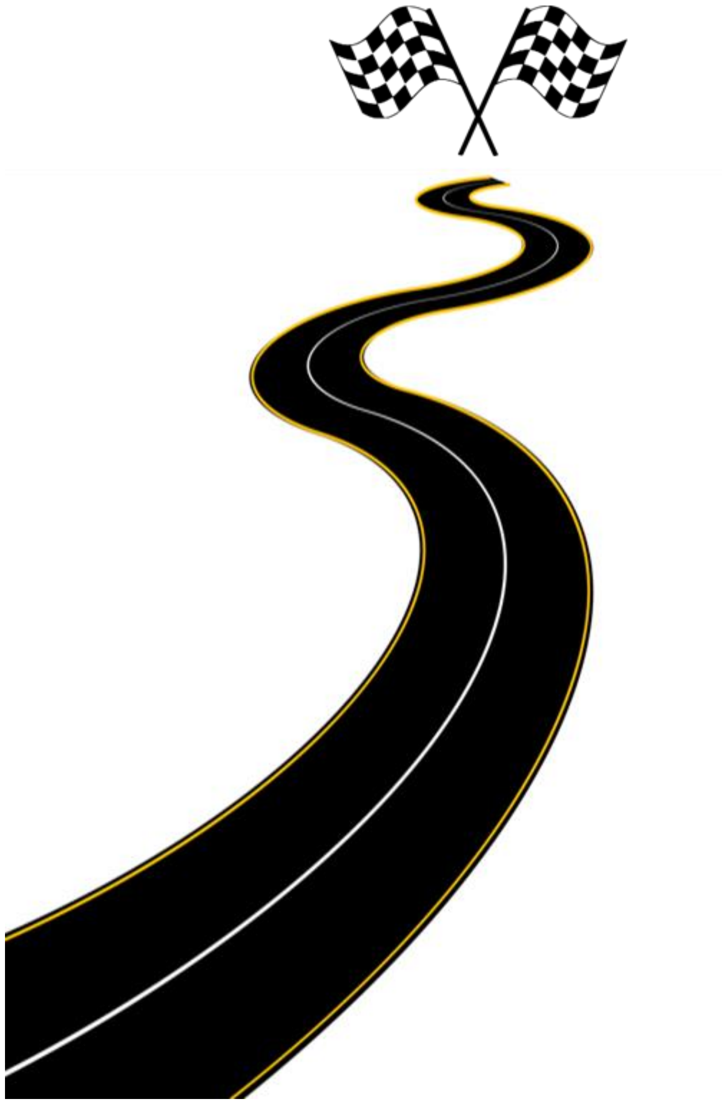
Improving the Core Resilience of Real-world Hypergraphs: Concepts, Observations, and Algorithms

Manh Tuan Do

Kijung Shin

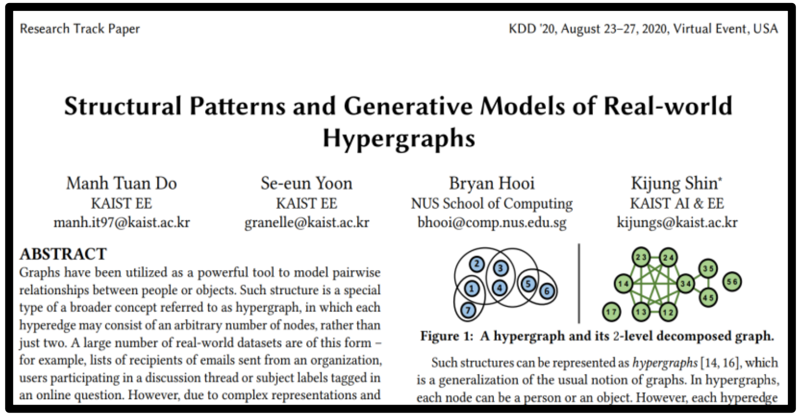
Roadmap

1. **Preliminary <<**
2. Observations
3. Method
4. Experiments
5. Conclusion



Motivations

- Group interactions are everywhere!


Research Track Paper

KDD '20, August 23–27, 2020, Virtual Event, USA

Structural Patterns and Generative Models of Real-world Hypergraphs

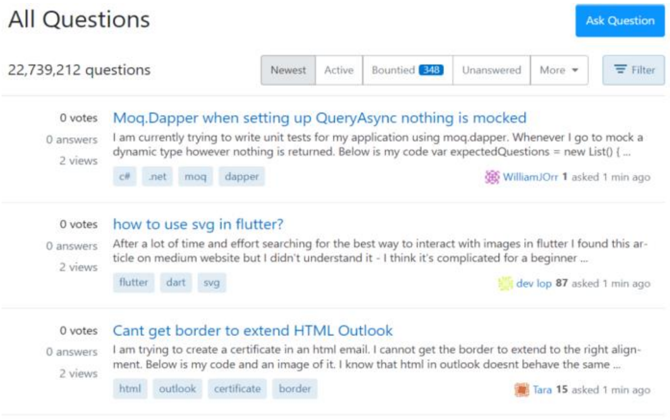
Manh Tuan Do, KAIST EE, manh.it97@kaist.ac.kr
Se-eun Yoon, KAIST EE, granelle@kaist.ac.kr
Bryan Hooi, NUS School of Computing, bhooi@comp.nus.edu.sg
Kijung Shin*, KAIST AI & EE, kijungs@kaist.ac.kr

ABSTRACT

Graphs have been utilized as a powerful tool to model pairwise relationships between people or objects. Such structure is a special type of a broader concept referred to as hypergraph, in which each hyperedge may consist of an arbitrary number of nodes, rather than just two. A large number of real-world datasets are of this form – for example, lists of recipients of emails sent from an organization, users participating in a discussion thread or subject labels tagged in an online question. However, due to complex representations and

Figure 1: A hypergraph and its 2-level decomposed graph.

Such structures can be represented as *hypergraphs* [14, 16], which is a generalization of the usual notion of graphs. In hypergraphs, each node can be a person or an object. However, each hyperedge



Paper co-authors

Email sender/recipients

Thread participants

Hypergraphs

- A **Hypergraph** $G = (V, E)$ consists of a node set V and a hyperedge set E
  - Each hyperedge constitutes a set of nodes
- Hypergraphs naturally represent *group interactions* among people/objects

Authors (Nodes)

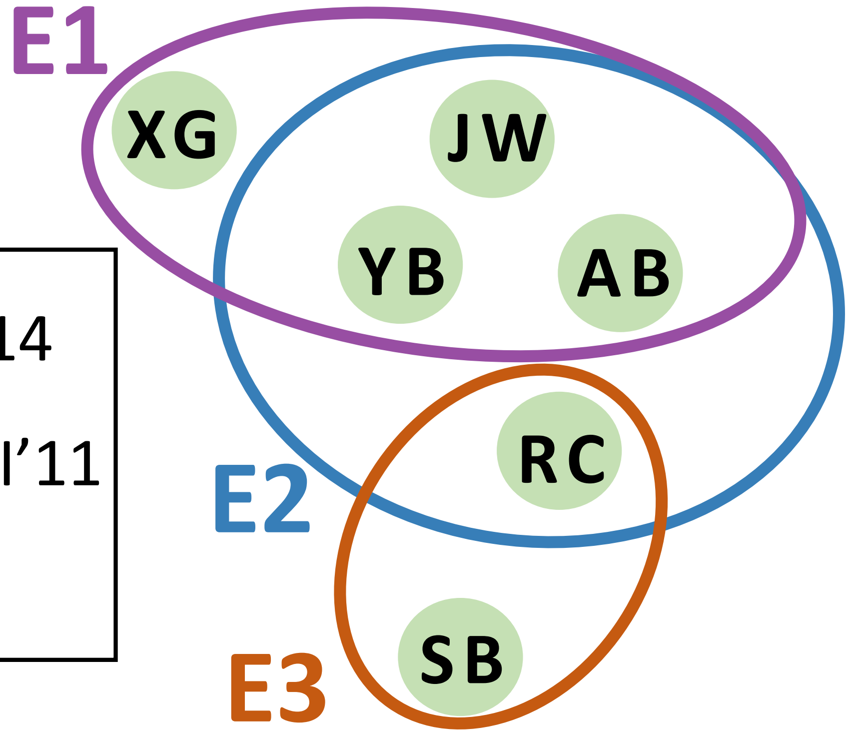
Y Bengio (**YB**)	J Weston (**JW**)
X Glorot (**XG**)	A Bordes (**AB**)
R Collobert (**RC**)	S Bengio (**SB**)

Publications (Hyperedges)

E1: (**AB**, **XG**, **JW**, **YB**) ML'14

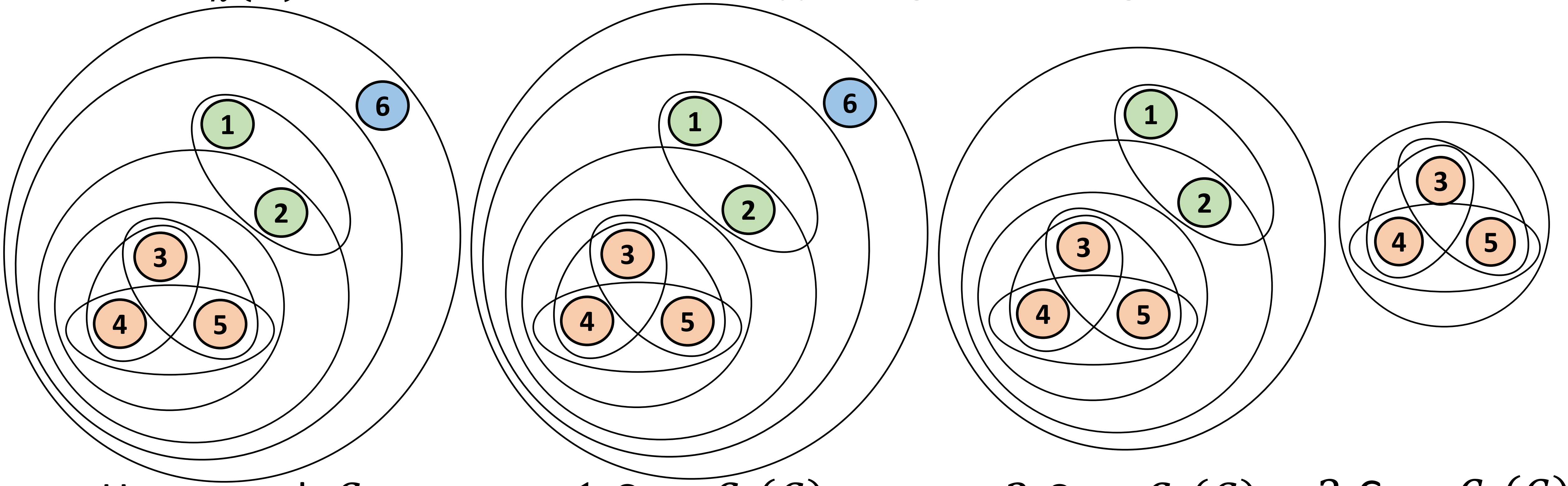
E2: (**AB**, **JW**, **RC**, **YB**) AAAI'11

E3: (**RC**, **SB**) JML'01



Core: Definition

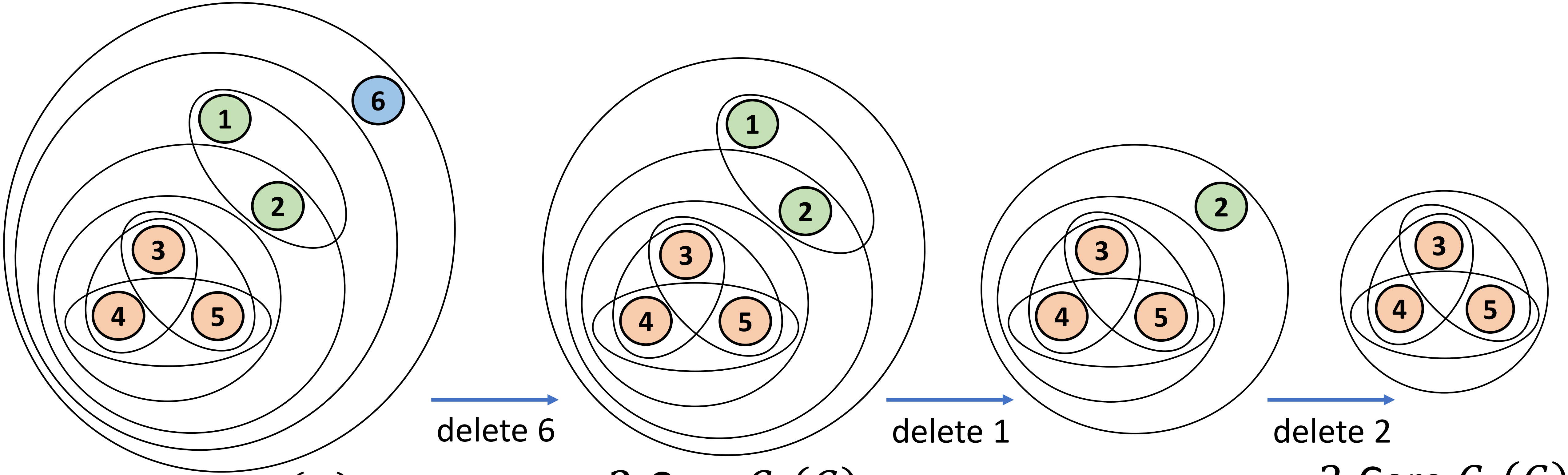
- Given $G = (V, E)$ and $k \in \mathbb{N}$
- The ***k*-Core** $C_k(G)$ of G is the *maximal sub-hypergraph* where each node in $C_k(G)$ is incident to at least k hyperedges, i.e., degree at least k.



Hypergraph G 1-Core $C_1(G)$ 2-Core $C_2(G)$ 3-Core $C_3(G)$

Core Decomposition Process

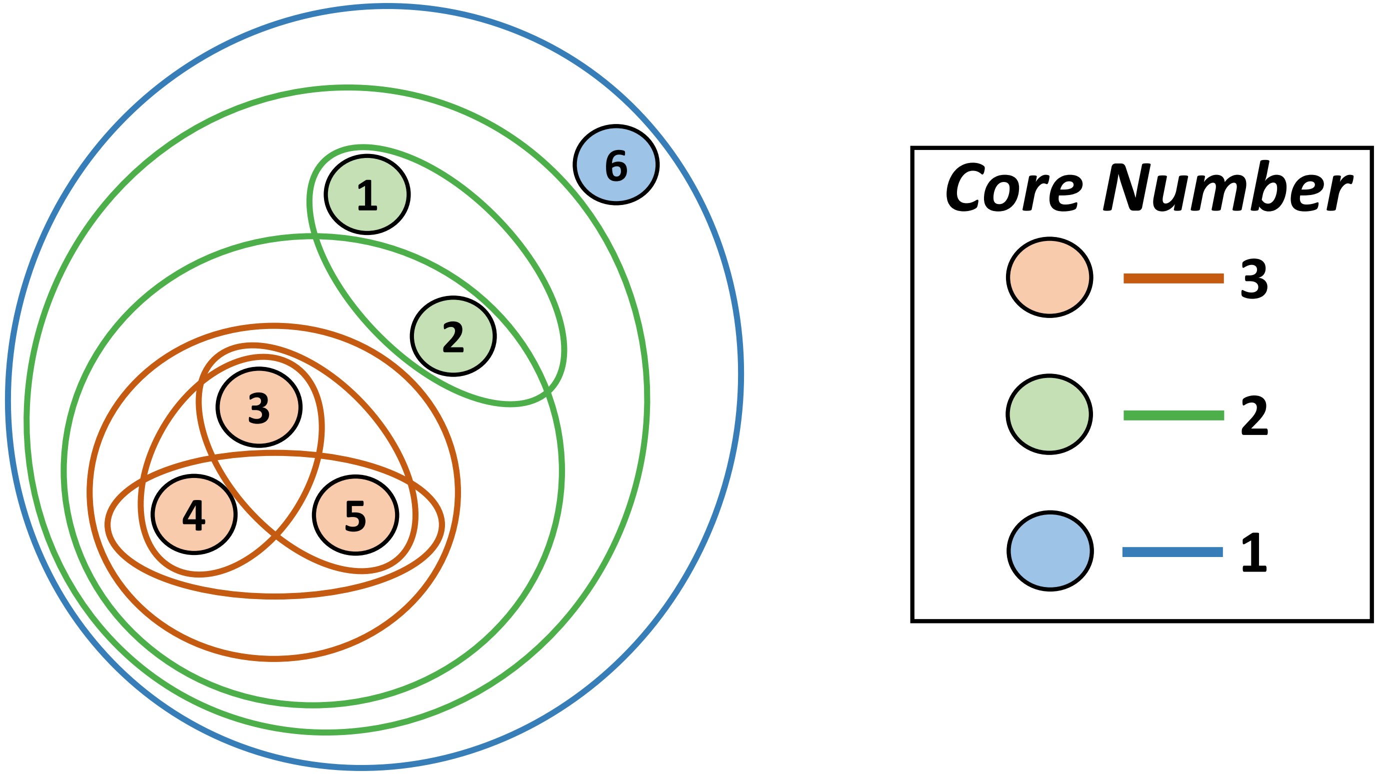
- The process of obtaining the cores of hypergraph $G = (V, E)$.
- In order to obtain the ***k*-Core** $C_k(G)$ of G, repeatedly remove the nodes having degree lower than k until that is no longer possible.



1-Core $C_1(G)$ 2-Core $C_2(G)$ 3-Core $C_3(G)$

Related Concepts: Core Number

- The **core number** of a node v is the maximum k s.t. v is in the k-core
- The **core number** of a hyperedge e is the maximum k s.t. e is in the k-core

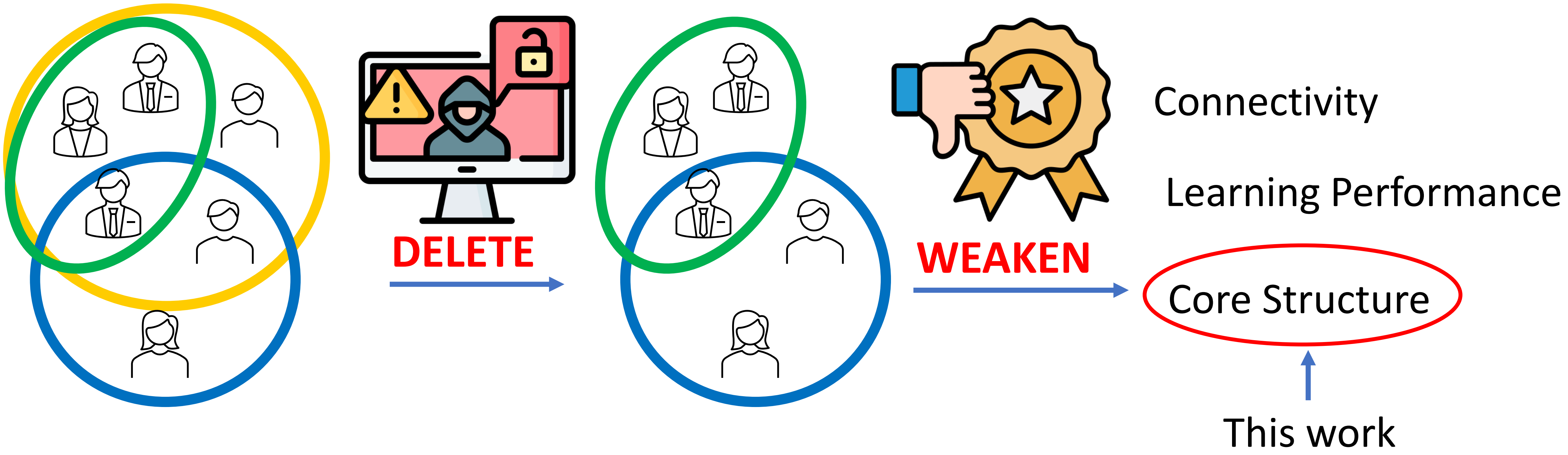


Scenario: System Attack

- Database damages by malicious penetration
  - Attackers may **remove** several records on the database, resulting in **information loss**.

Network Attacks

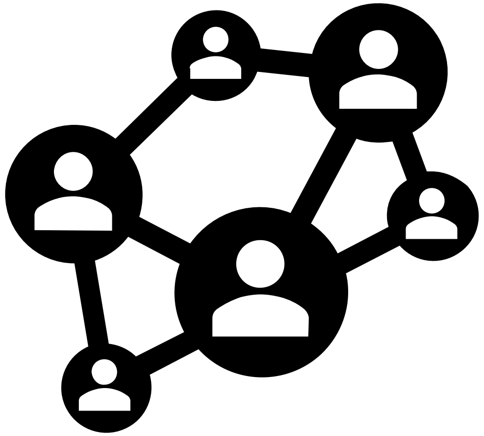
- Deletion attacks on a hypergraph G
  - Attackers may **remove** a portion of the nodes/hyperedges of G to **impair** some *properties/applications* of G.



Why k-Core and Core Resilience?

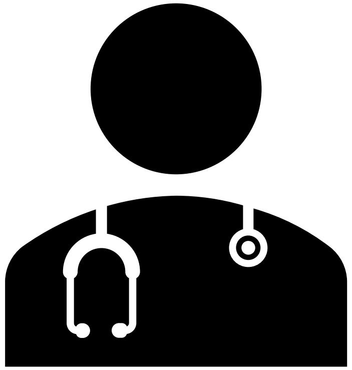
- **Core Resilience** of a hypergraph G
  - The **Spearman's rank correlation** of the rankings of nodes in terms of core numbers *before* and *after* a portion of G has been **removed**.

- Practical Applications of Core-based Ranking:



Finding Influential Nodes

Anomaly Detection



Medical Research

Problem Statement: Improving Core Resilience

- **Given:** hypergraph $G = (V, E)$, a budget $\mathrm{B} \in \mathbb{N}$
- **Find:** at most B hyperedges to augment to G
- **To Maximize:** the core resilience of G against deletion attacks
- **Subject to Constraints:**
  - All **core numbers** are all conserved
  - The original **hyperedge size distribution** is conserved

"Which B node combinations should we augment to <u>maximize the core resilience</u> and <u>preserve all core numbers</u> of G?"

Problem on Graphs: Improving Core Resilience

- **Given:** graph $G = (V, E)$, a budget $\mathrm{B} \in \mathbb{N}$
- **Find:** at most B edges to augment to G
- **To Maximize:** the core resilience of G against deletion attacks
- **Subject to Constraints:**
  - All **core numbers** are all conserved

- Addressed in: *Measuring and Improving the Core Resilience of Networks.* Ricky Laishram et al., 2018. The Web Conference.

Challenges of the Problem

- A naïve greedy algorithm
  - Among **all possible combinations of nodes**
    - Select those *guaranteed to preserve all core numbers* if added
  - Among the set of **all possible combinations of nodes selected**
    - Choose a subset that *maximizes the core resilience* if added

$O(2^{|V|})$ time!

(V: set of nodes.)

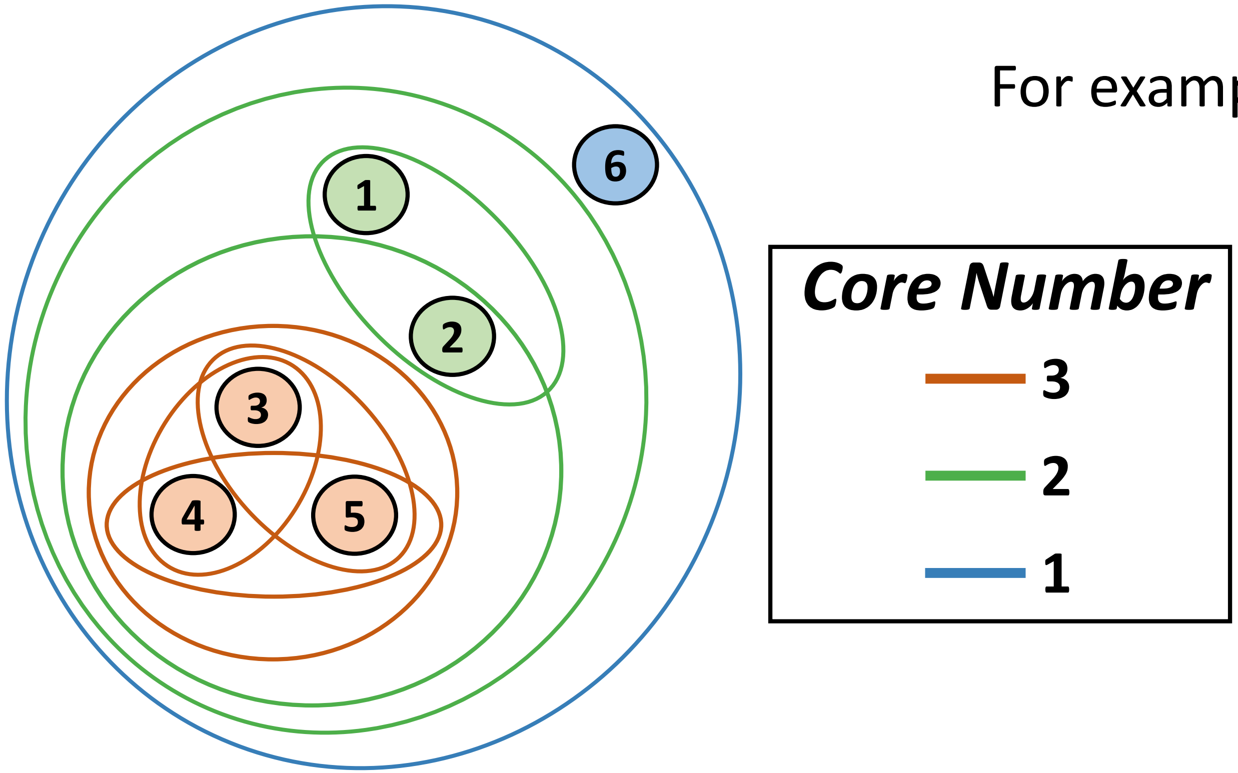
- **Challenge 1**: **Prohibitive Cost** of iterating through all node combinations.
- **Challenge 2**: **Computational Difficulty** of measuring & optimizing core resilience, a hypergraph-level measurement.

Roadmap

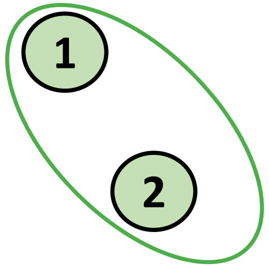
1. Preliminary
2. **Observations <<**
3. Method
4. Experiments
5. Conclusion

Characterizing core-beneficiary nodes of a hyperedge

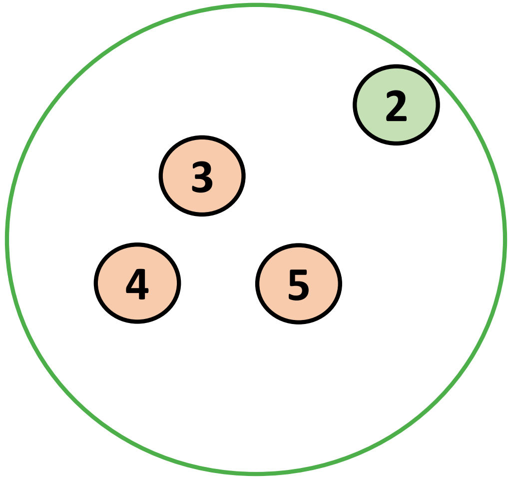
- The **anchor(s)** of a hyperedge e is (are) the node(s) $\{v|v \in e\}$ s.t. each v has the same core number as e.
- I.e., the nodes having the lowest core number in e.



For example:



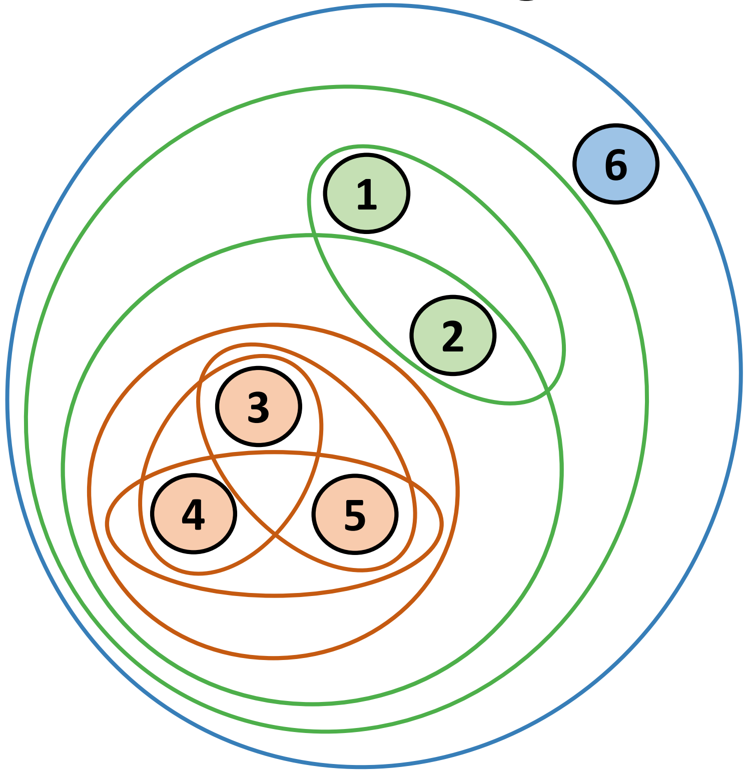
1 and 2 are the anchors



2 is the anchor

Characterizing Core Resilience at the Node level

- The **core strength** $\boldsymbol{CS}$ of a node v measures how robust v is in keeping its core number against hyperedge deletions. *
- The **core influence** $\boldsymbol{CI}$ of a node v measures how v contributes to the core numbers of neighbor nodes. *



Core Strength	
1	1
2	1
3	1
4	1
5	1
6	1

Core Influence	
1	5/2
2	5/2
3	21/4
4	21/4
5	21/4
6	1

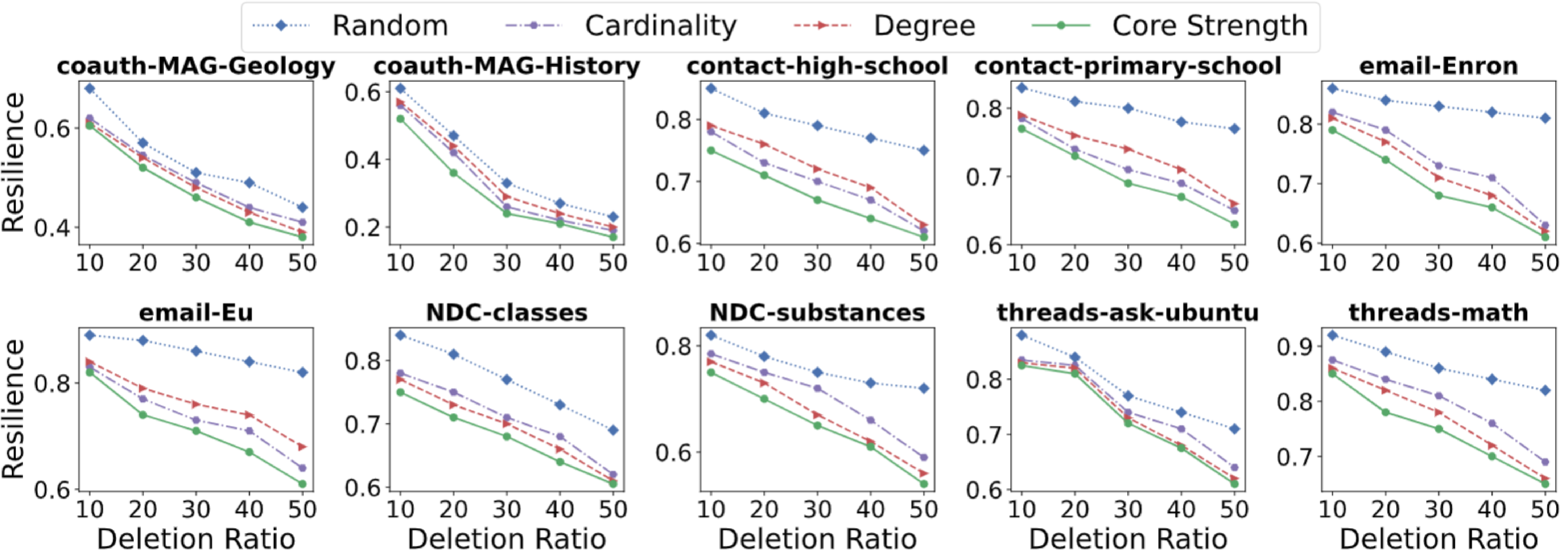
* Refer to the paper for exact formulas.

Attack Schemes: how hyperedges may be deleted.

- **Random:** delete hyperedges uniformly at random
- **Cardinality:** target large-size hyperedges
- **Degree:** delete hyperedges incident to high-degree nodes
- **Core Strength:** delete hyperedges incident to low core-strength nodes.

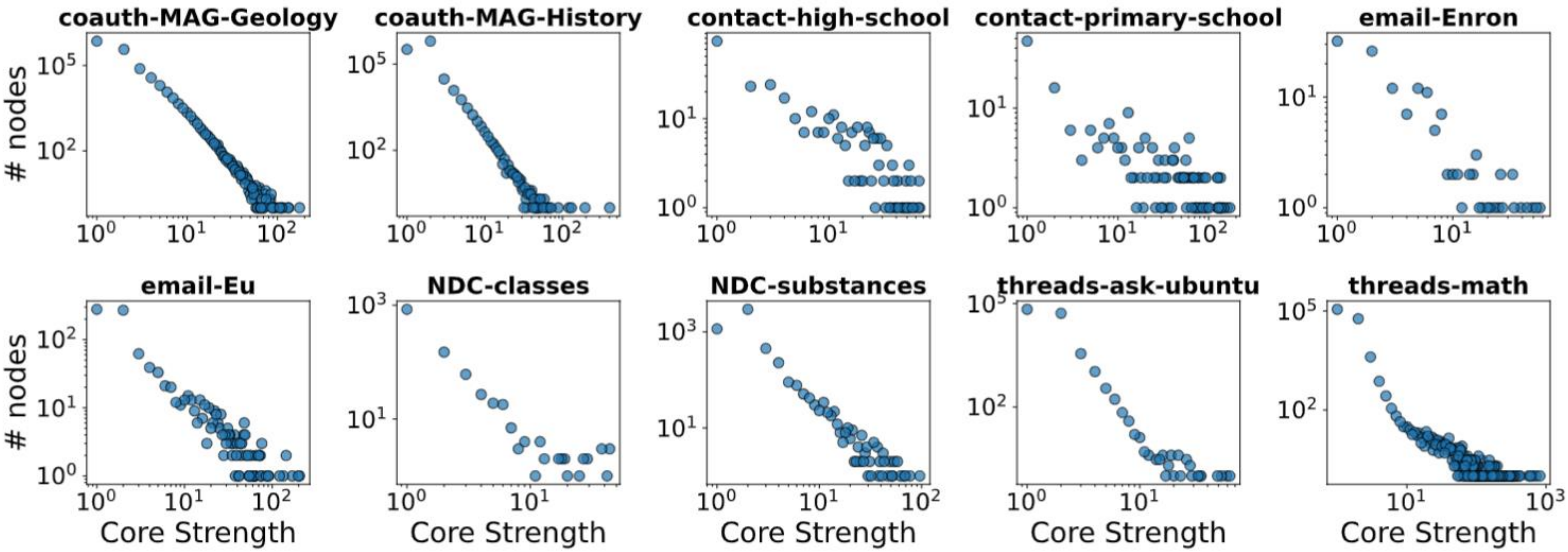
Observation 1: Resilience against attacks

- **Random** is consistently the least damaging while **Core Strength** is consistently the most destructive scheme.



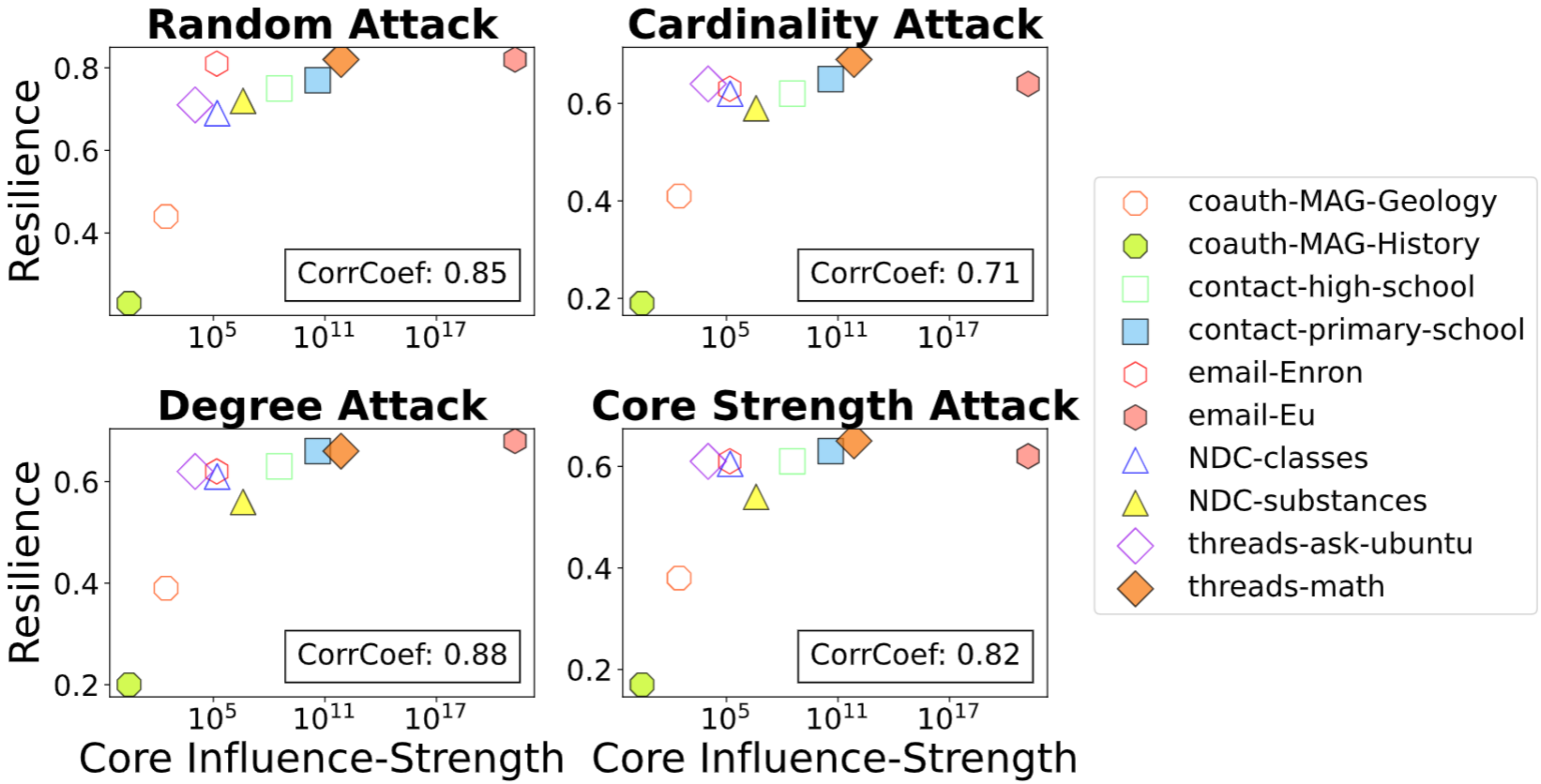
Observation 2: Core Strength Distributions

- Most nodes possess *low* core strengths (*robustness*), leaving room for improvement via supplementing hyperedges.



Observation 3: Core Influence-Strength

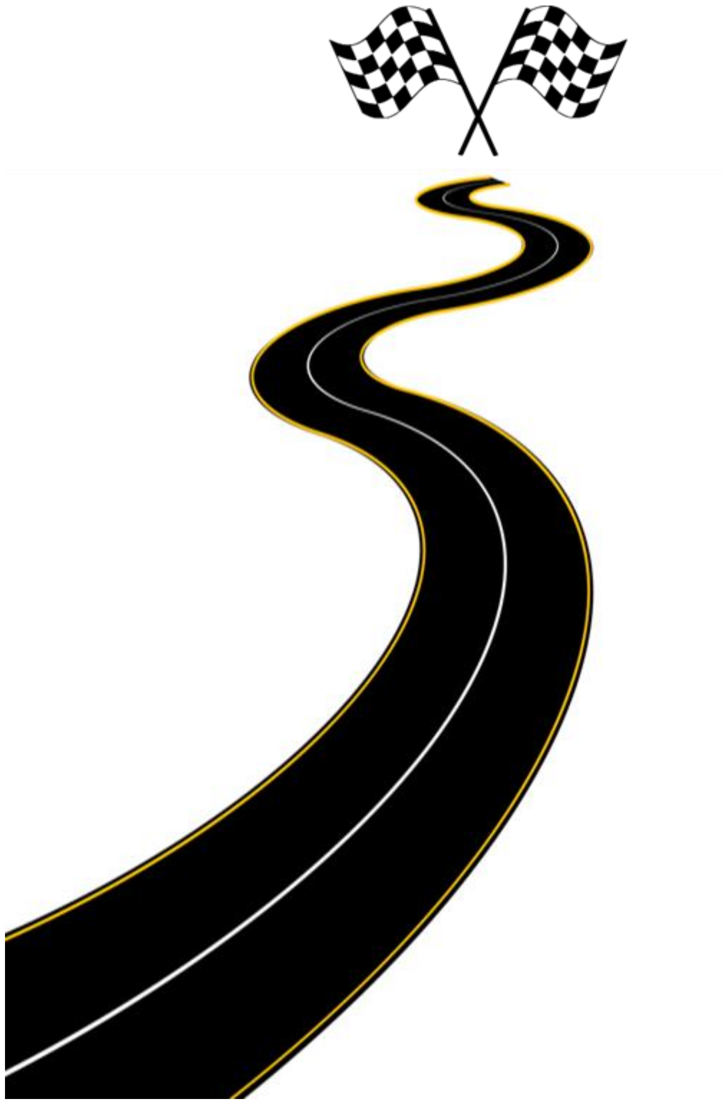
- The **core influence-strength*** of G reflects whether *important* (high core influence) nodes are *robust* (high core strength)
- This statistic is *positively correlated* with core resilience.



* Refer to the paper for the exact formula.

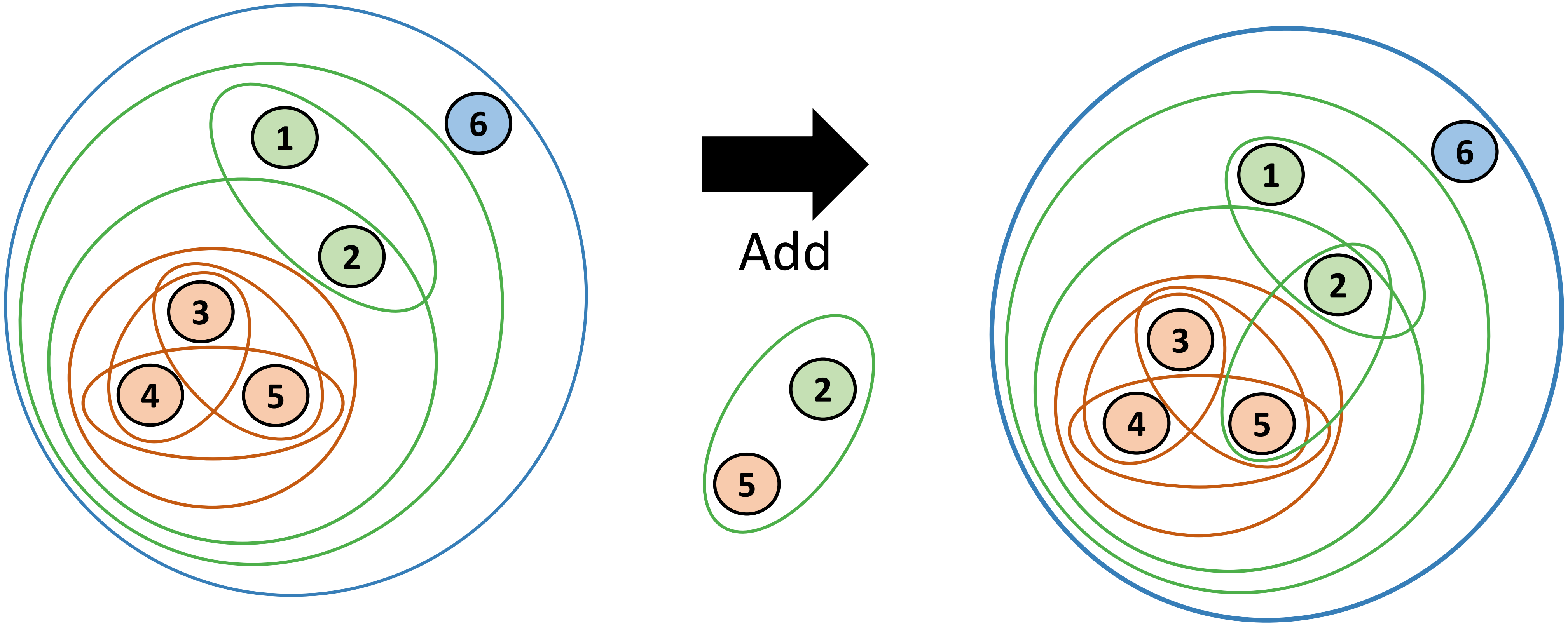
Roadmap

1. Preliminary
2. Observations
3. **Method <<**
4. Experiments
5. Conclusion



Naïve Approach

“Which groups of nodes should we augment to conserve all core numbers?”

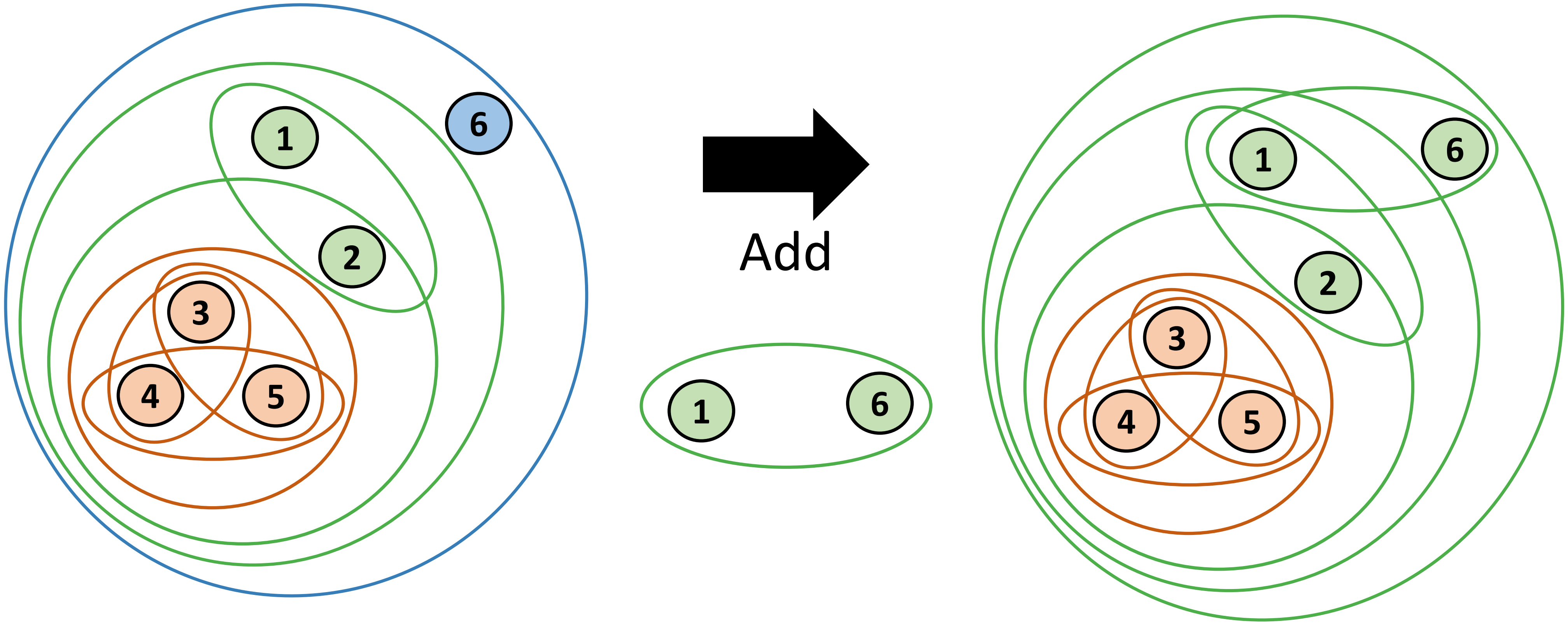


All core numbers conserved!

Adding this hyperedges preserves all core numbers.

Naïve Approach

“Which groups of nodes should we augment to conserve all core numbers?”

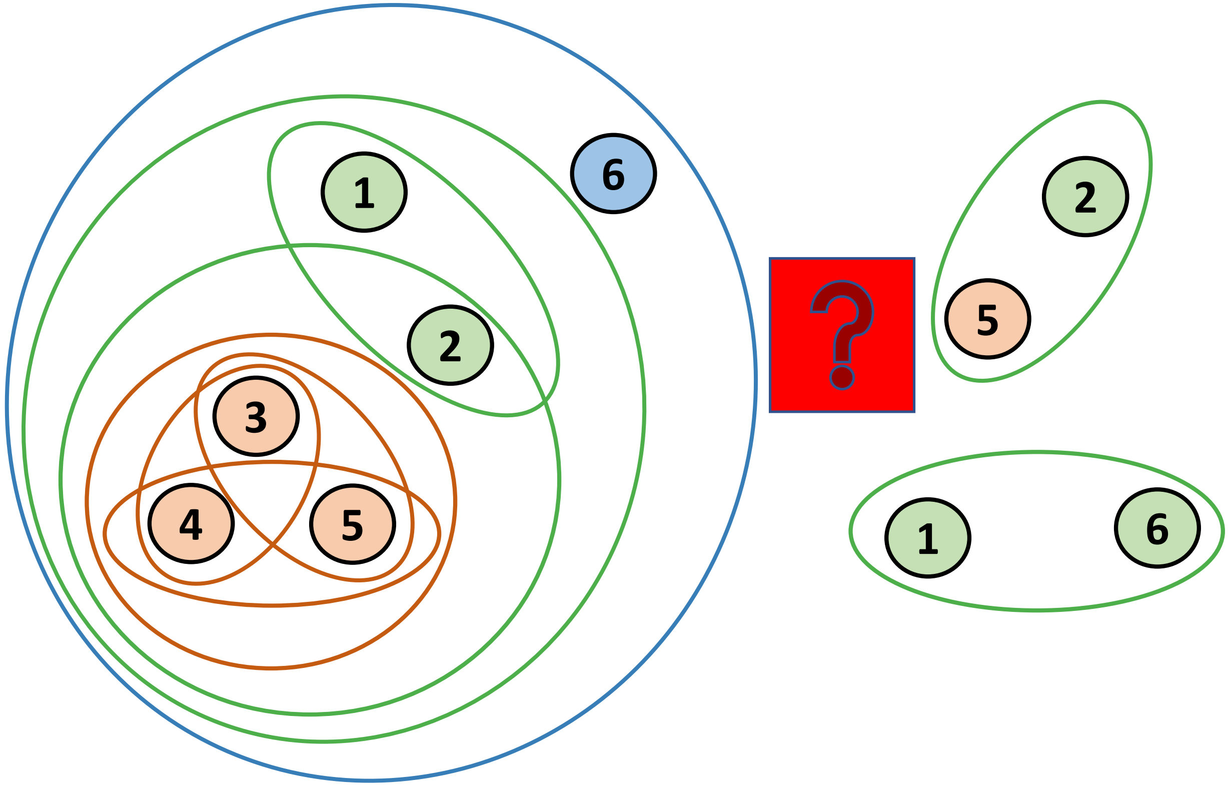


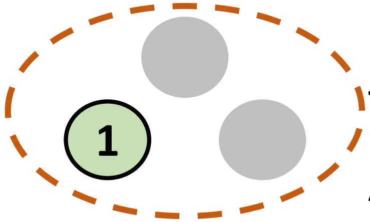
Core number of 6 changed!

Adding this hyperedges violates the constraint of preserving all core numbers.

Work-around: "Anchor availability"

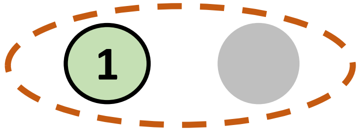
"For each node v, how many hyperedges contribute to the core number of v? And how hyperedges like that can we add?"



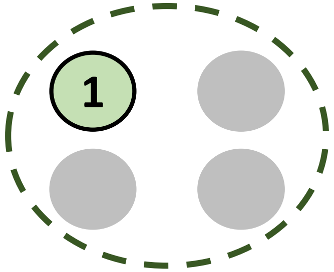


Q: Which hyperedges contribute to the core number of 1?

A: Those where 1 is the anchor.



Q: How many more hyperedges like that can we add?



A: The number depends on the core number & the number of such hyperedges.

Naïve Approach

Our Approach

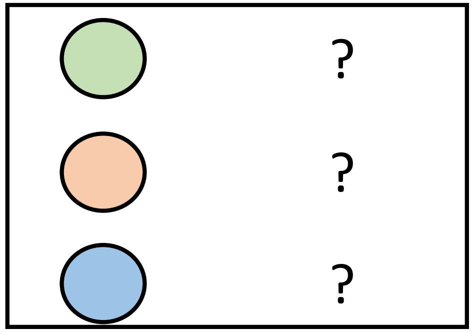
Method Design Questions

- Q1: How to determine the *maximum number* of hyperedges to add?
- Q2: How to construct *that number* of hyperedges?
- Q3: How to select the *best* hyperedges?

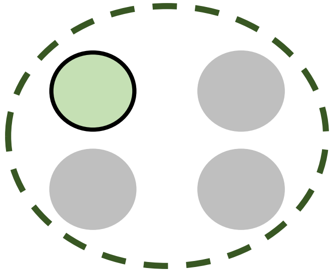
Proposed Method: **COREA**
(**CO**re **RE**silience Improvement by Hyperedge **A**ugmentation)

Algorithm Overview: COREA

- The steps to answer the method design questions:
  - **Step 1-1. Anchor Availabilities:** for each node, determine the number of hyperedges guaranteed to *preserve all core numbers* if augmented.
  - **Step 1-2. Candidate Construction:** form candidate hyperedges *based on the numbers* determined in Step 1-1.
  - **Step 2. Candidate Selection:** select the *best* candidate hyperedges within the given budget.



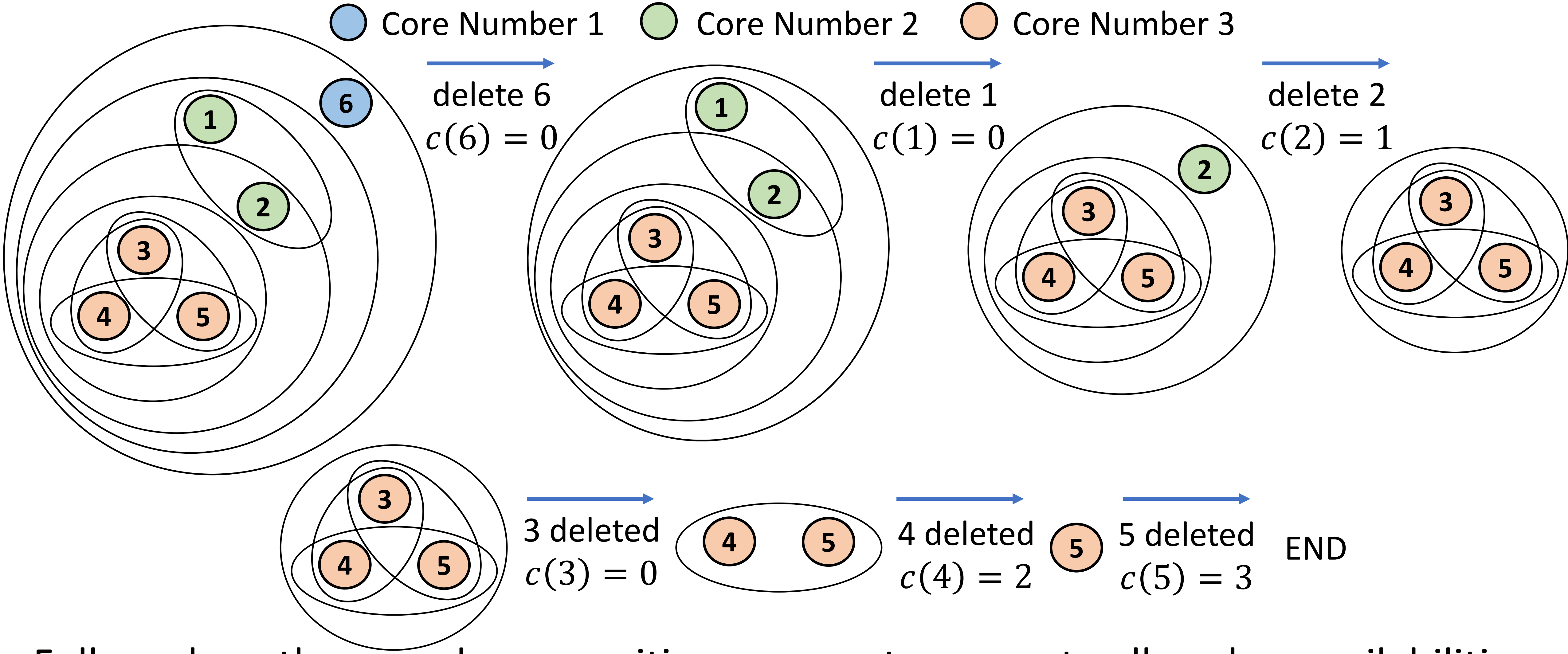
S1-1. Anchor Availability



S1-2. Construction

S2. Selection

Step 1-1. Anchor Availabilities



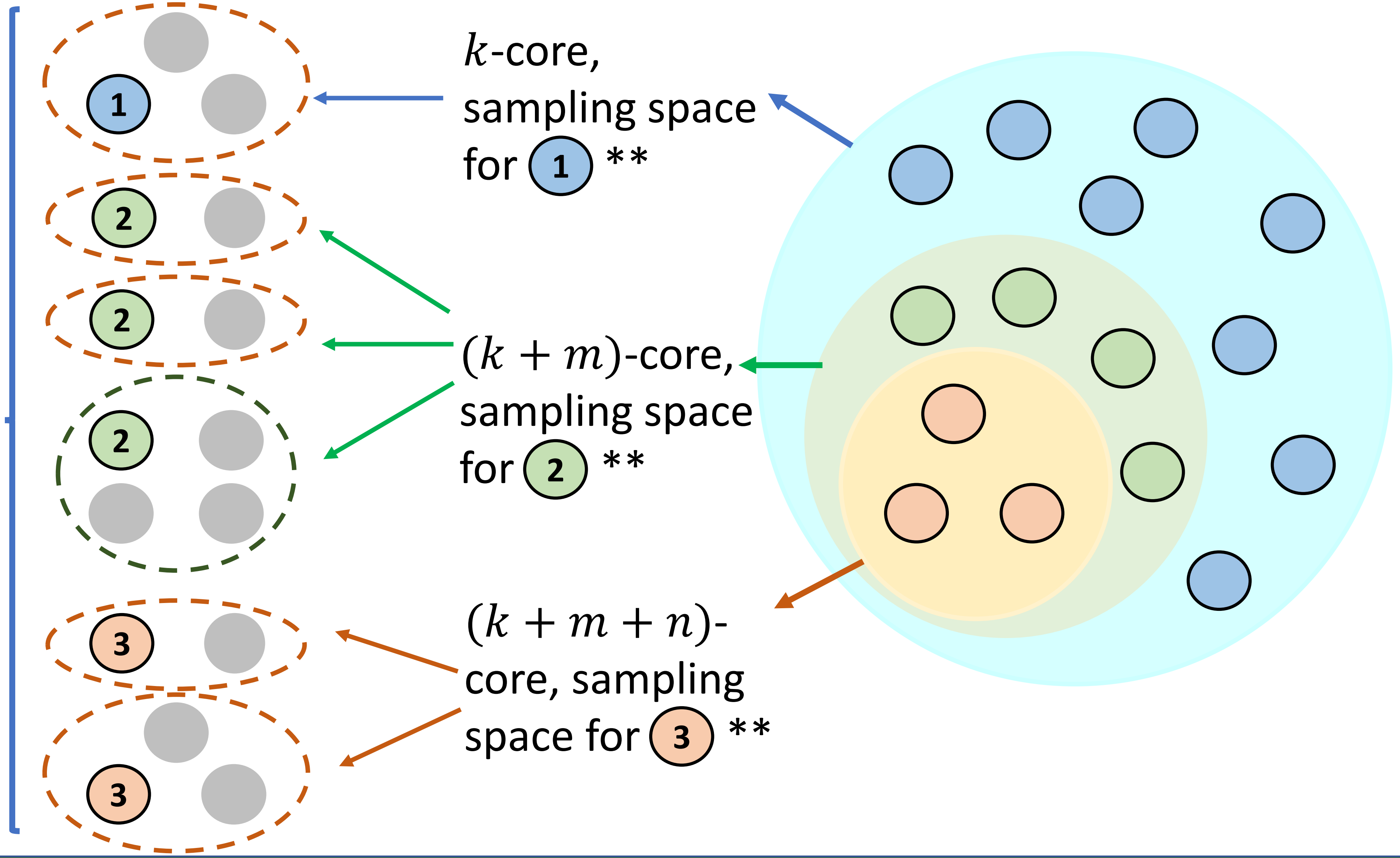
Follow along the core decomposition process to compute all anchor availabilities

Step 1-2. Candidate Construction

Anchor Availability	
1	1
2	3
3	2

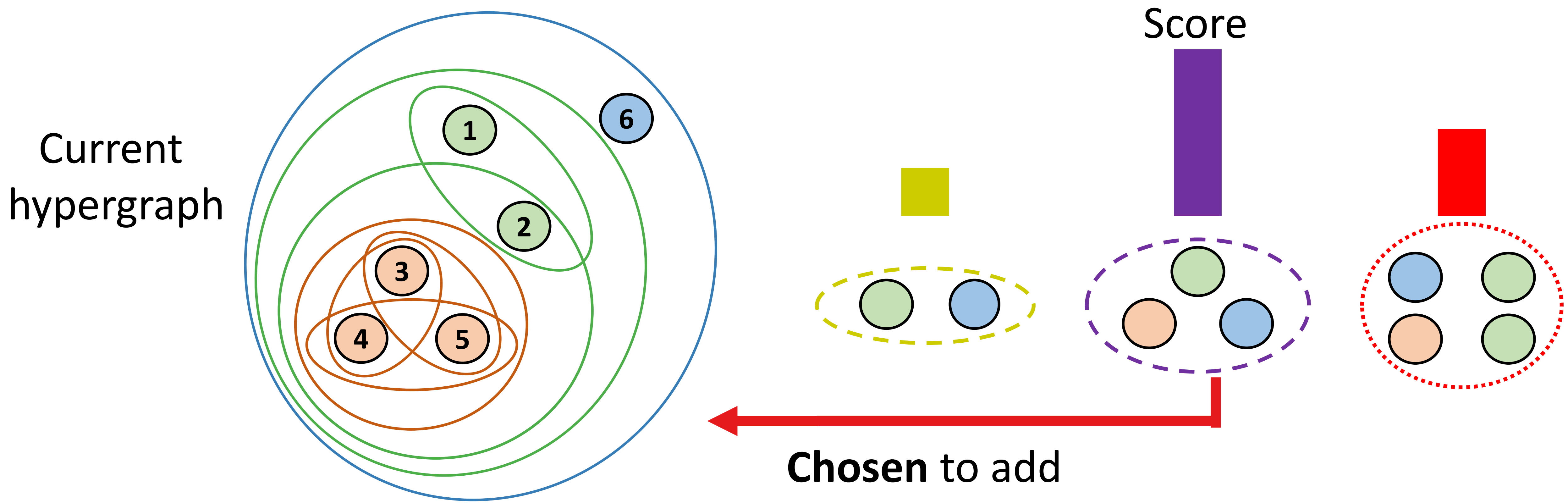
Size distribution follows the hyperedge size distribution D of G.

$**k, m, n \in \mathbb{N}^*$



Step 2. Candidate Selection

- Repeat until the budget is exhausted:
  - Choose c hyperedges with the *highest scores* to add to G.
  - Update the scores of all remaining candidate hyperedges.

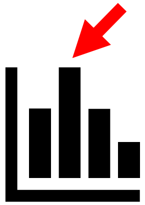


Theoretical Merits of COREA

- <u>**Feasibility**</u>: **COREA** constructs a set of candidate hyperedges that guarantee to <u>*preserve all core numbers*</u> if added to G.

- <u>**Invariance**</u>: **COREA** always returns the <u>*same number*</u> of candidate hyperedges despite the possibly different deletion orders of the core decomposition process.

- <u>**Exhaustiveness**</u>: **COREA** always returns the <u>*maximum number*</u> of candidate hyperedges, subject to the constraint of preserving all core numbers.



Roadmap

1. Preliminary
2. Observations
3. Method
4. **Experiments <<**
5. Conclusion

Experimental Settings

- **Datasets**: *10* real-worlds hypergraphs from 5 domains: *Co-Authorship*, *Contact*, *Email*, *NDC Drug*, and *Online Forums*.
- **Budget $\boldsymbol{B}$**: 5% of the number of hyperedges.
- **Deletions**: $5 - 25\%$ of the hyperedges by Core Strength Attack.
- Similar results regarding the relative performances of the considered methods for other attack schemes.
- **Performance**: improvement of core resilience due to the added hyperedges that are chosen by the method.

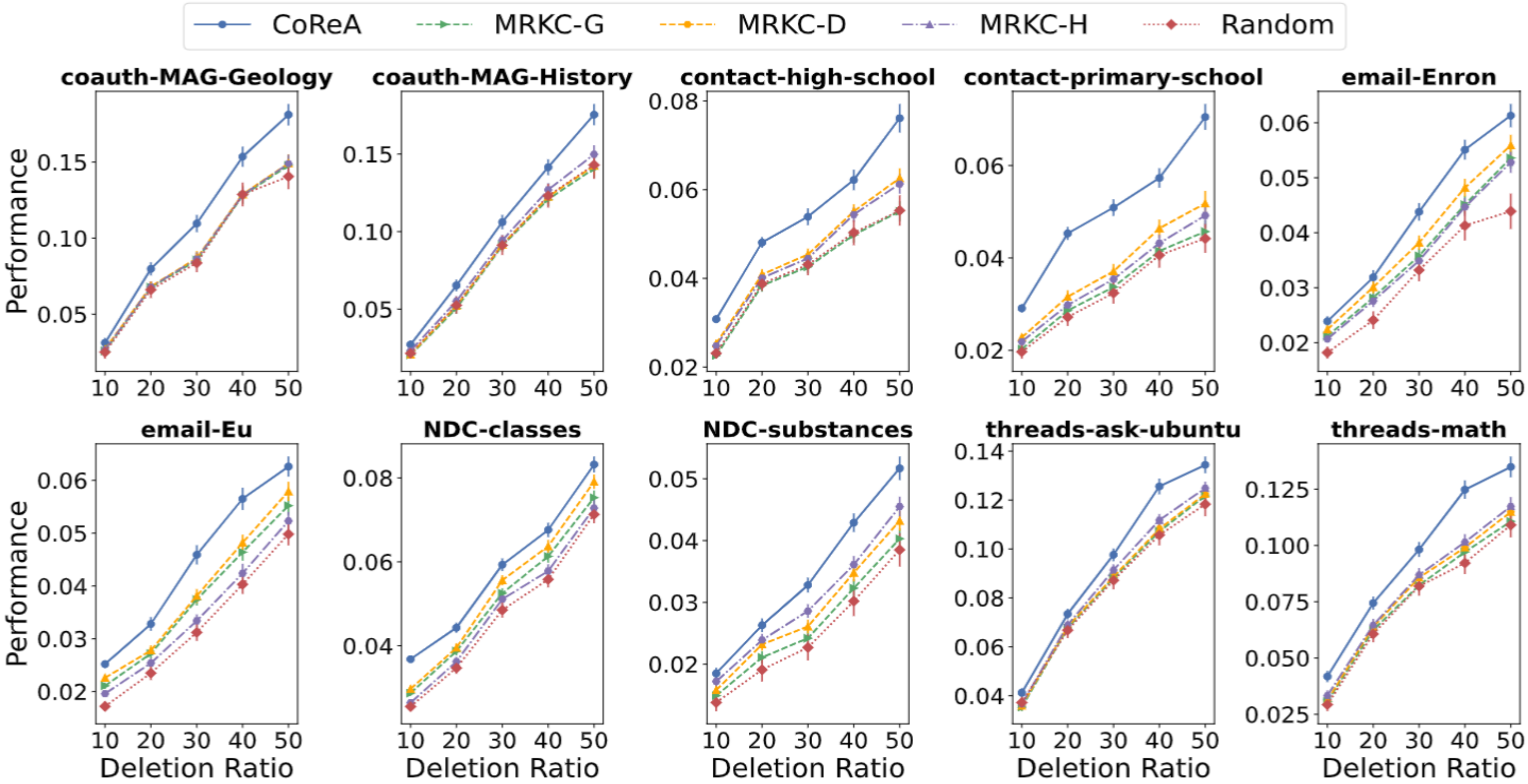
Experimental Settings (cont)

- **Baselines**: extensions of MRKC, the method for pair-wise graph [1], to hypergraphs
  - **MRKC-G**: apply MRKC to the clique expansion of the hypergraph
  - **MRKC-D**: apply MRKC to the multi-level decomposition of the hypergraph
  - **MRKC-H**: apply **COREA** to form size-2 hyperedges and the selection criterion in [1].
  - **Random**: random selection in each step of **COREA**.

[1] Ricky Laishram, et al., *"Measuring and Improving the Core Resilience of Networks",* The Web Conference 2018.

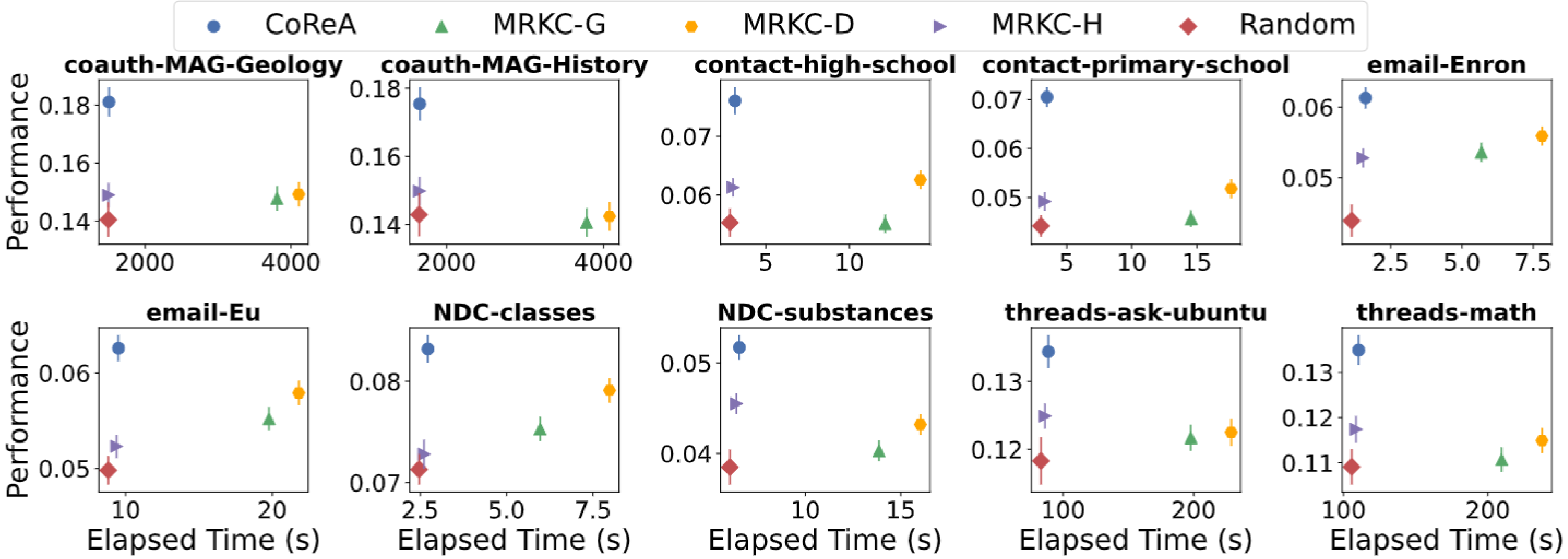
Main Performance

- **COREA** shows *consistent superiority* in improving core resilience



Time-Performance Tradeoff

- **COREA** provides *consistently better* time-performance tradeoff



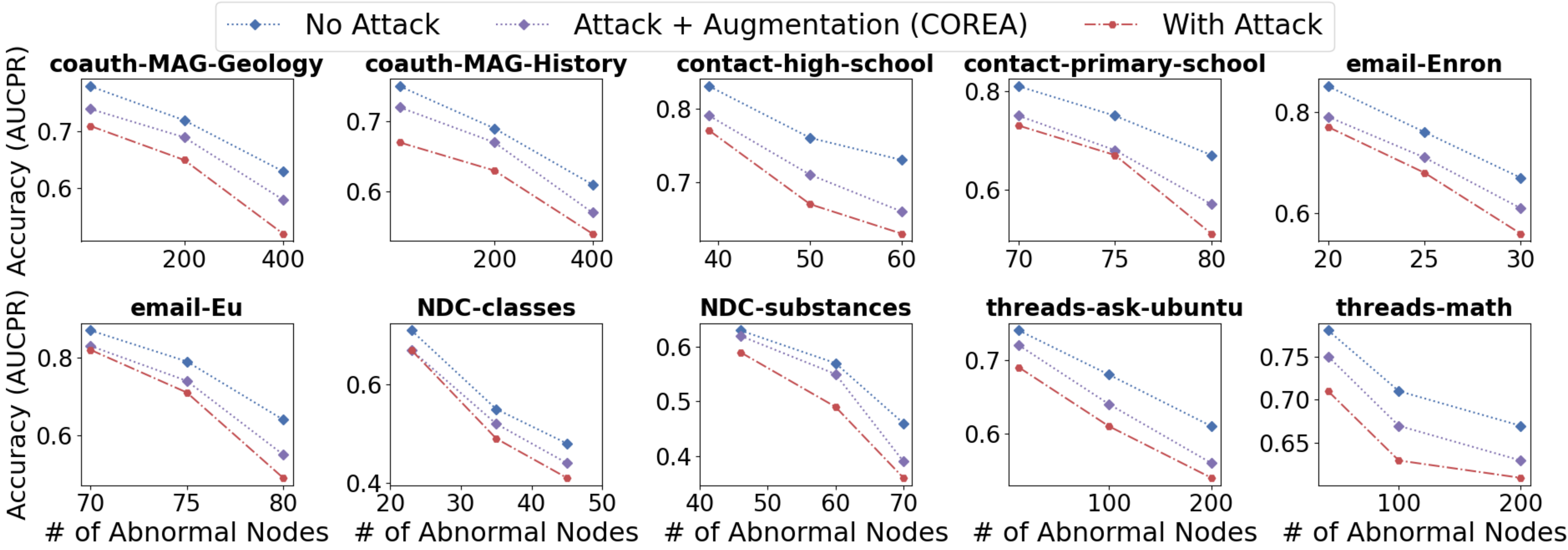
Application: Anomaly Detection

- **Settings:** designate some nodes as *abnormal nodes* and *inject large-size hyperedges* consisting of these nodes as *abnormally dense blocks*.
- **Objective:** measure the accuracy of a core-based scoring method in *detecting anomalies* [2].
- **Considered Performances:**
  - **No Attack:** Detection accuracy in the original network.
  - **With Attack:** Detection accuracy when the network is **under deletion attacks**.
  - **Attack + Augmentation (COREA):** Detection accuracy when the network, supplemented with the hyperedges by **COREA**, is **under deletion attacks**.

* Kijung Shin et al., *"Patterns and Anomalies in k-Cors of Real-World Graphs with Applications"*, KAIS 2018.

Application 1: Anomaly Detection

- The core numbers are less helpful due to deletion attacks
- **COREA** helps *alleviate* the drop in detection accuracy



Roadmap

1. Preliminary
2. Observations
3. Method
4. Results
5. **Conclusion <<**

Conclusions

- **Novel Problem:** of improving core resilience of hypergraphs by supplementing hyperedges.
- **Key Concepts & Observations:** relevant to the core resilience of real-world hypergraphs and motivate the method design.
- **Proposed method (COREA):** *fast*, *effective*, and *theoretically sound* method for improving the core resilience of the hypergrpahs.

Code & Dataset: https://github.com/manhtuando97/CoReA

Image Credit: Wikipedia, flaticon