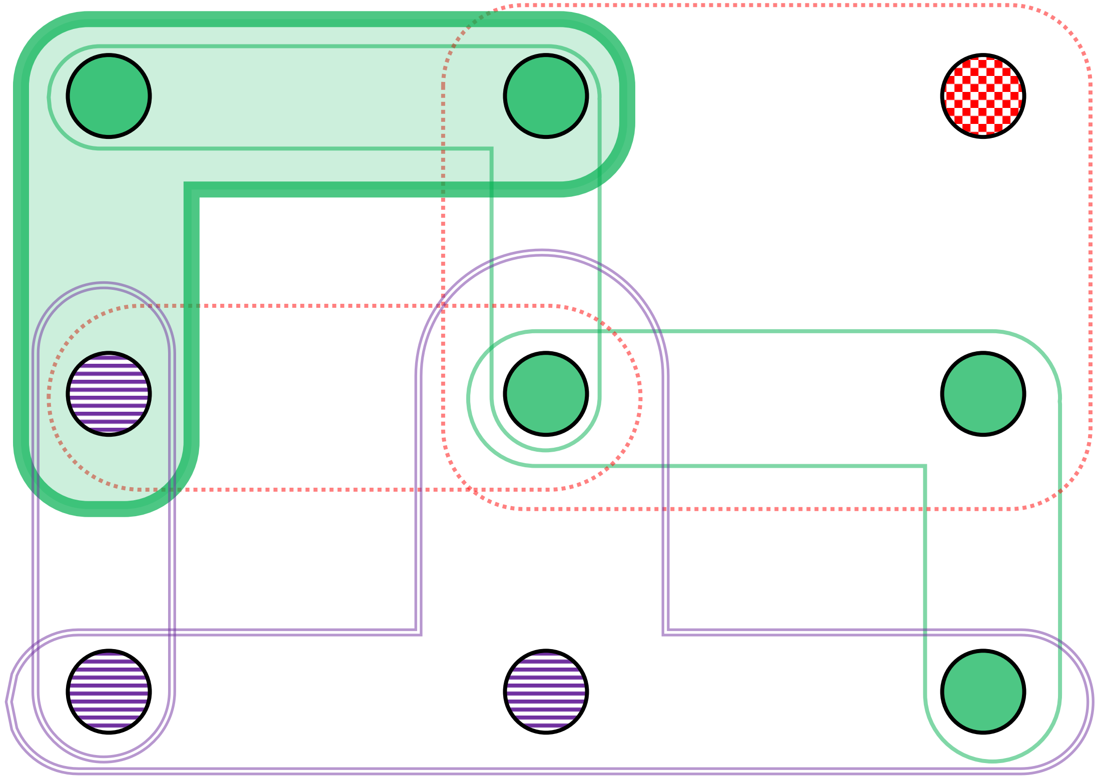
- An edge-colored hypergraph is given; assign one color to each node



Edge-colored clustering (ECC)

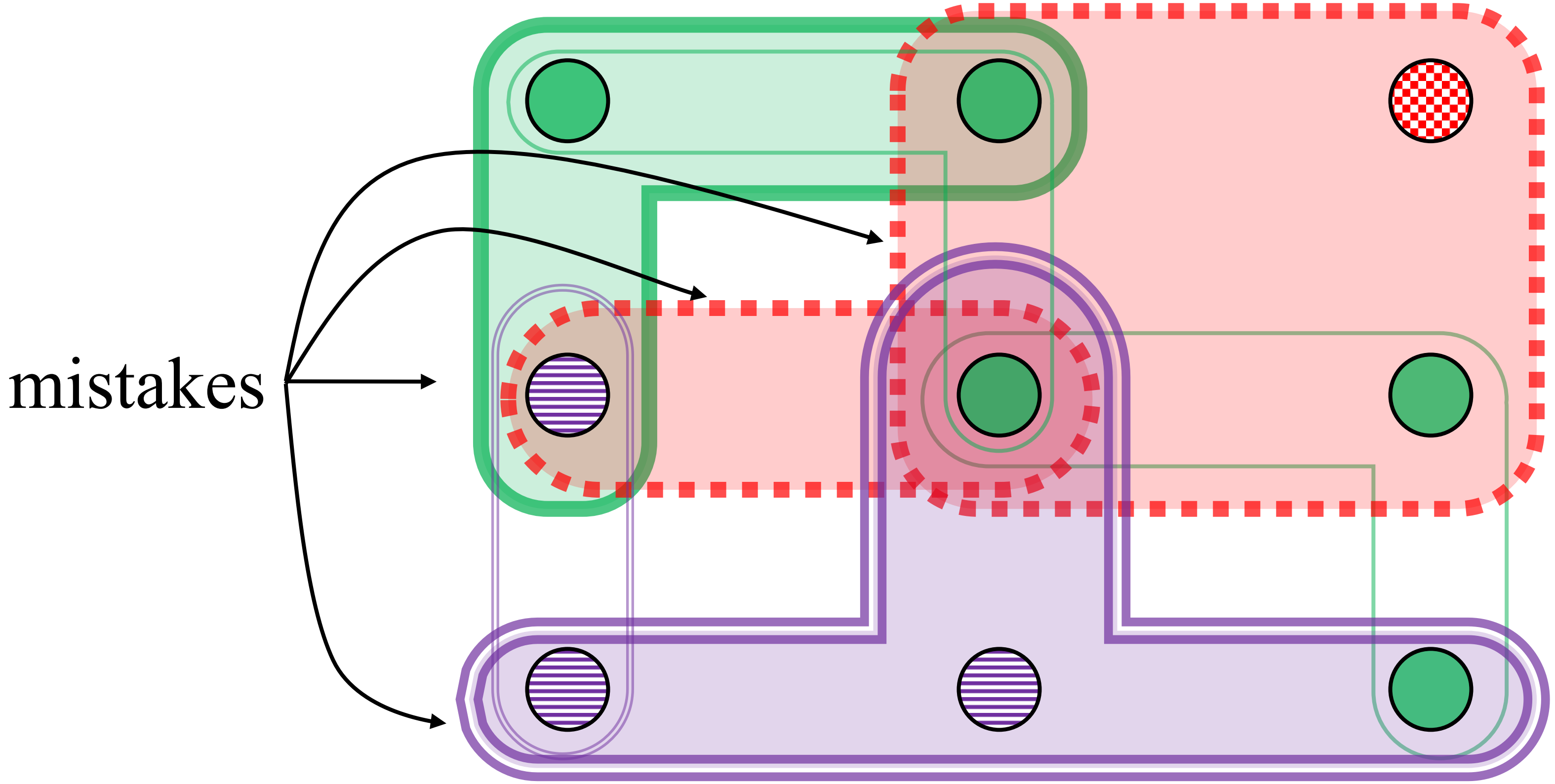
- An edge-colored hypergraph is given; assign one color to each node

mistake →



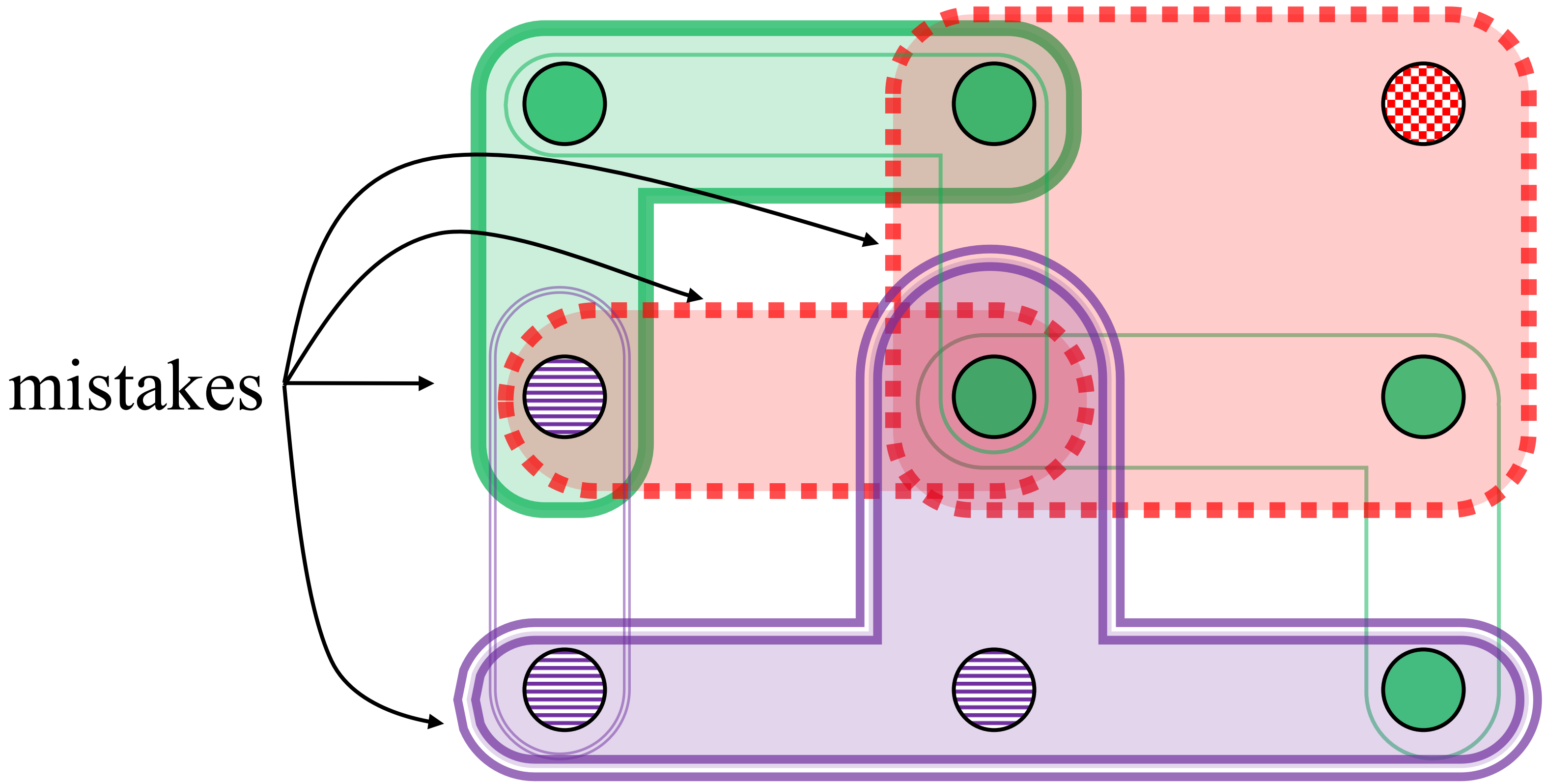
Edge-colored clustering (ECC)

- An edge-colored hypergraph is given; assign one color to each node

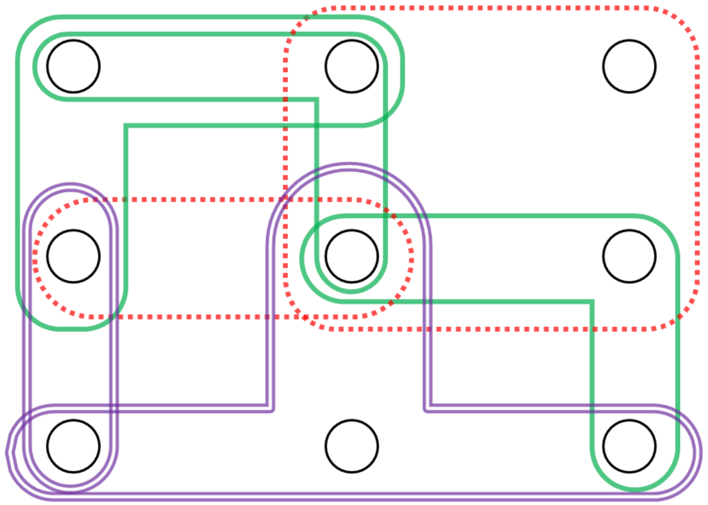


Edge-colored clustering (ECC)

- **Goal**: minimize #mistakes



Edge-colored clustering (ECC)



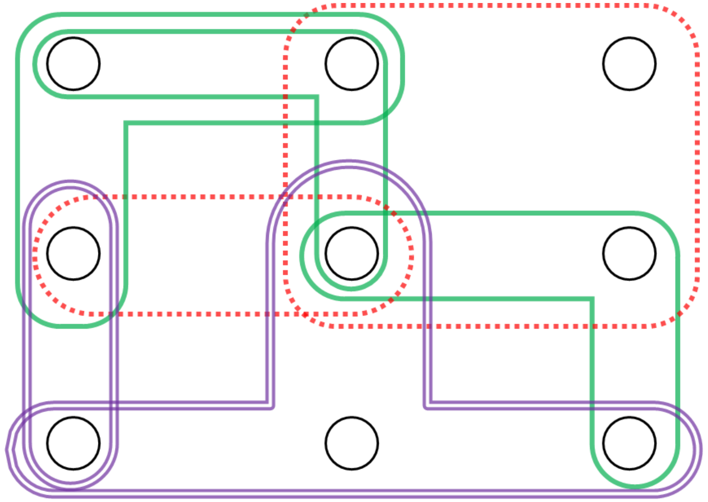
- **Goal**: minimize #mistakes
- Useful for clustering data with higher-order categorical interactions
- Widely studied:

  Amburg et al. (WWW'20; SDM'22), Veldt (ICML'23), Crane et al. (ICML'25)

- **Limitation**: enforces *nonoverlapping* and *exhaustive* clustering

  must assign exactly one color to every node

Edge-colored clustering (ECC)



- **Goal**: minimize #mistakes
- Useful for clustering data with higher-order categorical interactions
- Widely studied:

  Amburg et al. (WWW'20; SDM'22), Veldt (ICML'23), Crane et al. (ICML'25)

- **Limitation**: enforces *nonoverlapping* and *exhaustive* clustering

  must assign exactly one color to every node

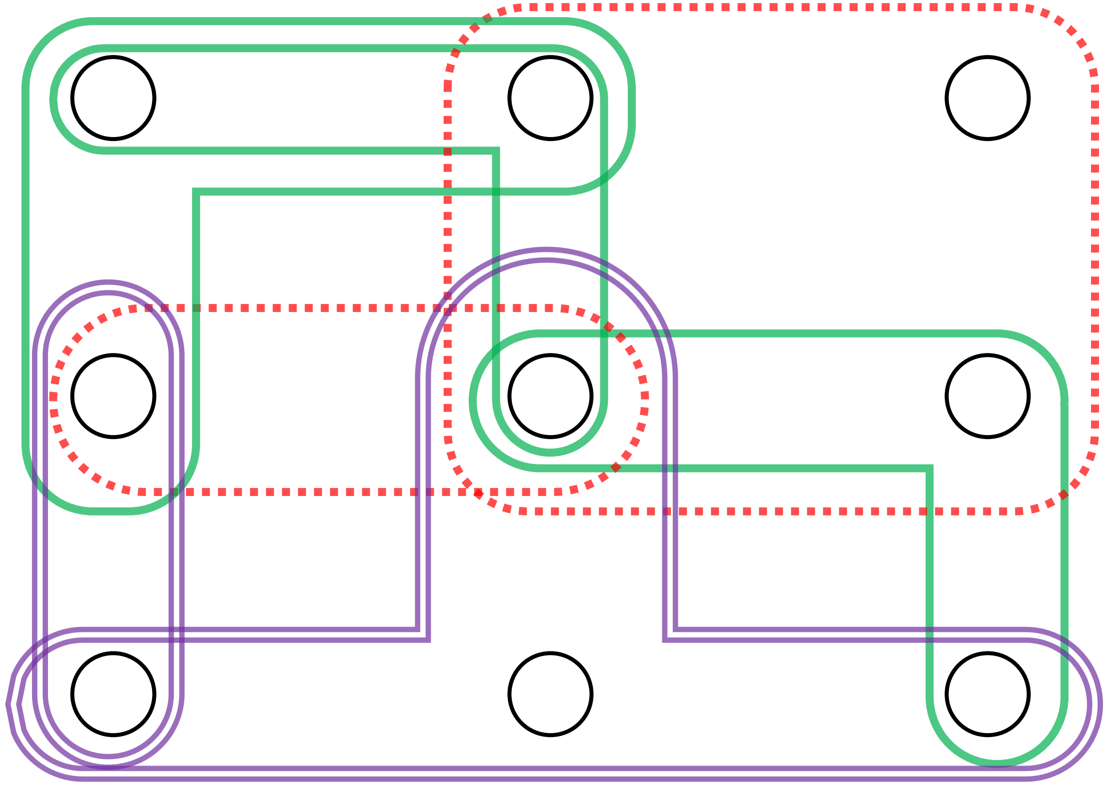
overlapping

- Local ECC, Global ECC, and Robust ECC proposed by Crane et al. relax these requirements (WSDM'24)

non-exhaustive

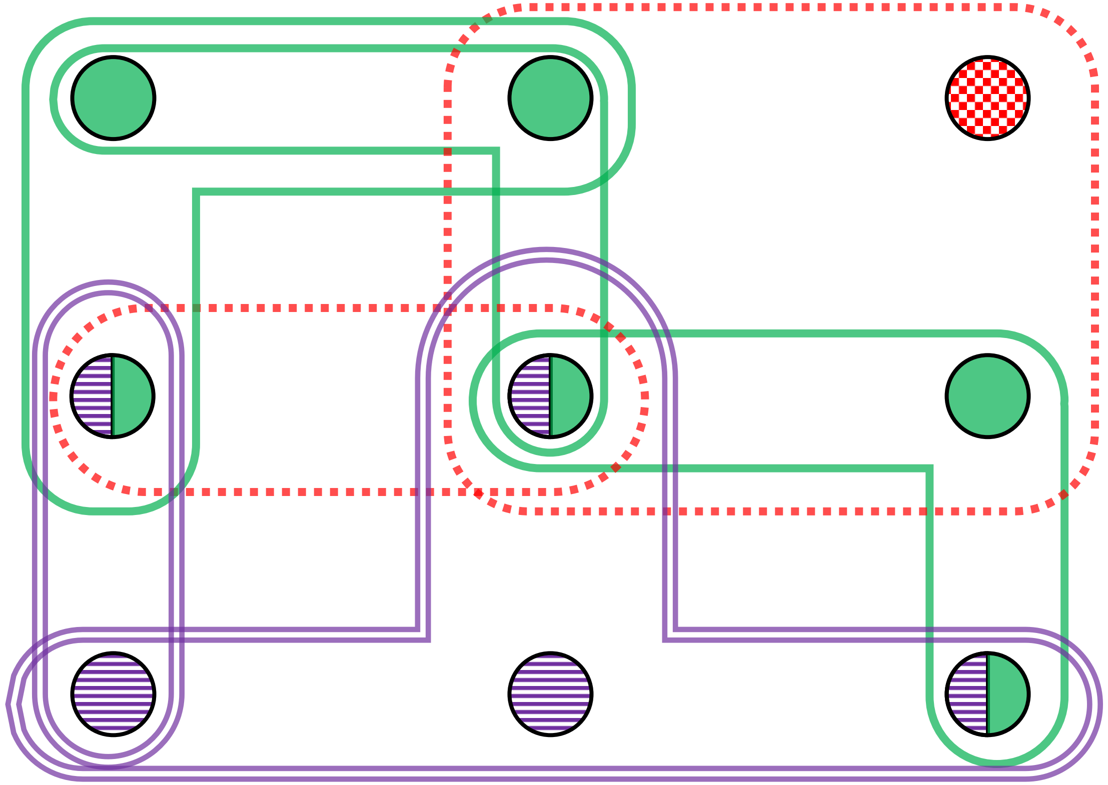
Local ECC

- A *local budget* b_{local} is given; assign (at most) b_{local} colors per node



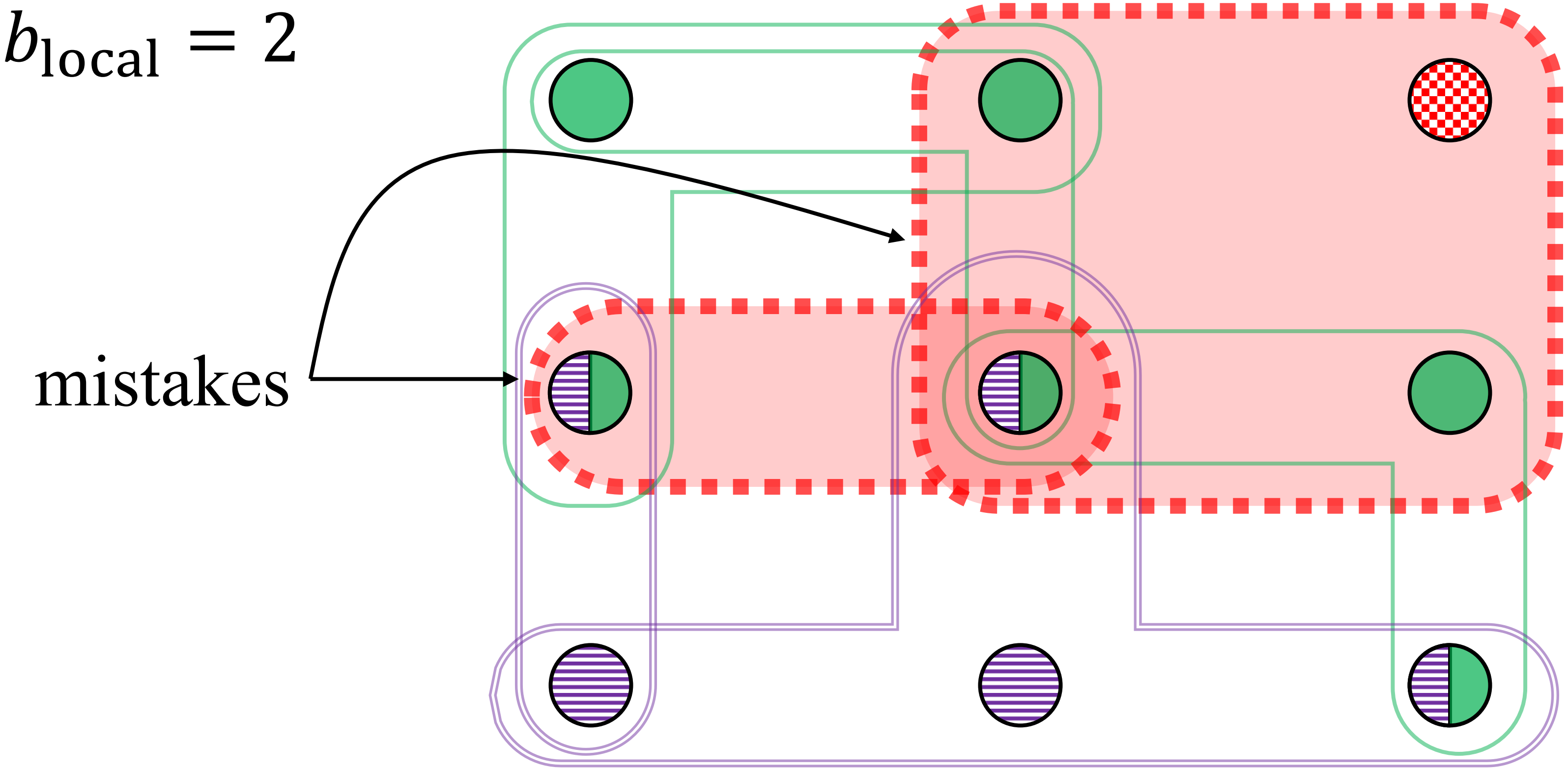
Local ECC

- A *local budget* b_{local} is given; assign (at most) b_{local} colors per node
- $b_{\text{local}} = 2$



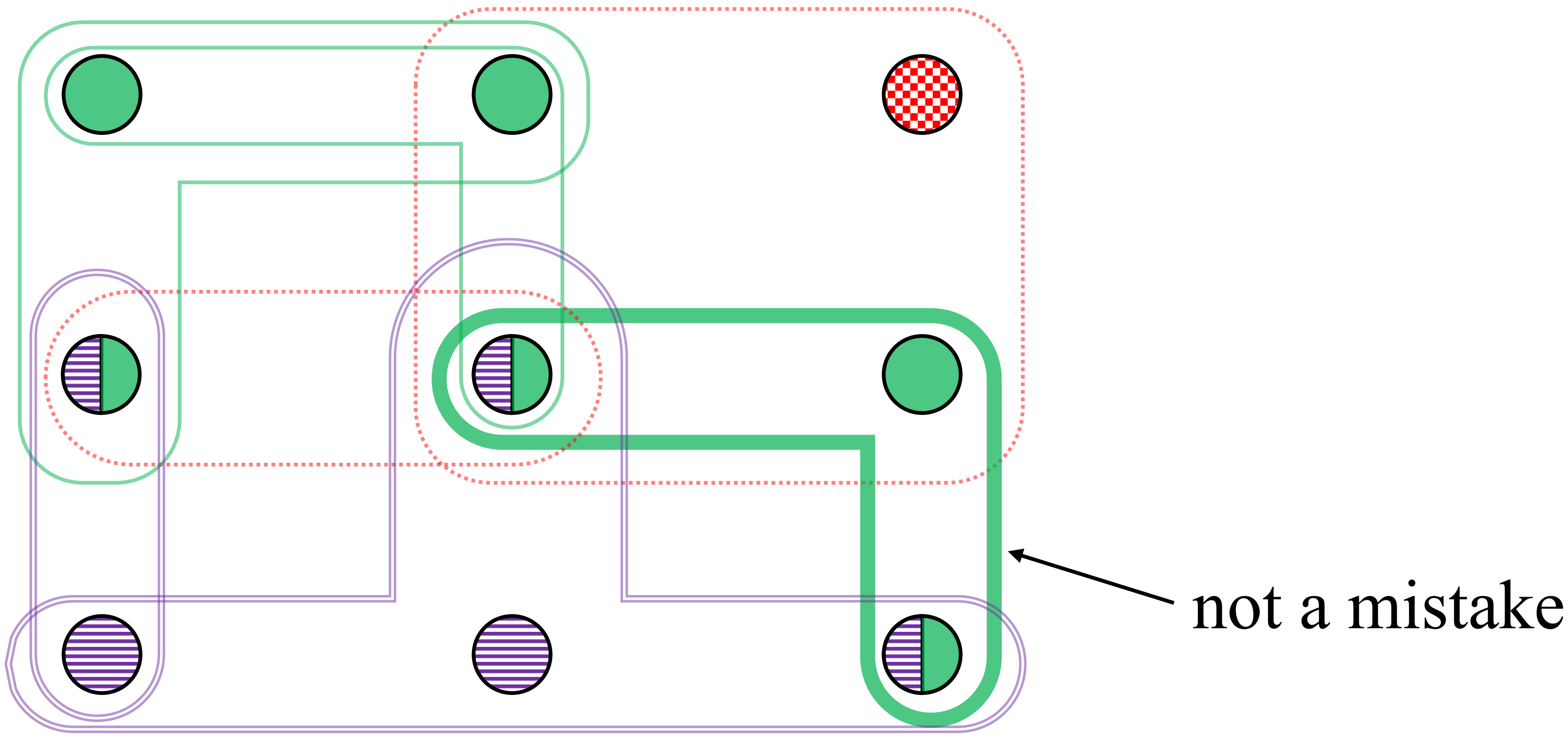
Local ECC

- A *local budget* b_{local} is given; assign (at most) b_{local} colors per node
- $b_{\text{local}} = 2$



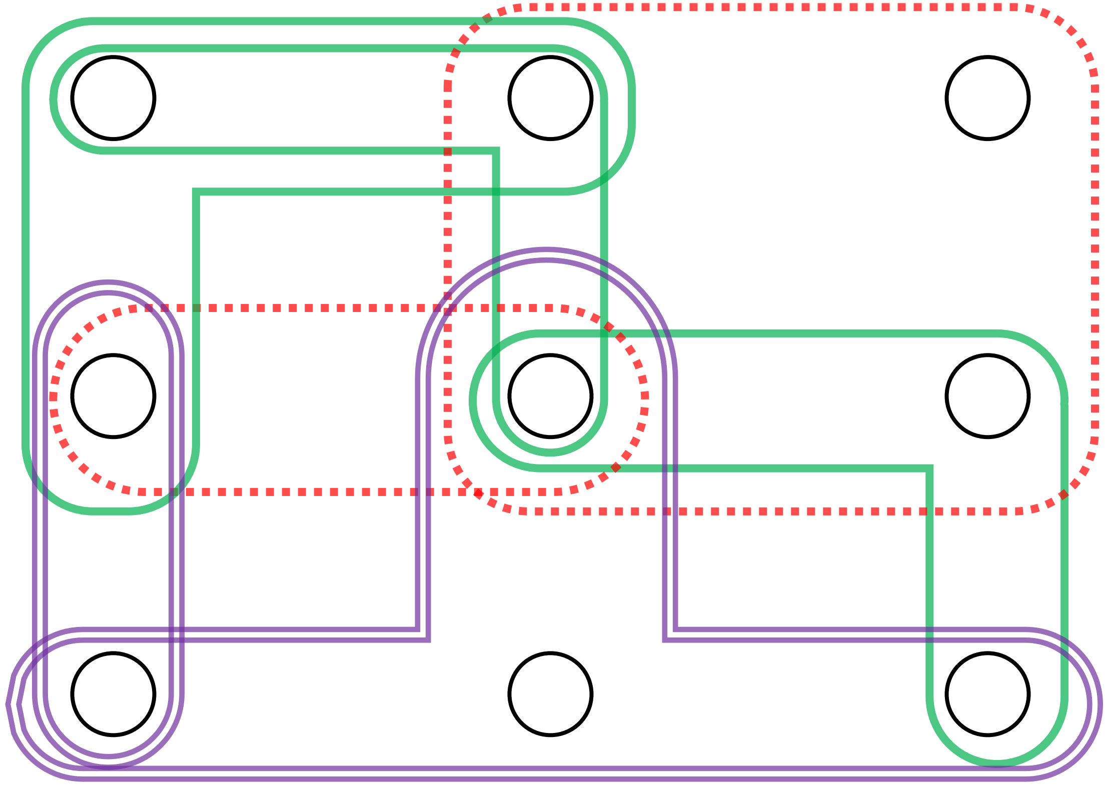
Local ECC

- A *local budget* b_{local} is given; assign (at most) b_{local} colors per node
- $b_{\text{local}} = 2$



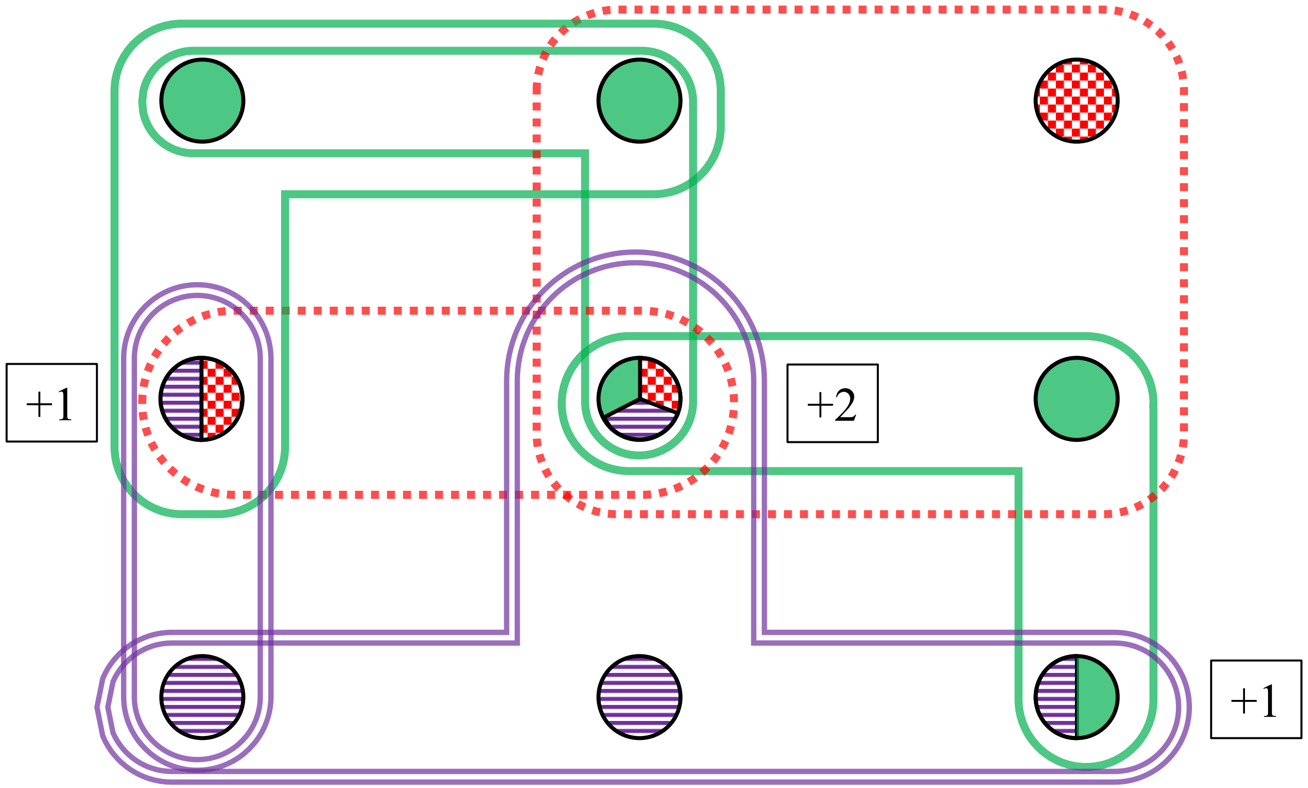
Global ECC

- A *global budget* b_{global} is given; assign additional b_{global} colors globally



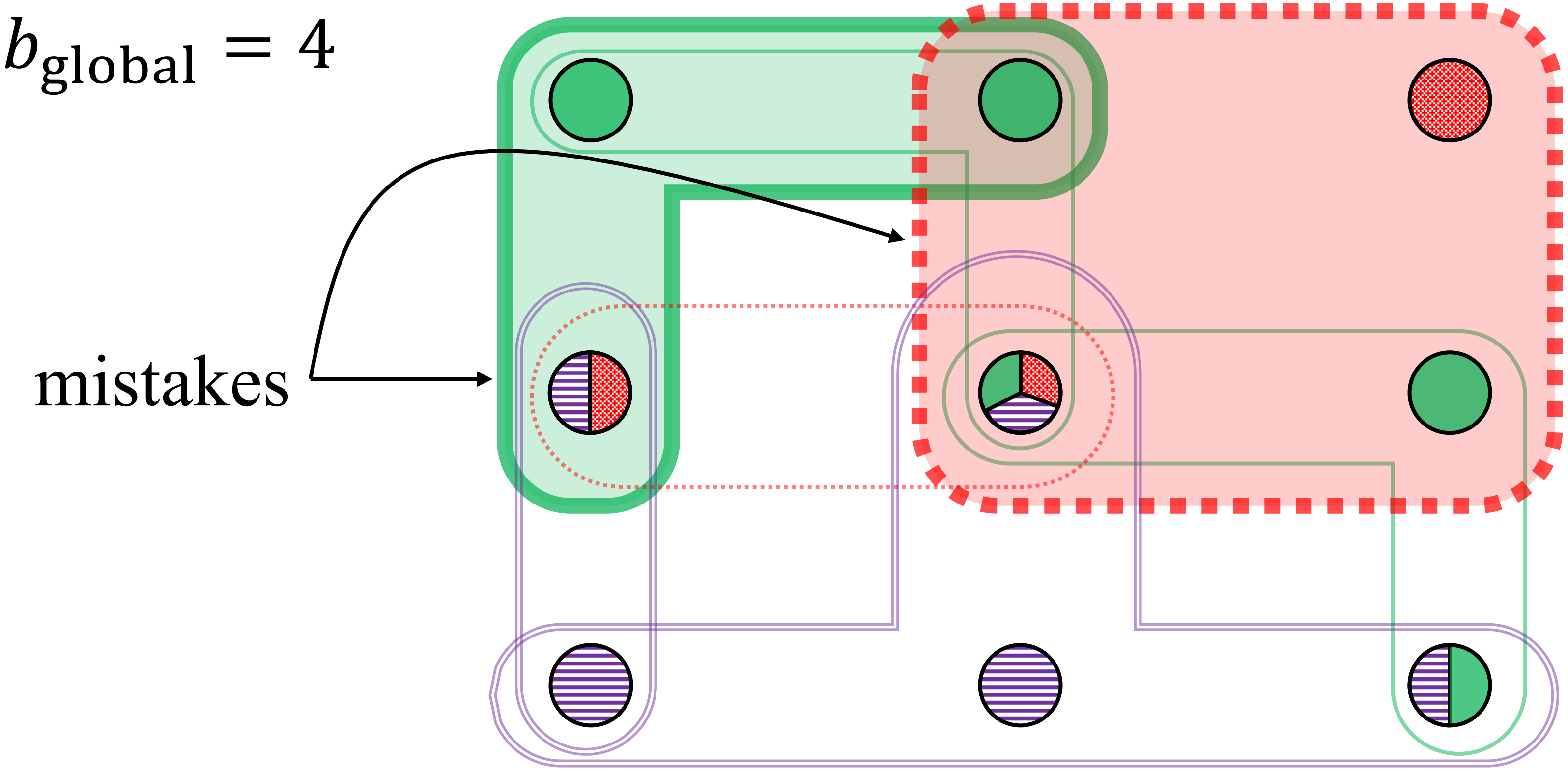
Global ECC

- A *global budget* b_{global} is given; assign additional b_{global} colors globally
- $b_{\text{global}} = 4$



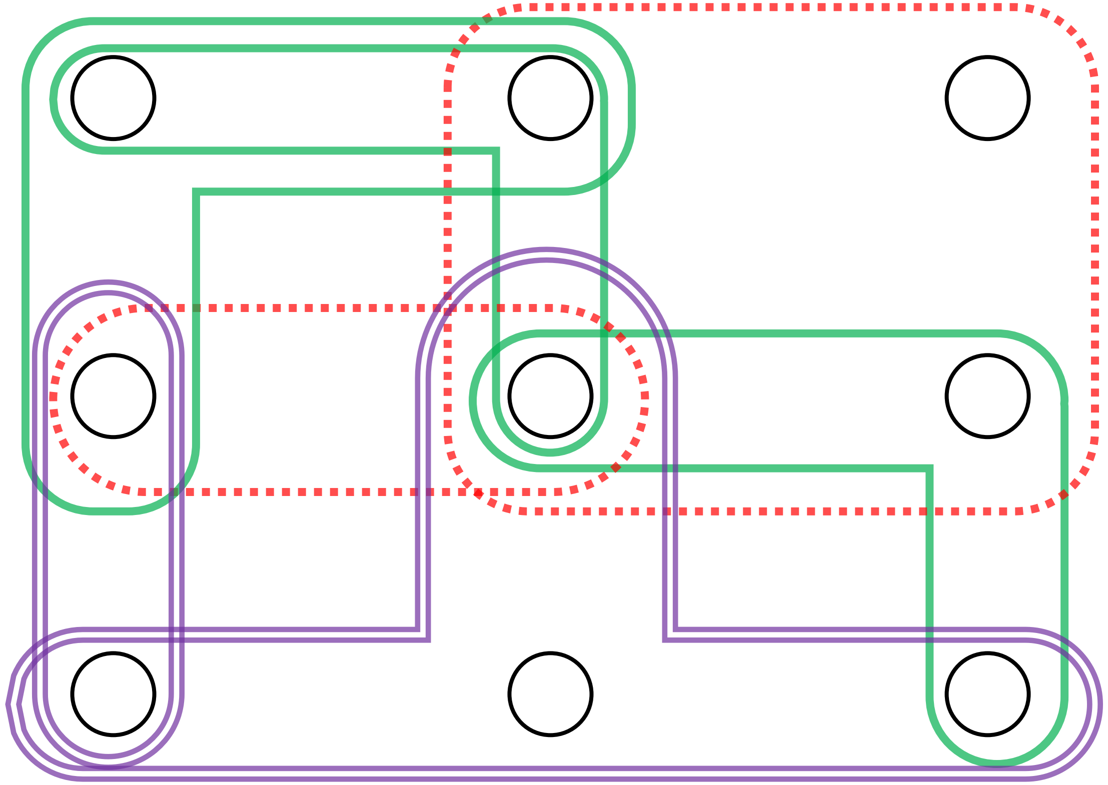
Global ECC

- A *global budget* b_{global} is given; assign additional b_{global} colors globally
- $b_{\text{global}} = 4$



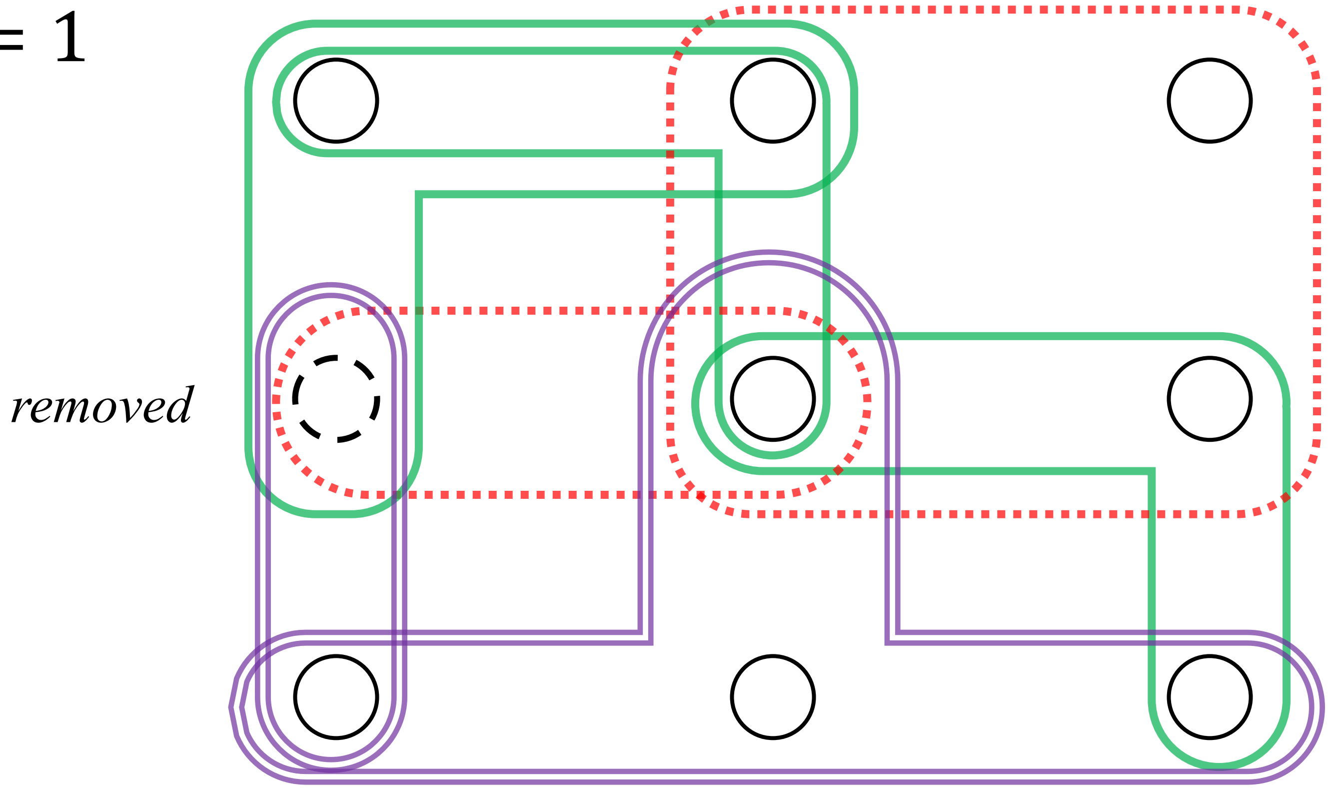
Robust ECC

- b_{robust} is given; may remove b_{robust} nodes before the color assignment



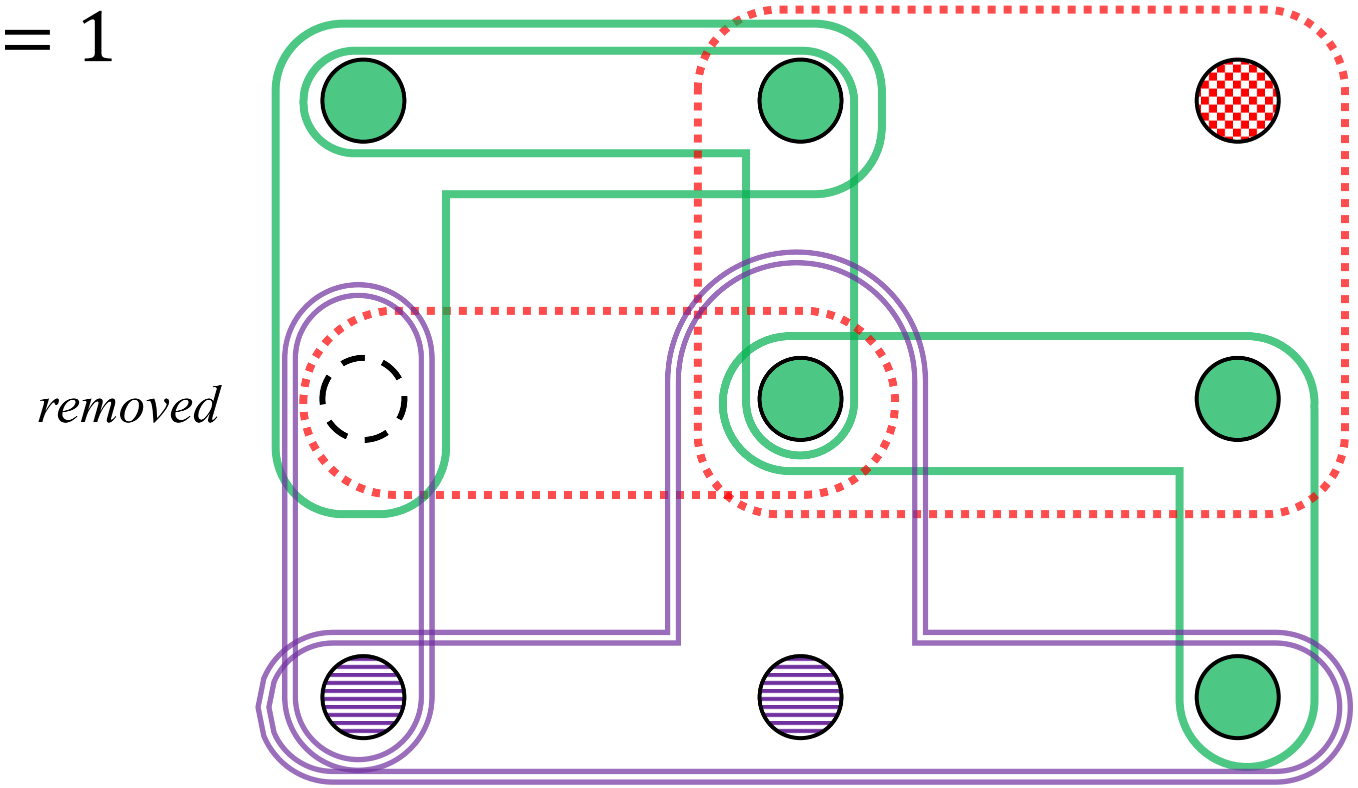
Robust ECC

- b_{robust} is given; may remove b_{robust} nodes before the color assignment
- $b_{\text{robust}} = 1$



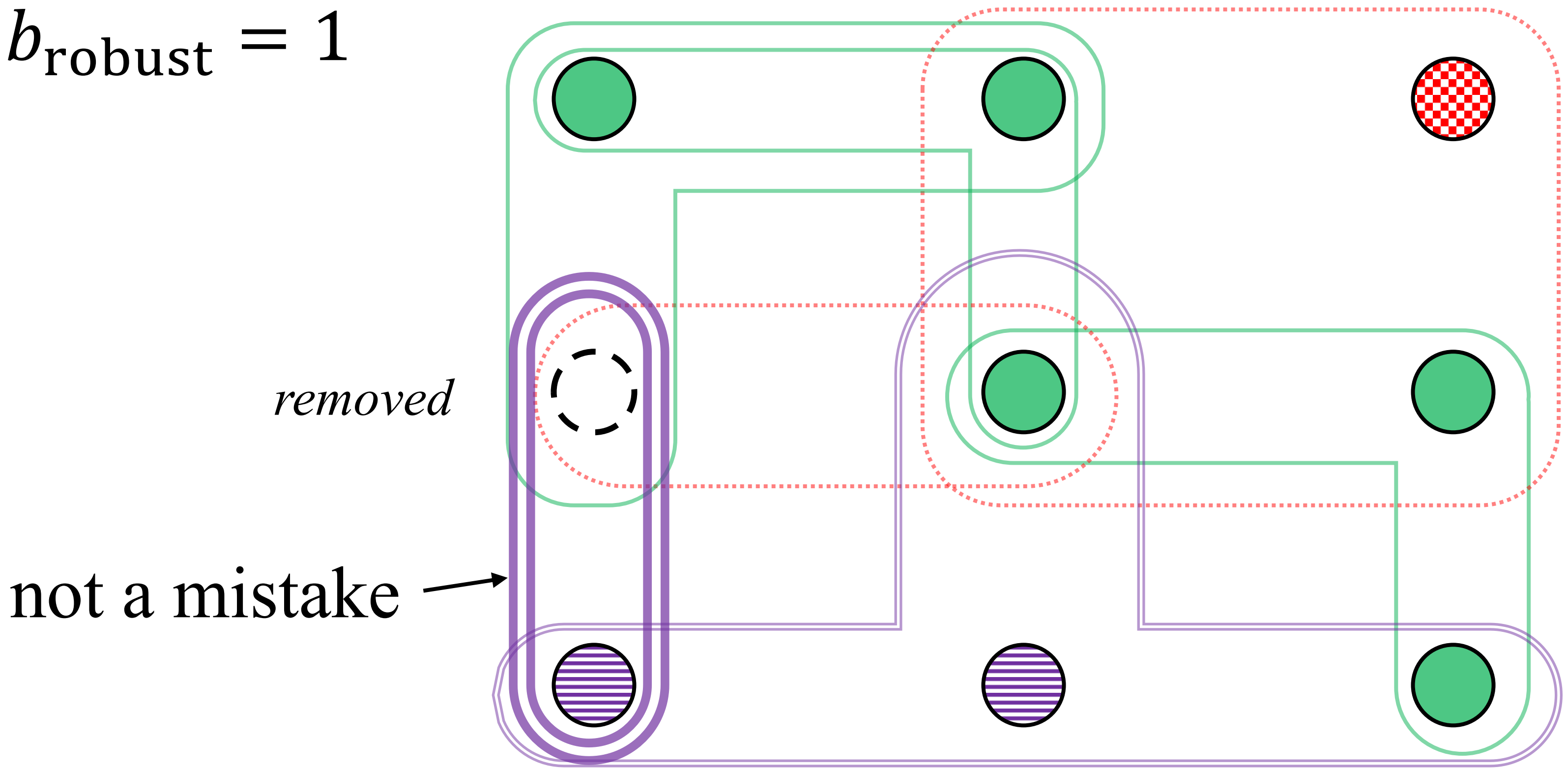
Robust ECC

- b_{robust} is given; may remove b_{robust} nodes before the color assignment
- $b_{\text{robust}} = 1$



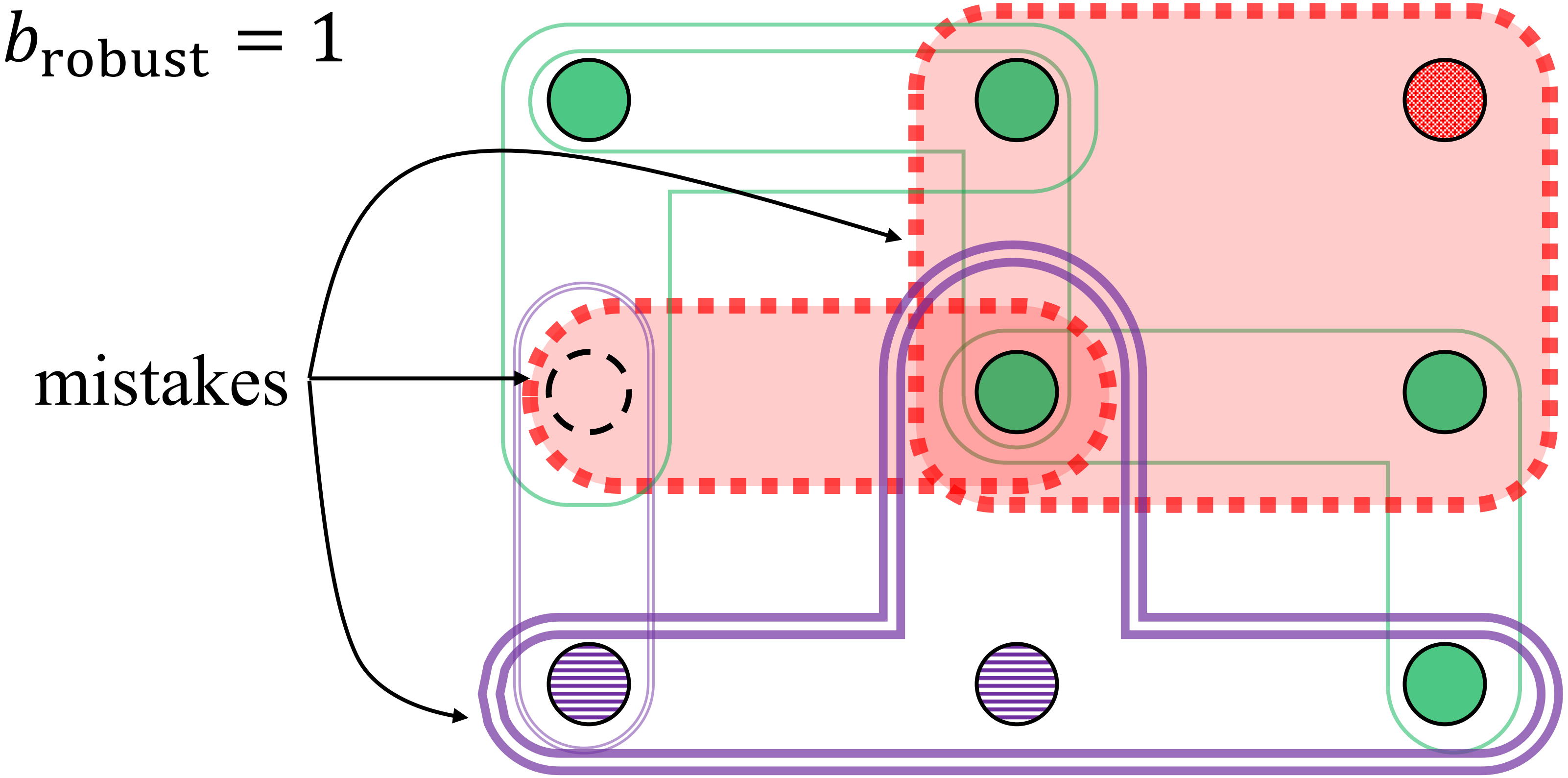
Robust ECC

- b_{robust} is given; may remove b_{robust} nodes before the color assignment
- $b_{\text{robust}} = 1$



Robust ECC

- b_{robust} is given; may remove b_{robust} nodes before the color assignment
- $b_{\text{robust}} = 1$



Previous results

- **LP-rounding algorithms** [Crane et al., 2024]
  - solve a linear program (LP)
  - convert the LP solution into an actual coloring

- **Greedy algorithms** [Crane et al., 2024]
  - combinatorial algorithms

Previous results

- **LP-rounding algorithms** [Crane et al., 2024]
  - solve a linear program (LP)
  - convert the LP solution into an actual coloring
  - high solution quality (but slow)

- **Greedy algorithms** [Crane et al., 2024]
  - combinatorial algorithms
  - remarkably fast (but often with low solution quality)

Previous results

$r \coloneqq \max|e|$

- Local ECC
  - $(b_{\text{local}} + 1)$-approx. alg. (LP-rounding); r-approx. alg. (greedy) [Crane et al., 2024]

- Global ECC
  - *bicriteria* approx. alg. (LP-rounding); r-approx. alg. (greedy) [Crane et al., 2024]

- Robust ECC
  - *bicriteria* approx. alg. (LP-rounding); r-approx. alg. (greedy) [Crane et al., 2024]

Our results

- **Proposed algorithms: LP-based & combinatorial**
    - a *primal-dual* method
    - combine the strengths of both worlds

Our results

- **Proposed algorithms: LP-based & combinatorial**
  - a *primal-dual* method
  - combine the strengths of both worlds

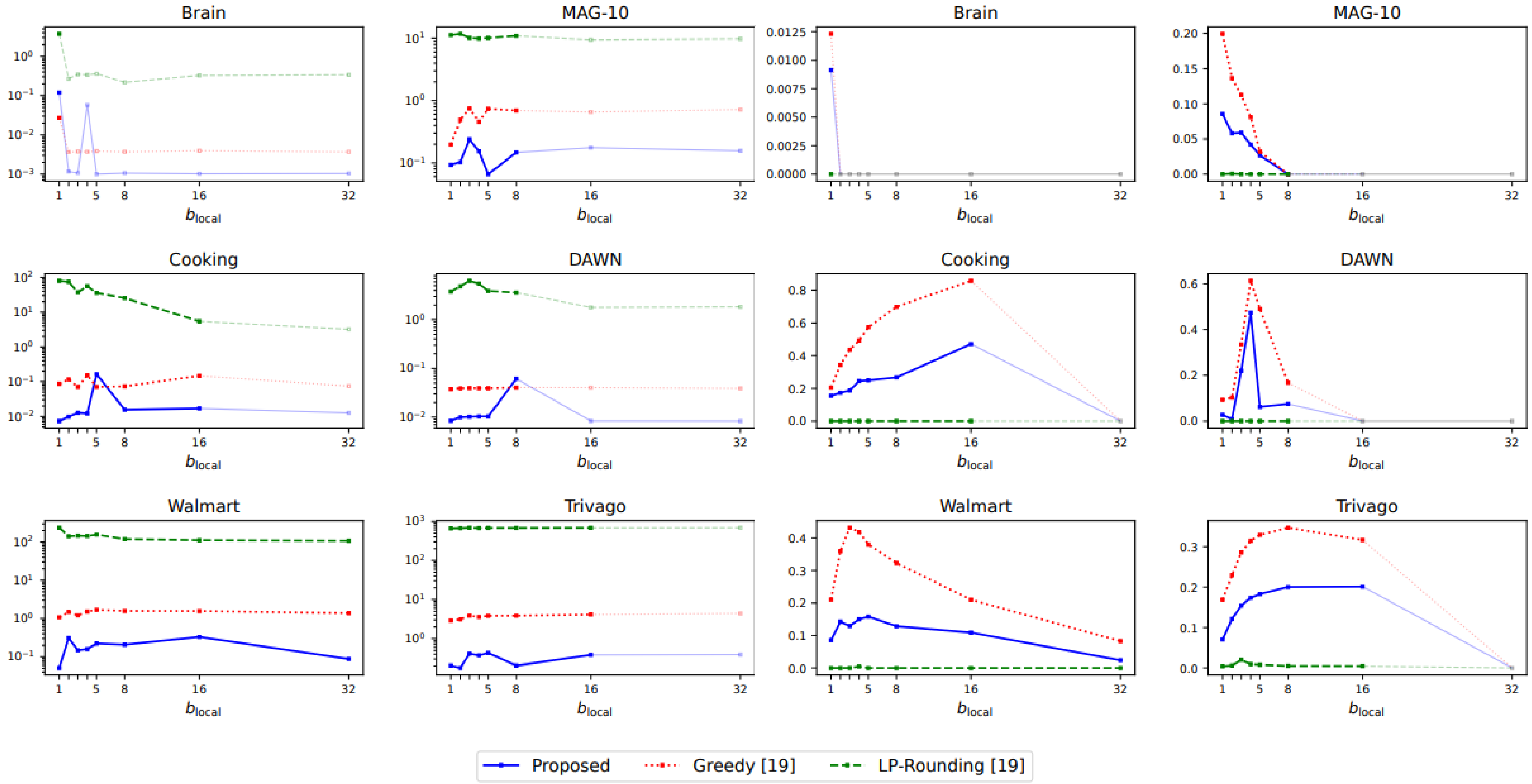
- Computational evaluation:
  - achieve **better solution quality** than greedy algs.
  - significantly **faster** than LP-rounding (bicriteria) algs.

Our results

$r \coloneqq \max|e|$

- Local ECC
  - $(b_{\text{local}} + 1)$-approx. alg. (LP-rounding); r-approx. alg. (greedy) [Crane et al., 2024]
  - **$(\boldsymbol{b}_{\text{local}} + 1)$-approx. alg. (combinatorial)**
- Global ECC
  - bicriteria approx. alg. (LP-rounding); r-approx. alg. (greedy) [Crane et al., 2024]
  - **$(2\boldsymbol{b}_{\text{global}} + 2)$-approx. alg. (combinatorial)**
- Robust ECC
  - bicriteria approx. alg. (LP-rounding); r-approx. alg. (greedy) [Crane et al., 2024]
  - **$(2\boldsymbol{b}_{\text{robust}} + 2)$-approx. alg. (combinatorial)**

Paper URL



(a) Running times ($\log_{10}$ scaled, in seconds)

(b) Relative error estimates

Figure 1: (a) Running times (in seconds, log scale) and (b) relative error estimates of the LOCAL ECC algorithms. Empty square markers denote trivial instances.

Paper URL

Table 2: Average running times of each dataset (in seconds): LOCAL ECC. Values in parentheses are averages excluding trivial instances.

	Proposed		Greedy		LP-rounding	
Brain	0.023	(0.120)	0.007	(0.028)	0.743	(3.739)
MAG-10	0.142	(0.134)	0.587	(0.554)	10.413	(10.677)
Cooking	0.032	(0.035)	0.099	(0.103)	39.702	(44.916)
DAWN	0.016	(0.019)	0.040	(0.040)	3.948	(4.658)
Walmart	0.190	(0.190)	1.443	(1.443)	145.427	(145.427)
Trivago	0.323	(0.313)	3.709	(3.608)	678.585	(677.036)

Our results

$r := \max|e|$

- Local ECC
  - $(b_{\text{local}} + 1)$-approx. alg. (LP-rounding); r-approx. alg. (greedy) [Crane et al., 2024]
  - **$(\boldsymbol{b}_{\textbf{local}} + \mathbf{1})$-approx. alg. (combinatorial); integrality gap $\cong \boldsymbol{b}_{\textbf{local}} + \mathbf{1}$**
  - **$(\boldsymbol{b}_{\textbf{local}} + \mathbf{1})$-approx. is *essentially optimal***

answers an open question of Crane et al.: $O(1)$-approx. for Local ECC?

- Global ECC
  - bicriteria approx. alg. (LP-rounding); r-approx. alg. (greedy) [Crane et al., 2024]
  - **$(\mathbf{2}\boldsymbol{b}_{\textbf{global}} + \mathbf{2})$-approx. alg. (combinatorial); integrality gap $\geq \boldsymbol{b}_{\textbf{global}} + \mathbf{1}$**
- Robust ECC
  - bicriteria approx. alg. (LP-rounding); r-approx. alg. (greedy) [Crane et al., 2024]
  - **$(\mathbf{2}\boldsymbol{b}_{\textbf{robust}} + \mathbf{2})$-approx. alg. (combinatorial); integrality gap $\geq \boldsymbol{b}_{\textbf{robust}} + \mathbf{1}$**

Additional results

- The proposed algorithm for Local ECC works in the *online* setting

a vertex arrives and must color it irrevocably

- All proposed algs. admit a bicriteria approx. factor of $(O(1), O(1))$

answers an open question of Crane et al.:
bicriteria $(O(1), O(1))$-approx. for Global ECC?

Paper URL

Thank you for listening!