Santiago Chile

SIGMOD PODS 2024

Play like a Vertex: A Stackelberg Game Approach for Streaming Graph Partitioning

ZEZHONG DING, YONGAN XIANG, SHANGYOU WANG,
XIKE XIE*, and S. KEVIN ZHOU

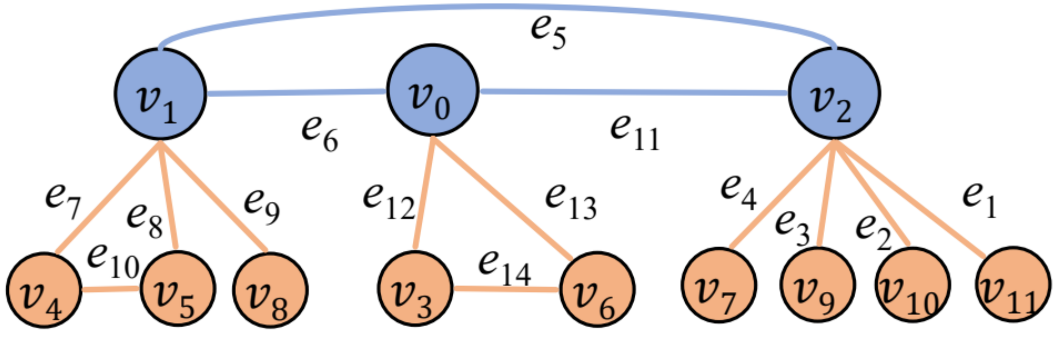
June. 13, 2024

中国科学技术大学
University of Science and Technology of China

Data Darkness Laboratory

Graph Partitioning

- **Graph partitioning is a key technology in distributed graph processing**
  - ① **partition input large graph data into sub-graphs (Graph Partitioning)**
  - ② assign each sub-graph to each computer
  - ③ make graph analysis over the distributed graph



Stream Load

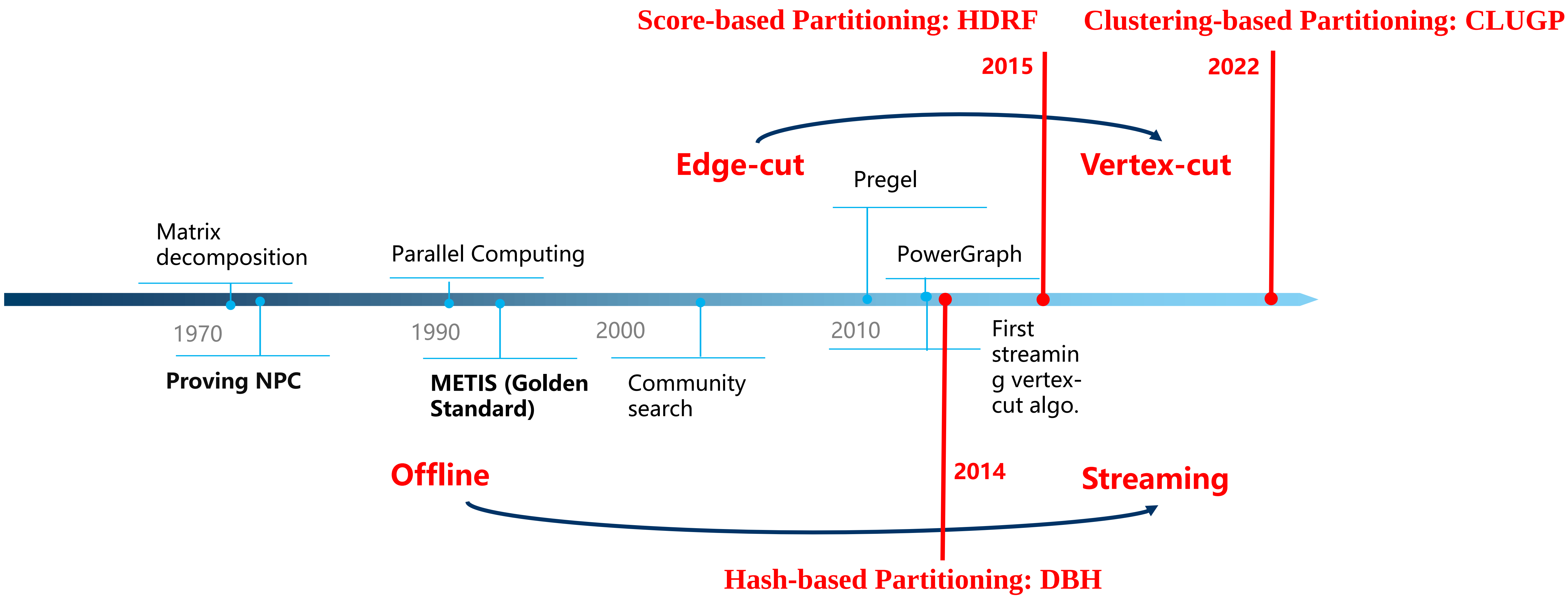
Distributed Nodes

- **Objective**: Replication Factor (under the same load balance)

Replication Factor(RF) → Reduce # of mirror vertices. → Communication Latency

Load Balance ↔ Keep the size of per partition similar

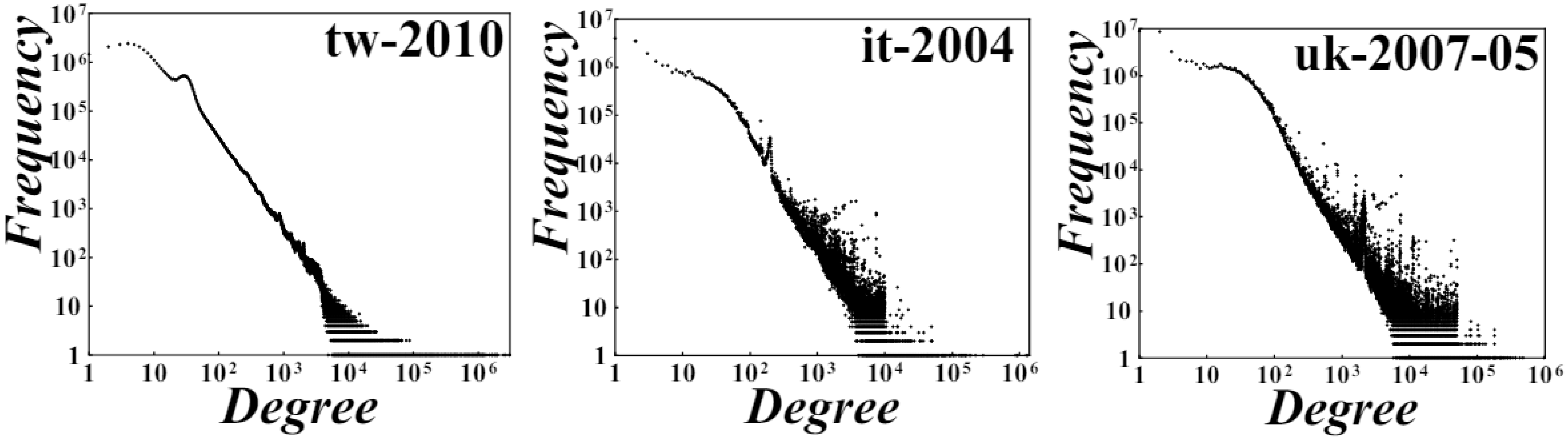
Graph Partitioning: History

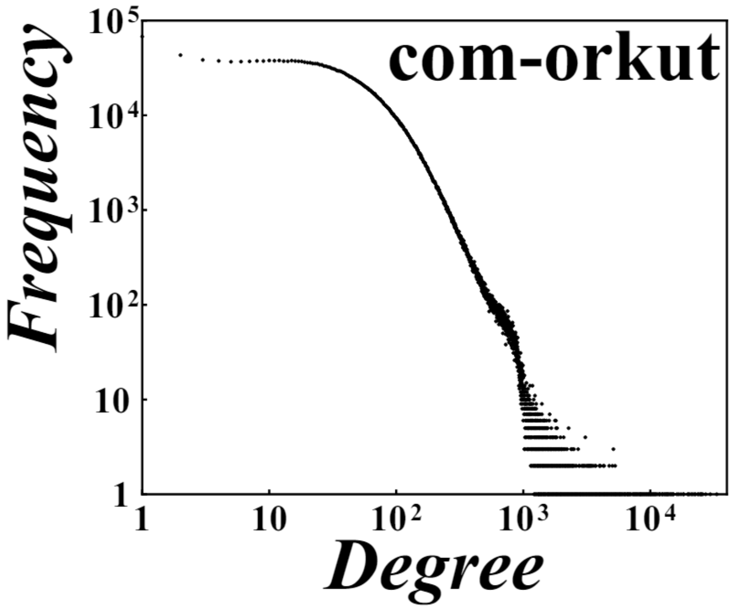


What would be the potentials in optimizing stream graph partitioning?

Graph Skewness

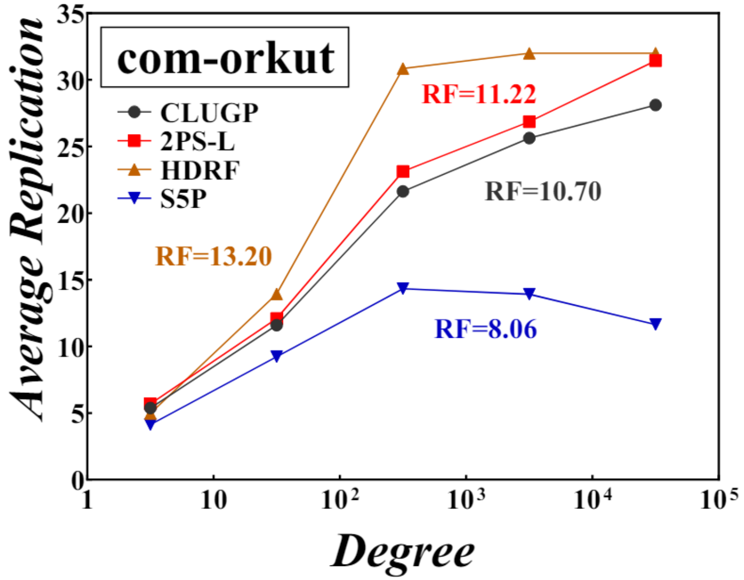
Graph Skewness





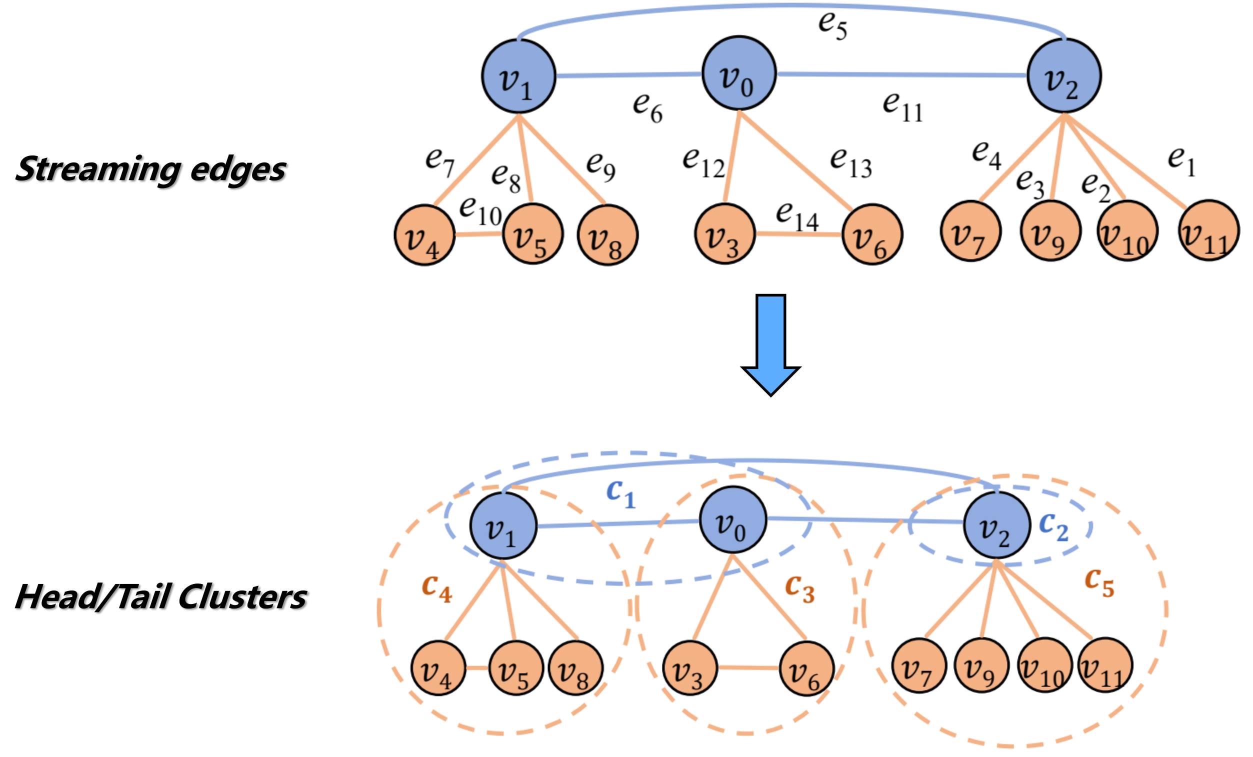
"80% of the edges are connected to only 20 % of the nodes."

- **Existing graph partitioning algorithms do not adequately consider the skewness of the graph.**

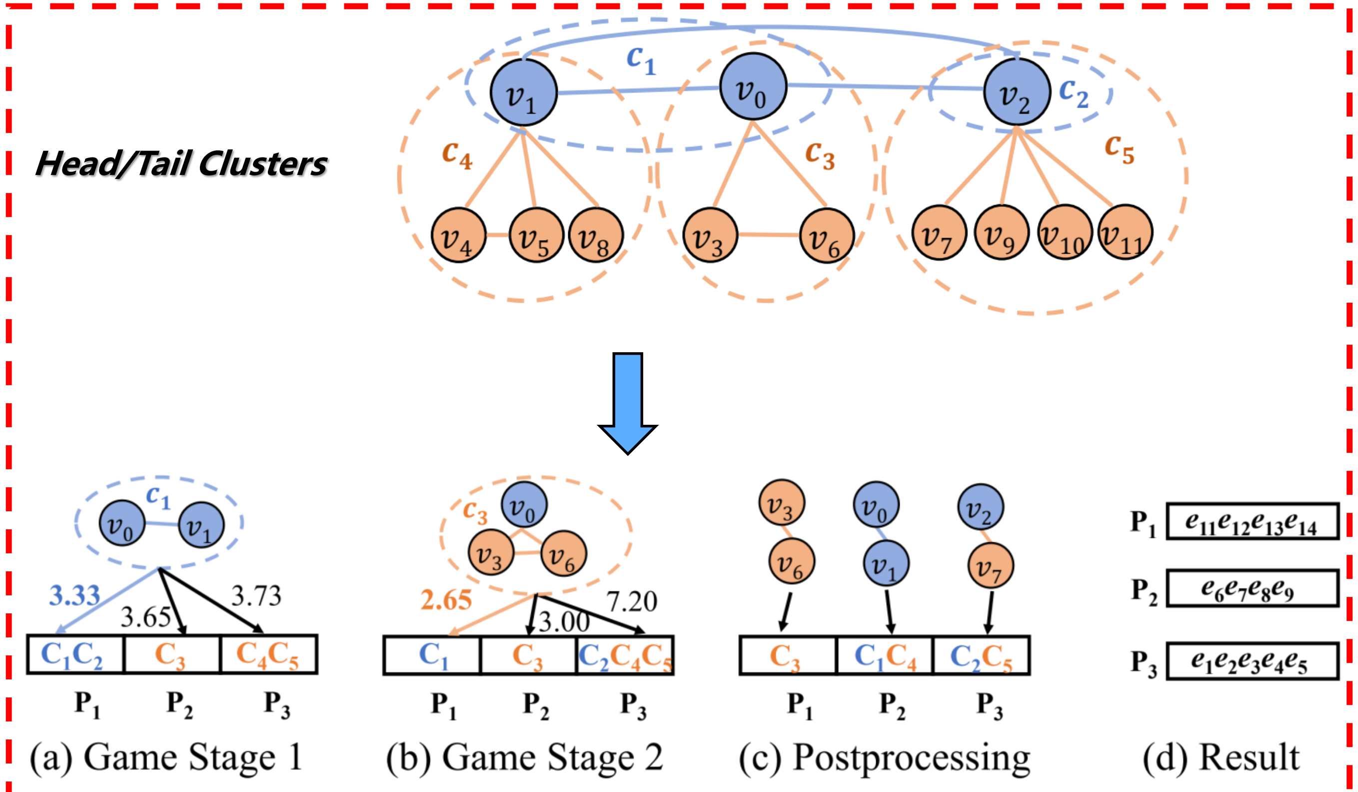


Skewness-aware Vertex-cut Partitioning(S5P) Framework

Clustering Phase

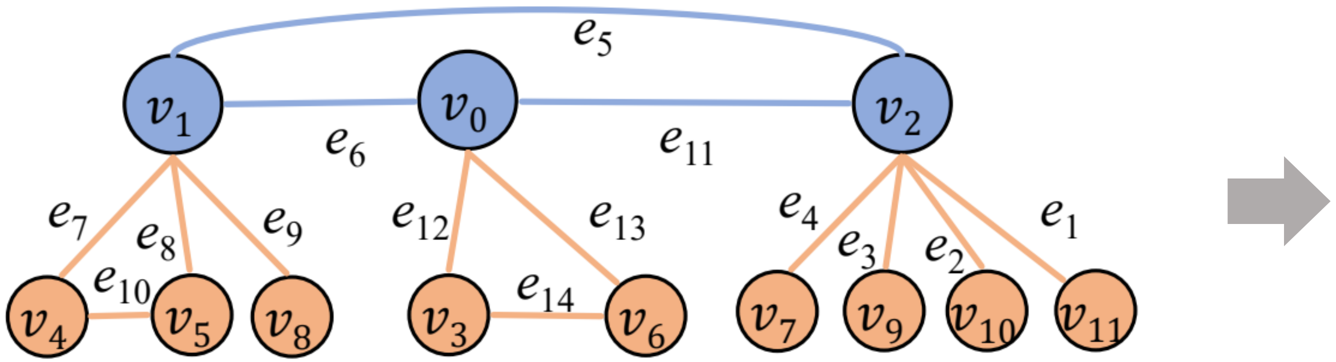


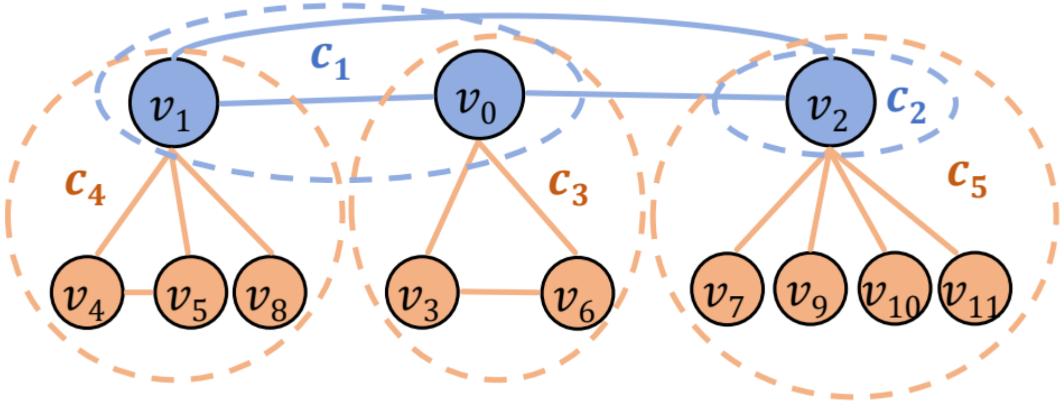
Refinement (Game) Phase



Skewness-aware Graph Clustering

- **Head and Tail vertices/edges are separated by a parameter β:**





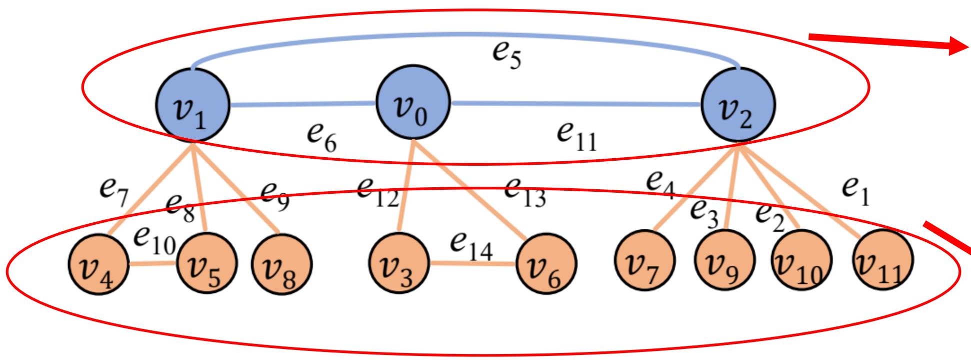
① **one-pass manner to get head and tail clusters**

② **global degree-aware operation for head edges and get head clusters**

③ **local degree-aware operation for tail edges**

- **Advantages**
  - ✓ **Skewness-aware**
  - ✓ **$O(|E|)$ to $O(|C|)$, $|C|$ is the number of clusters, $|C| \ll |E|$**

Why consider game theory?



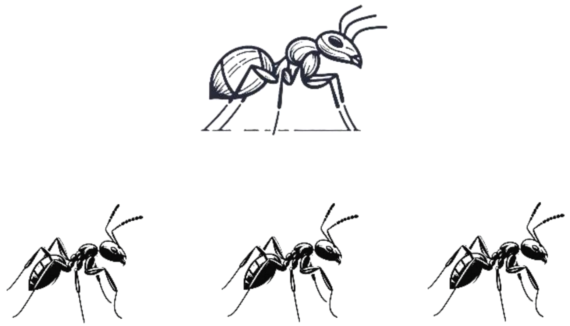
Graph

High-degree(Head) vertices

Low-degree(Tail) vertices

Queen ant

Worker ant



Ant Colony

- **Quality:**

  The optimization objectives of the game theory and the partitioning are consistent.

- **Efficiency:**

  The game theory problems can be solved using parallel computing techniques.

Stackelberg Graph Game: Overview

- **Notations**
  - Ω and Φ are cost functions
  - θ and λ are strategies of leader and Follower

- **Stackelberg Game Model**

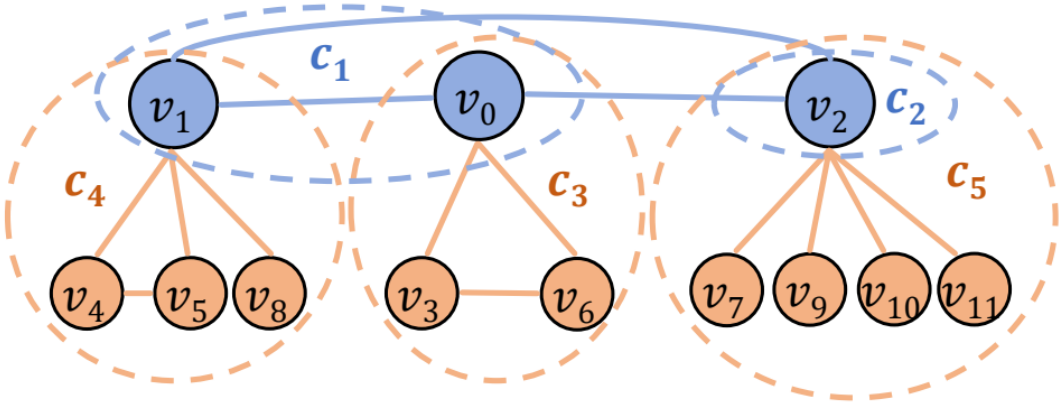
Stage 1 [Leaders′ Side]: $\theta^* = argmin_{\theta}\Omega(\theta, \lambda)$

Stage 2 [Followers′ Side]: $\lambda^* = argmin_{\lambda}\Phi(\theta, \lambda)$

- **Stackelberg Equlibrium**

$$\Omega(\theta^*, \lambda^*) \leq \Omega(\theta, \lambda^*)$$

$$\Phi(\theta^*, \lambda^*) \leq \Phi(\theta^*, \lambda)$$



Leaders: head clusters (in blue)
Followers: Tail clusters (in orange)

- ***Two Questions***
  - ☐ *how to enhance the quality?*
  - ☐ *how to optimize the efficiency and memory overhead?*

Stackelberg Graph Game: Quality

- **Total Cost Function(based on size and intersection)**

$$\underbrace{\delta \frac{\sum_{i=1}^{k} |p_i|^2}{k}}_{load\ balance\ part} + \underbrace{\frac{\sum_{i=1}^{k} \Theta(p_i, V)}{k}}_{communication\ part}$$

k is the number of partitions

- **Individual Cost of Clusters (based on size and intersection)**

$$S_{c_i}(p_i) = \frac{\delta}{k}|c_i| \cdot |p_i| + \frac{F(c_i) + |c_i|}{k}$$

$$F(c_i) = \sum_{c_j \in C_H \cup C_L} \Theta(c_i, c_j) \mathrm{I}(i, j)$$

The stackelberg game cost is the sum of all individual cost of clusters. (See Theorem 4)

- **OPT(Game) vs. Nash equilibrium: Price of Anarchy**

(Measure the maximum gap between the Nash equilibrium solution and the optimal solution)

$$k + 1$$

- **OPT(Partitioning) vs. S5P: RF**

Graph skewness

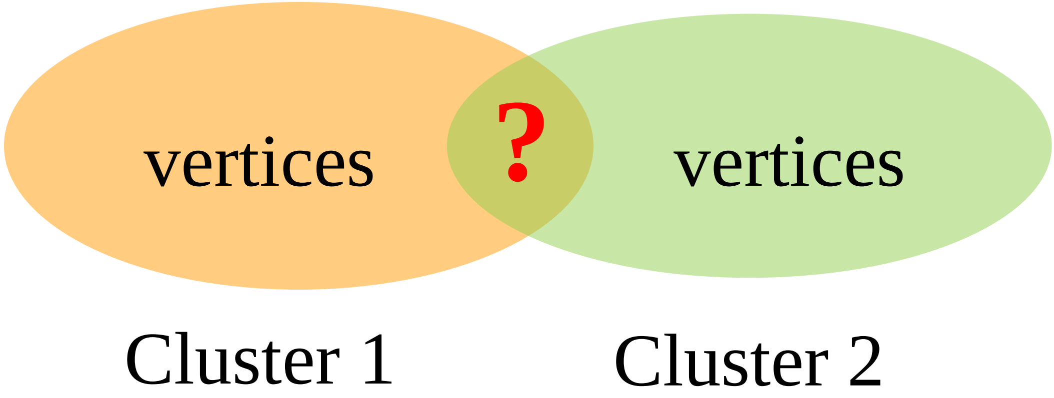
$$RF \leq \underbrace{\chi_H \cdot k}_{head\ part} + \underbrace{\sum_{i=1}^{\chi_T |V|} \frac{d_m \left(\left(\frac{k-1}{d_m} \right)^{1-\rho} + \frac{i-1}{|V|} \right)^{-1}}{\chi_T \cdot |V|} + 1}_{tail\ part} = f(\rho)$$

The graph is more skewed, the RF bound of S5P is tighter.

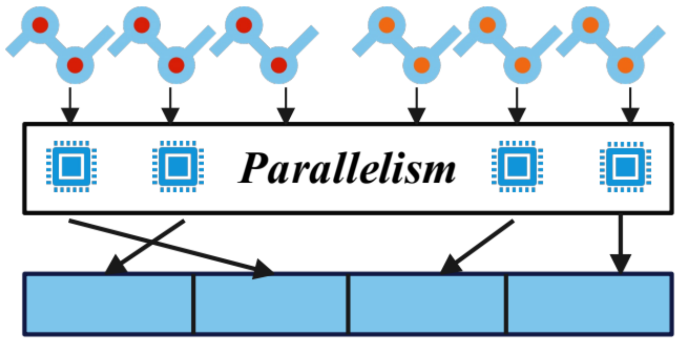
See our paper for more details.

Stackelberg Graph Game: Efficiency and Memory Overhead

- **Sketching:** Using the CM-Sketch probabilistic data structure to achieve approximate estimation of local optimization objectives, with theoretical guarantees provided by the sketch.



- **Parallelization:** Implementing parallel acceleration of the game process using cluster sharding and multithreading.



Evaluations

- Regression-based Graph Skewness
- Pearson's First/Second Graph Skewness
- Planarization Graph Skewness

➢ Baseline partitioners

- ☐ Offline partitioners: NE, METIS, HEP
- ☐ Streaming partitoners: HDRF, Greedy, DBH, 2PS-L, CLUGP
- ☐ Other game-based partitioners: RMGP, MDSGP, CVSP

➢ Partitioning metrics

- ☐ Replication factor (The load balance is set as 1.0)
- ☐ Run-time
- ☐ Memory overhead

➢ Real-world Graphs

- ☐ 4 Social Graphs: e.g., FR(|V|=66M, |E|=1.8B, SIZE=31GiB)
- ☐ 7 Web Graphs: e.g., UK7(|V|=106M, |E|=3.7B, SIZE=63GiB)

➢ Synthetic Graphs Generated by R-MAT

Graphs	|V|	|E|	Skewness
G_1	1.04M	314M	(0.89,0.15,0.44,102M)
G_2	1.04M	629M	(0.87,0.17,0.48,626M)
G_3	1.04M	1.04B	(0.84,0.19,0.52,1B)
G_4	67.1M	671M	(1.16,0.048,0.145,469M)
G_5	67.1M	2.01B	(1.11,0.051,0.152,1B)
G_6	67.1M	3.36B	(1.07,0.053,0.157,3B)

Skewness: $G_1<G_2<G_3$; $G_4<G_5<G_6$

Performance(For more information, please refer to our paper)

Replication Factor of Different Graphs (lower is better)

30% 91%

Graph \ Partitioner	CLUGP			2PS-L			HDRF			S5P		
	k:64	*k*:128	*k*:256	*k*:64	*k*:128	*k*:256	*k*:64	*k*:128	*k*:256	*k*:64	*k*:128	*k*:256
OK	14.288	17.522	20.636	15.112	18.915	23.200	17.860	22.617	27.023	**11.614**	**15.391**	**19.055**
TW	8.808	10.817	11.861	10.642	13.074	15.577	9.520	11.789	14.408	**7.583**	**9.068**	**10.526**
FR	10.311	13.432	17.011	11.241	14.359	17.457	11.324	14.757	18.122	**7.870**	**11.244**	**14.995**
LJ	4.913	5.471	5.945	5.036	5.593	6.045	6.778	7.763	8.545	**4.549**	**5.112**	**5.636**
IT	1.908	1.973	2.041	3.680	4.110	4.420	12.538	14.500	16.469	**1.273**	**1.232**	**1.210**
UK7	1.754	1.876	1.839	3.338	3.760	4.077	14.190	16.700	19.181	**1.265**	**1.213**	**1.196**
IN	1.415	1.542	1.621	1.895	2.241	2.887	6.884	8.028	8.890	**1.229**	**1.207**	**1.225**
SK	2.299	2.584	2.566	4.001	5.466	7.029	16.561	19.413	21.766	**1.337**	**1.310**	**1.293**
UK2	1.561	1.698	1.692	2.644	2.752	2.921	9.414	10.673	11.791	**1.371**	**1.227**	**1.238**
AR	2.015	1.929	2.005	3.409	3.803	4.119	12.599	14.768	16.762	**1.131**	**1.213**	**1.233**
WB	1.446	1.493	1.485	1.829	1.836	1.822	5.951	6.646	7.283	**1.296**	**1.178**	**1.188**

➢ **Better replication factor than all streaming vertex partitioners**

G. \ Par.	RMGP			MDSGP			CVSP			CLUGP			S5P		
	RF	Time	Mem.	RF	Time	Mem.	RF	Time	Mem.	RF	Time	Mem.	RF	Time	Mem.
OK	16.7	535	4.01	9.9	324	8.95	17.4	141	2.25	10.7	91	1.02	**8.5**	**60**	**0.38**
TW	-	>24h	48.70	6.8	5189	99.08	-	>24h	56.01	7.6	1333	11.65	**6.0**	**808**	**4.64**
FR	10.9	4553	70.20	7.6	4934	144.96	11.2	2078	80.69	7.2	3045	14.12	**7.0**	**1466**	**7.22**
LJ	5.4	65	2.08	4.5	184	3.83	5.7	32	2.25	4.2	111	1.11	**3.9**	**28**	**0.48**
WB	4.2	1871	61.10	6.2	6320	119.45	4.8	822	79.46	1.5	1101	25.11	**1.1**	**696**	**12.90**
G_6	-	>24h	115.5	4.9	11915	231.87	-	>24h	110.8	4.8	4847	18.01	**4.4**	**2620**	**8.06**

Game-based Methods

➢ **Better RF, efficiency, and memory overhead than all streaming game-based vertex partitioners**

Component Analysis

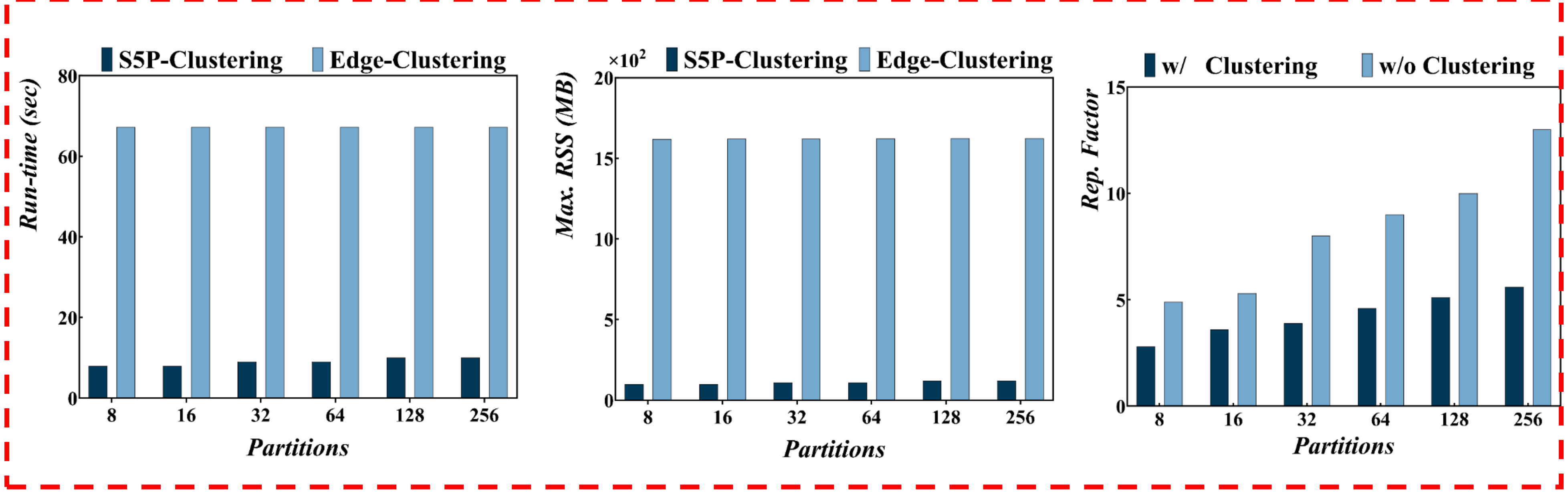
➢ **Skewness-aware Clustering**

✓ **8×** speedup and **6%** memory cost compared with Edge-Clustering method

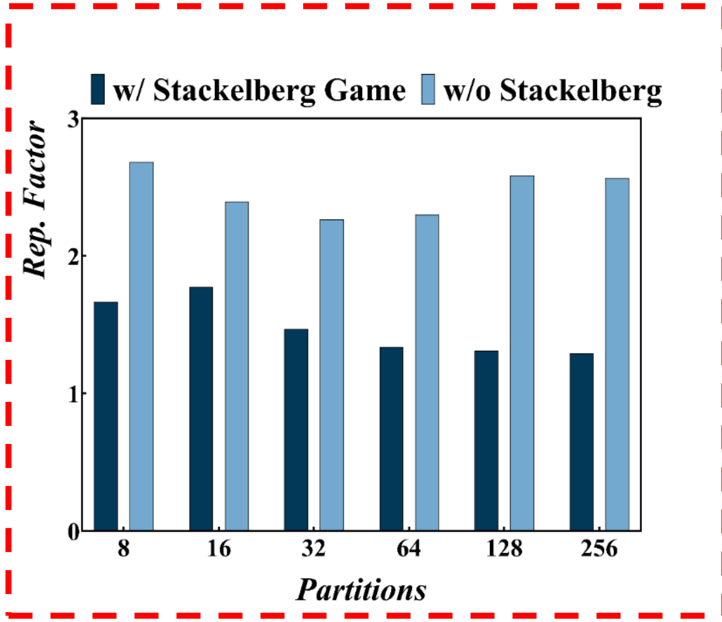
✓ **replication factor** reduction with clustering

➢ **Stackelberg Game**

✓ **replication factor** reduction with Stacklberg Game



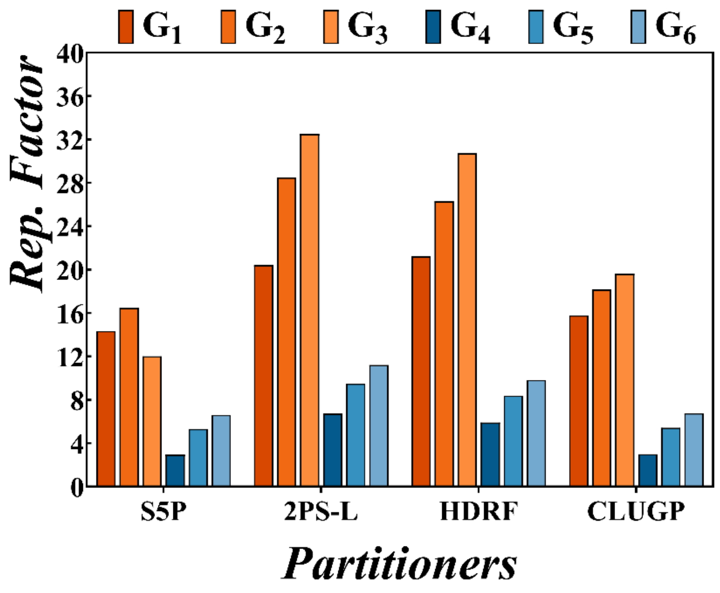
Clustering

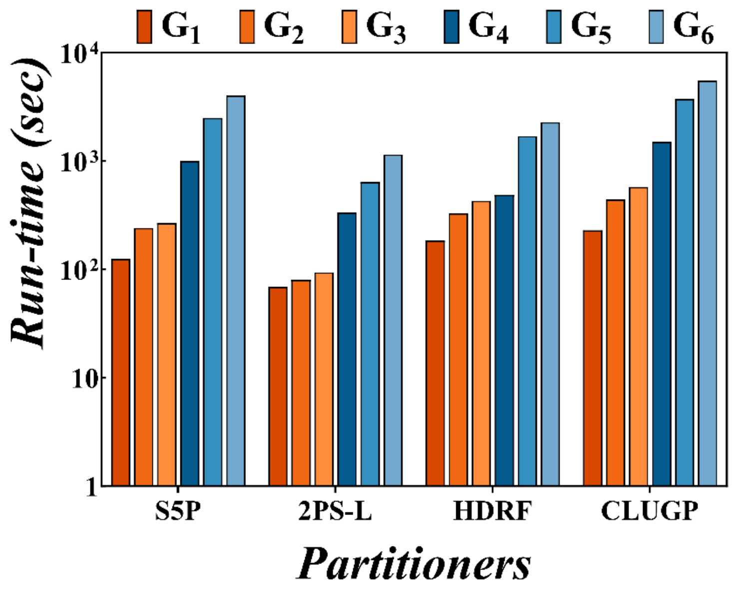


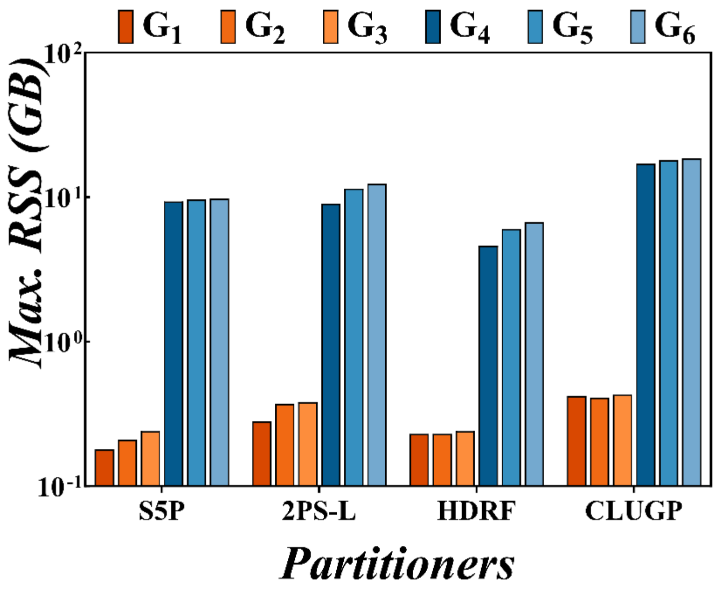
Game

Skewness Analysis

- Other partitioners exhibit **a substantial increase** in RF as the graphs are more skewed, while S5P has **smallest** RF increments
- Skewness: **(0.87, 0.17, 0.48, 626M)** to **(0.84, 0.19, 0.52,1B)** ~ RF: **16.460** to **12.011**







Conclusions

- **Graph skewness is an important but intractable property**

- **S5P achieves high partitioning quality by considering graph skewness**
  - ✓ The **Stackelberg Graph Game** can utilize **graph skewness information** to improve partition quality.
  - ✓ **The key to improving the quality of streaming graph partitioning lies in how effectively you can leverage the information about graph skewness.**

- **Future work**
  - ✓ Extend the skewness-aware partitioning paradigm to traditional graph computing systems and graph learning systems

中国科学技术大学 University of Science and Technology of China

Data Darkness Laboratory

Thank you!

S5P Source Code

Personal Website