

# p-Norm Flow Diffusion for Local Graph Clustering

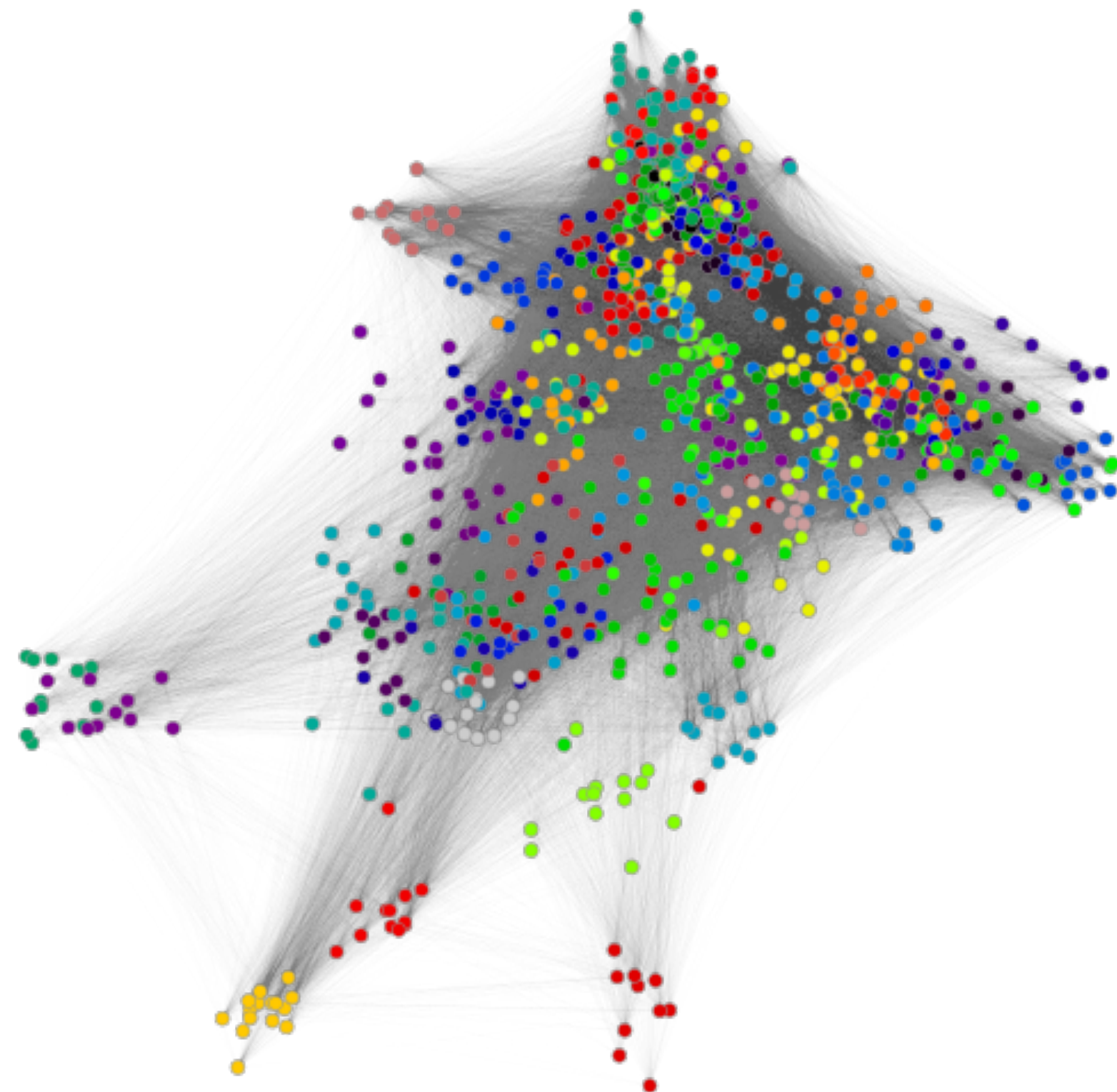
Kimon Fountoulakis<sup>1</sup>, Di Wang<sup>2</sup>, Shenghao Yang<sup>1</sup>

<sup>1</sup>University of Waterloo    <sup>2</sup>Google Research

ICML 2020

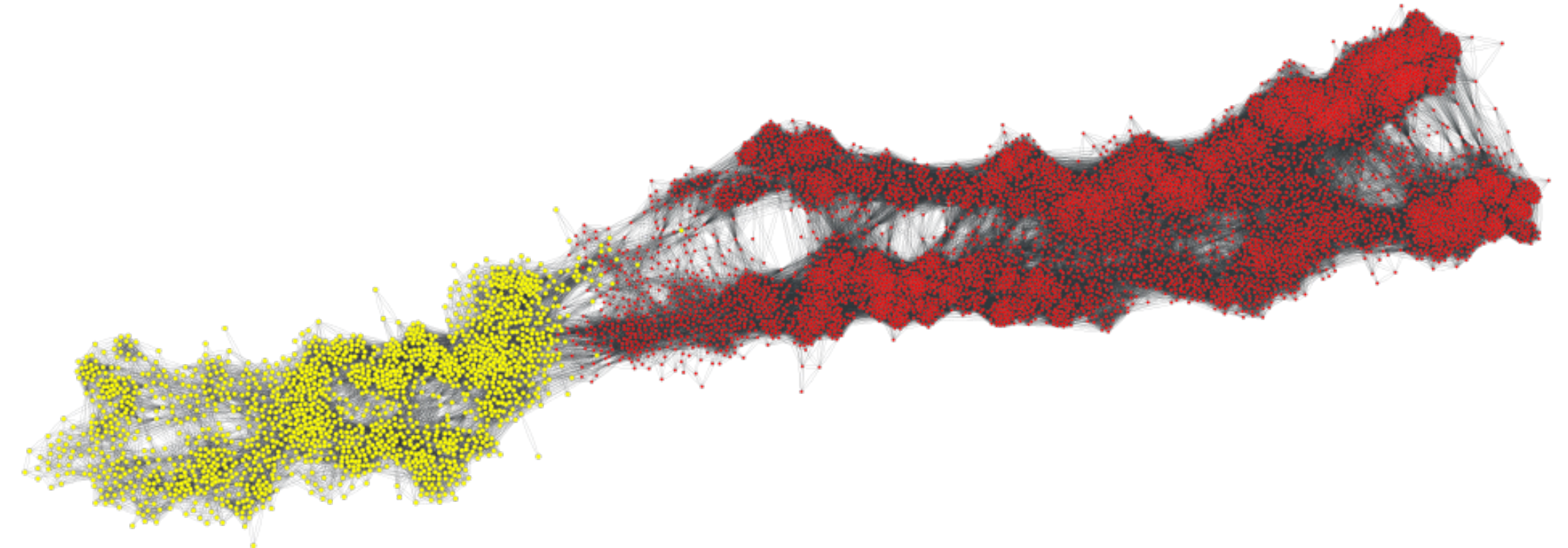
# Motivation: detection of small clusters in large and noisy graphs

- Real large-scale graphs have rich local structure
- We often have to detect small clusters in large graphs:



protein-protein interaction graph,  
color denotes similar functionality

Rather than partitioning graphs with  
nice structure



US-Senate graph,  
nice bi-partition in year 1865 around the end of  
the American civil war





Our goals: **simple** **local** algorithm with **good** theoretical guarantees

*Detection of small clusters in large graphs call for new methods that*

- run in time proportional to the size of the output (but not the whole graph),
- supported by good theoretical guarantees,
- require few tuning parameters.

Our goals: **simple** **local** algorithm with **good** theoretical guarantees

(Approximate Personalized) PageRank?

- run in time proportional to the size of the output (but not the whole graph),  
- supported by good theoretical guarantees, 
- require few tuning parameters. 

Our goals: **simple** **local** algorithm with **good** theoretical guarantees

Graph cut or max-flow approach?

- run in time proportional to the size of the output (but not the whole graph), ✓
- supported by good theoretical guarantees, ✓
- require few tuning parameters. ✗✗



Our goals: **simple** **local** algorithm with **good** theoretical guarantees

### This work

Let's replace PageRank with an even simpler model

- run in time proportional to the size of the output (but not the whole graph), ✓✓
- supported by good theoretical guarantees, ✓
- require few tuning parameters. ✓

# Existing local graph clustering methods

## Spectral diffusions

## Combinatorial diffusions



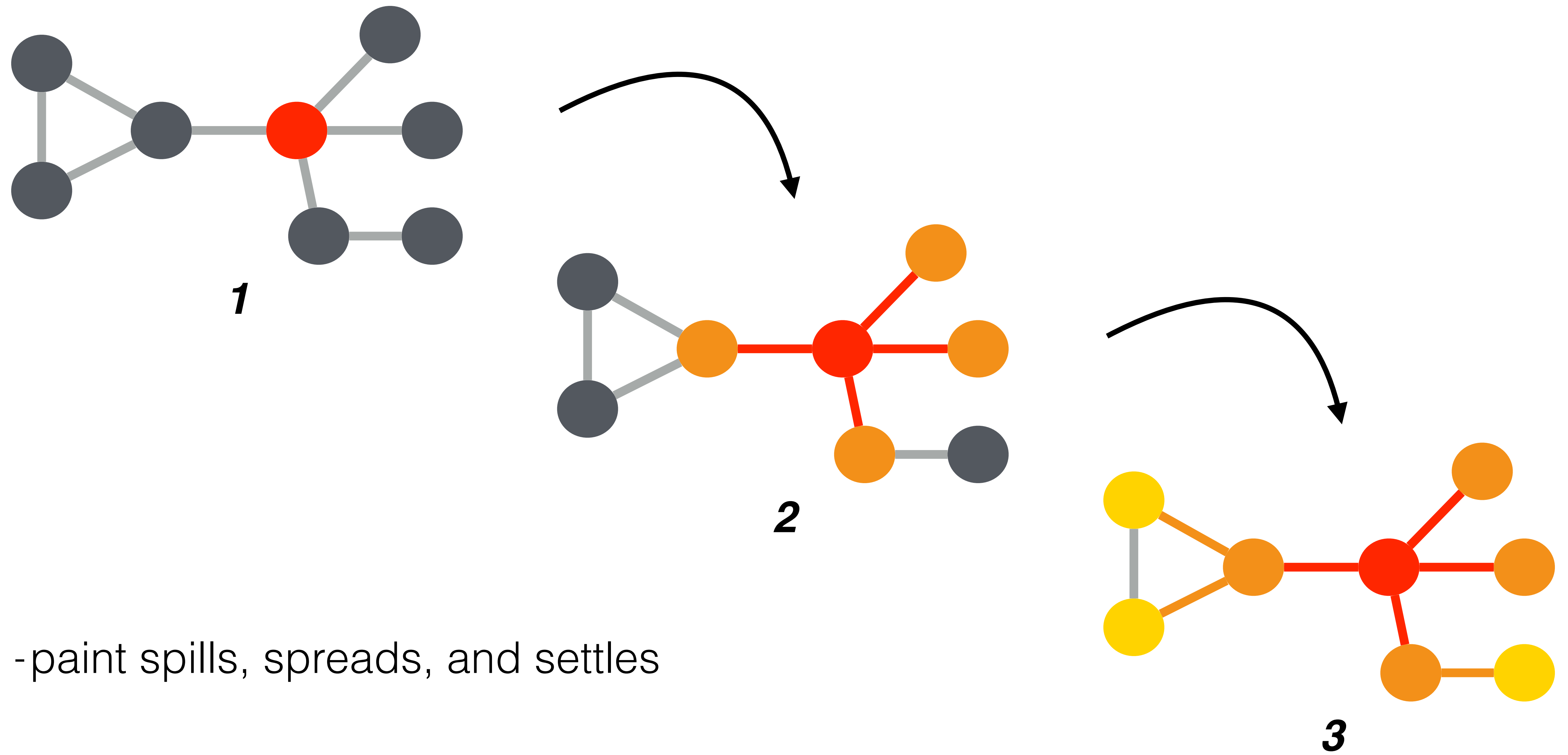
based on the  
dynamics of  
*random walks*

e.g., Approx. PageRank  
[Andersen *et al.*, 2006]

based on the  
dynamics of  
*network flows*

e.g., Capacity Releasing  
Diffusion [Wang *et al.*, 2017]

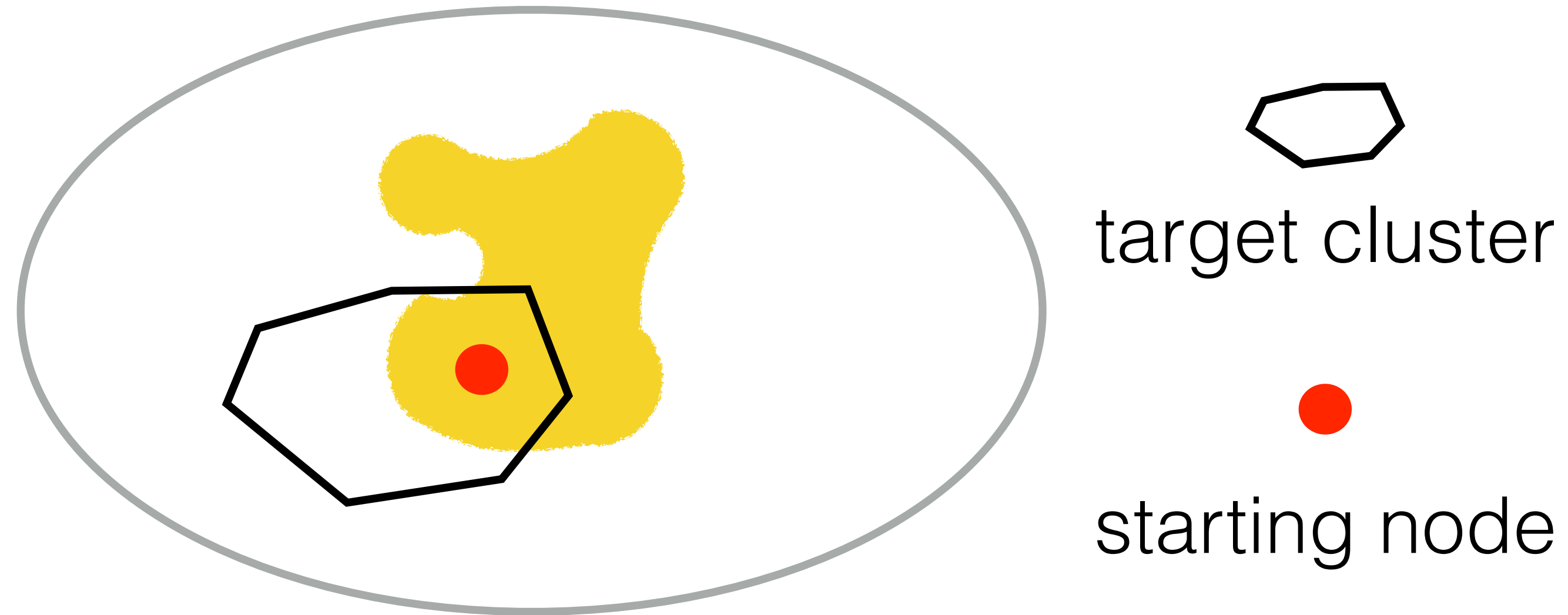
# Diffusion as physical phenomenon



-paint spills, spreads, and settles



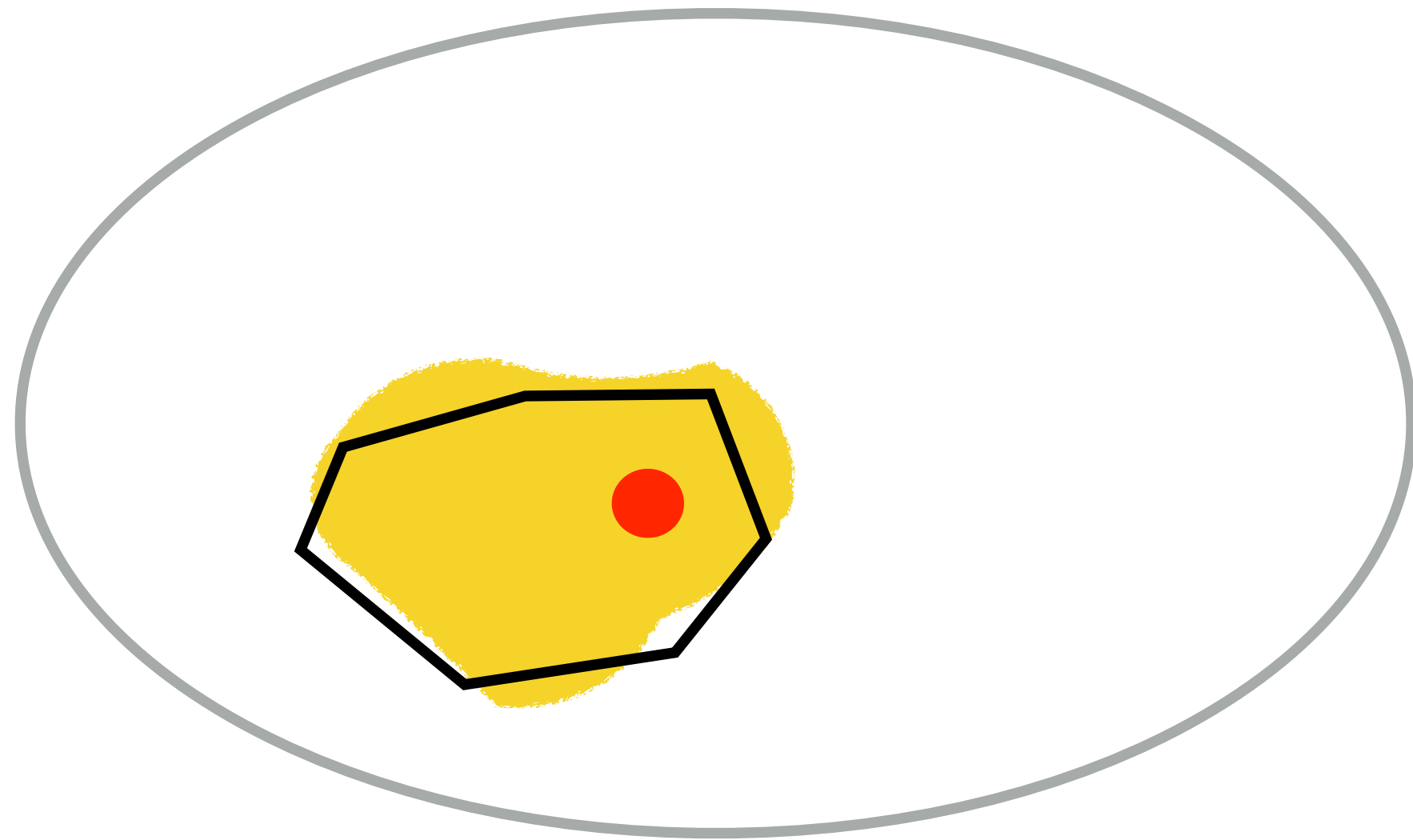
# Spectral diffusions leak mass



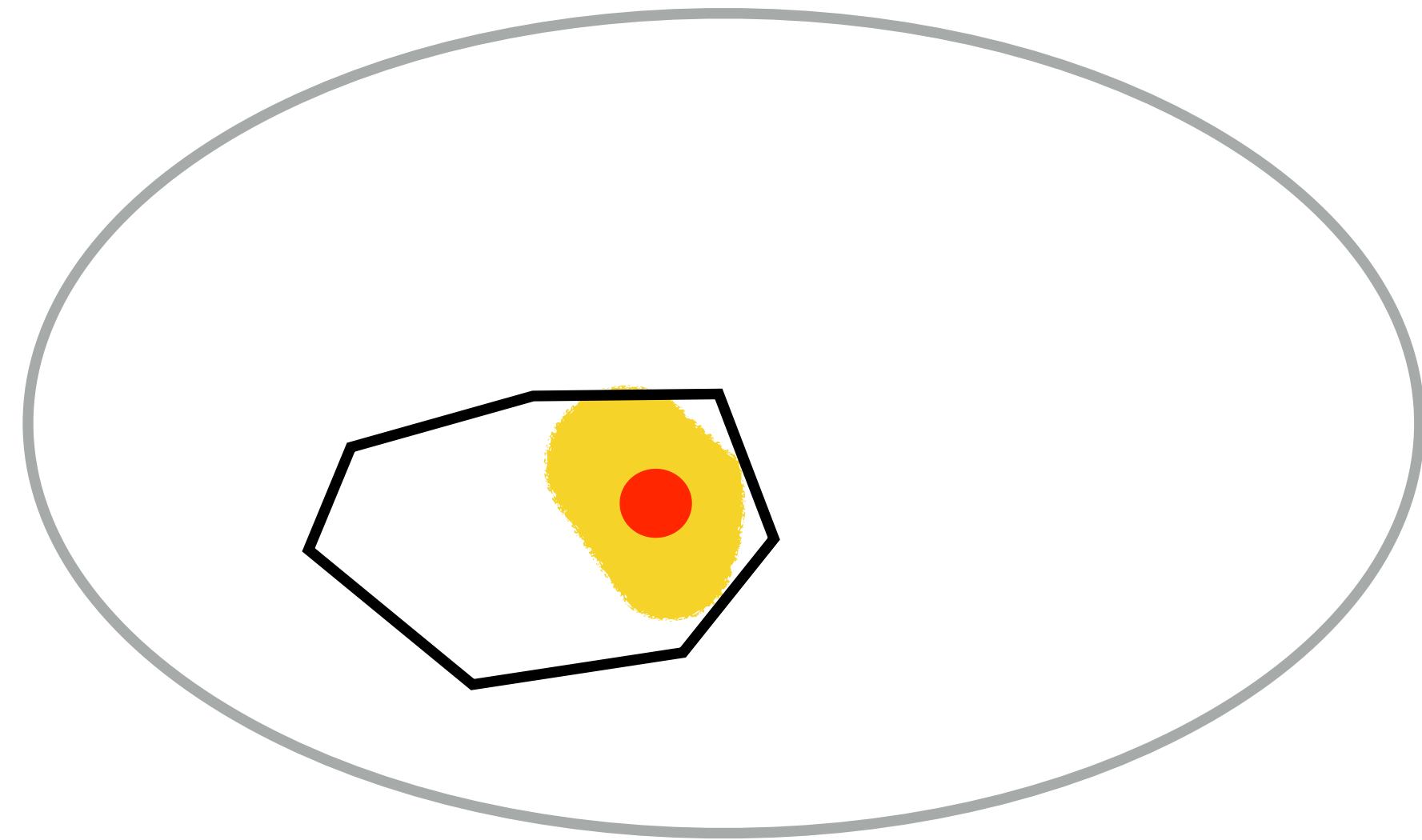
-low precision

-low recall

# Combinatorial diffusions are hard to tune



- strong theoretical guarantees
- work very well if tuned correctly



- poor performance if not tuned well

# New local graph clustering paradigm

**Spectral diffusions**

**Combinatorial diffusions**



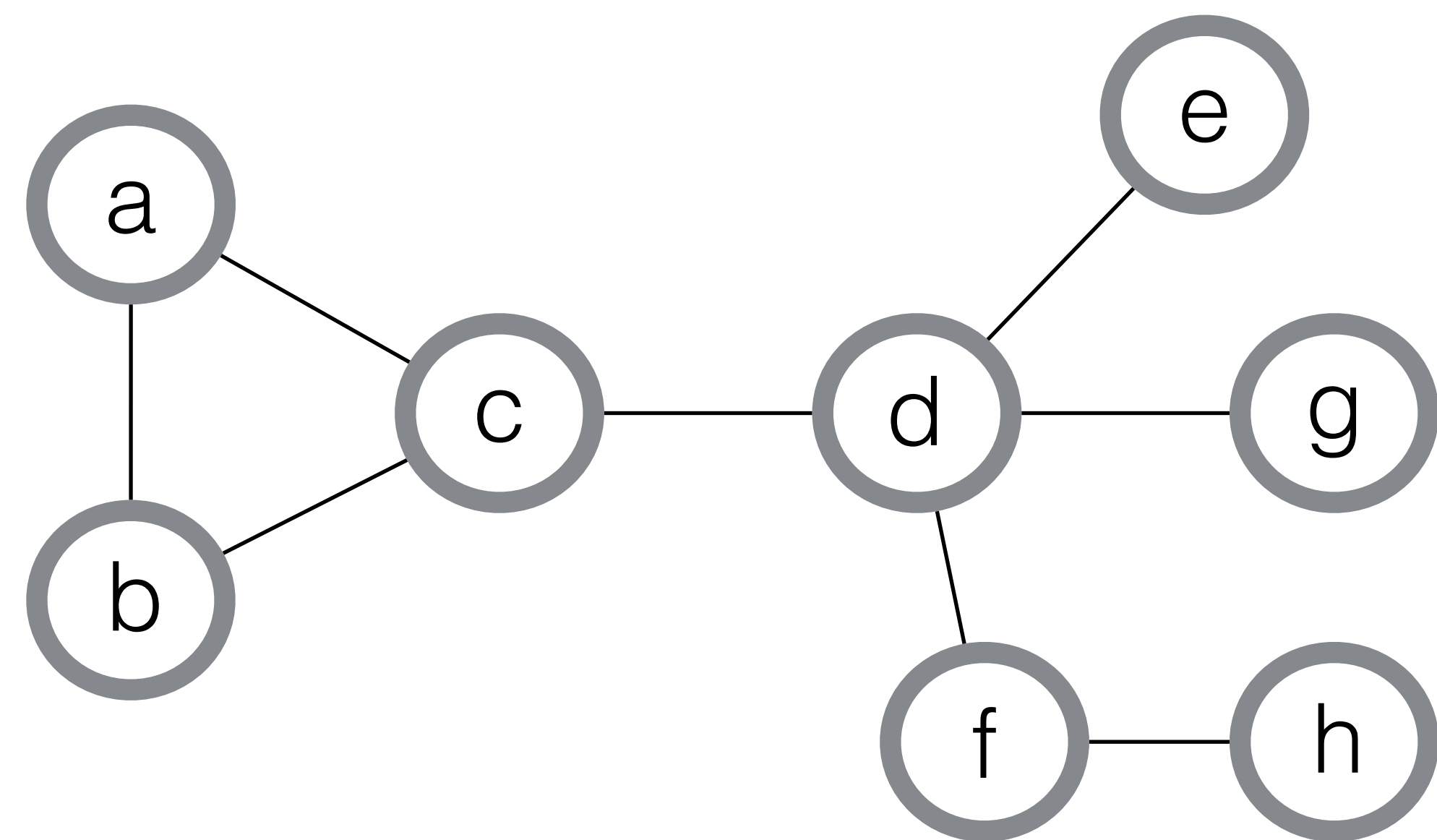
***p*-Norm flow diffusions**

based on the idea of  
*p*-norm network flow

- as **fast** as spectral methods 😊
- asymptotically as **strong** as combinatorial methods 😊
- intuitive interpretation, **simple** algorithm 😊
- **fewer tuning** parameters (than both spectral and combinatorial) 😊

# Notations and definitions

-Undirected graph  $G = (V, E)$



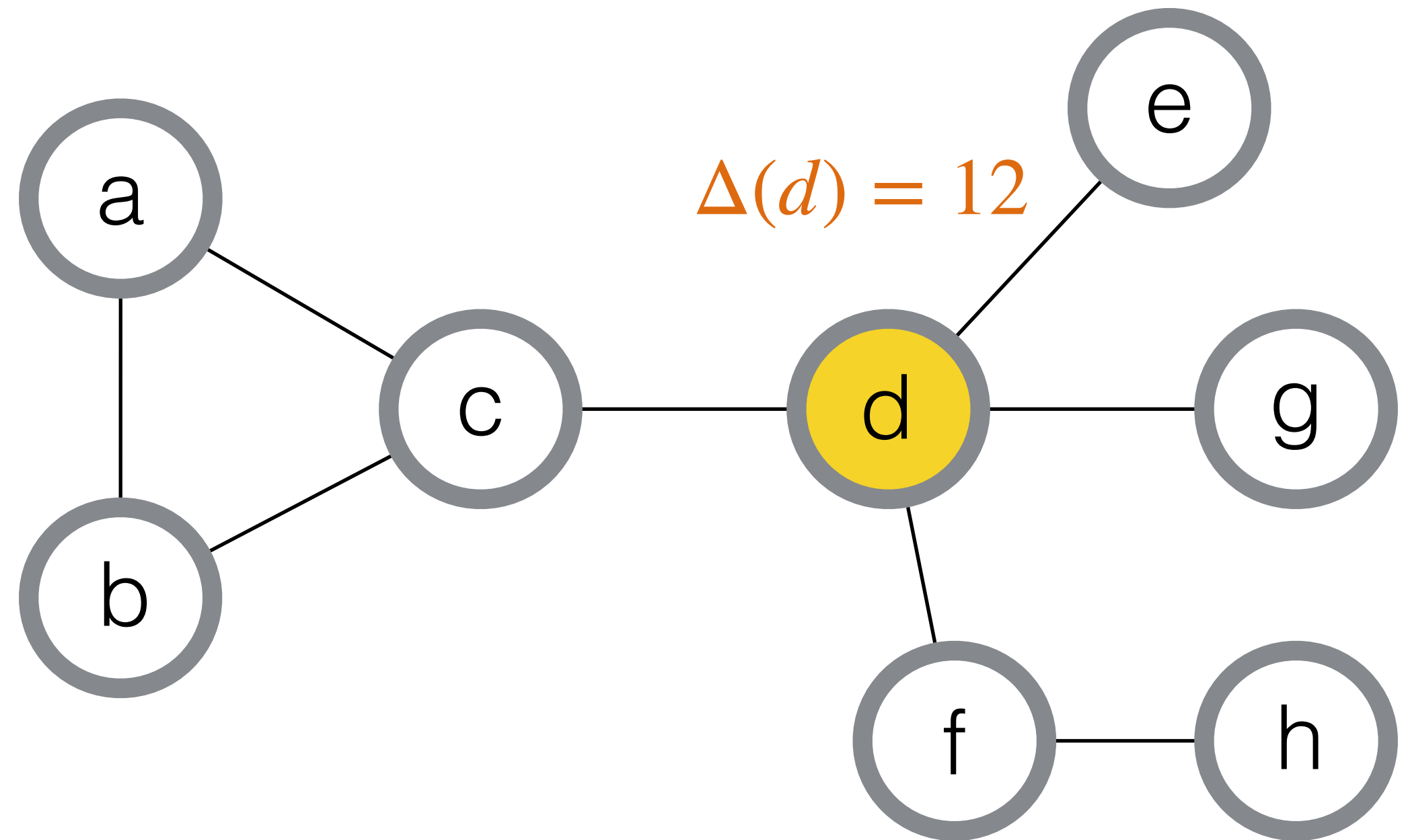
Incidence matrix B

|       | a | b  | c  | d  | e  | f  | g  | h  |
|-------|---|----|----|----|----|----|----|----|
| (a,b) | 1 | -1 |    |    |    |    |    |    |
| (a,c) | 1 |    | -1 |    |    |    |    |    |
| (b,c) |   | 1  | -1 |    |    |    |    |    |
| (c,d) |   |    | 1  | -1 |    |    |    |    |
| (d,e) |   |    |    | 1  | -1 |    |    |    |
| (d,f) |   |    |    | 1  |    | -1 |    |    |
| (d,g) |   |    |    | 1  |    |    | -1 |    |
| (f,h) |   |    |    |    |    | 1  |    | -1 |

- B is  $|E| \times |V|$  signed incidence matrix where the row of edge  $(u, v)$  has two non-zero entries, -1 at column  $u$  and 1 at column  $v$
- Ordering of edges and direction is arbitrary

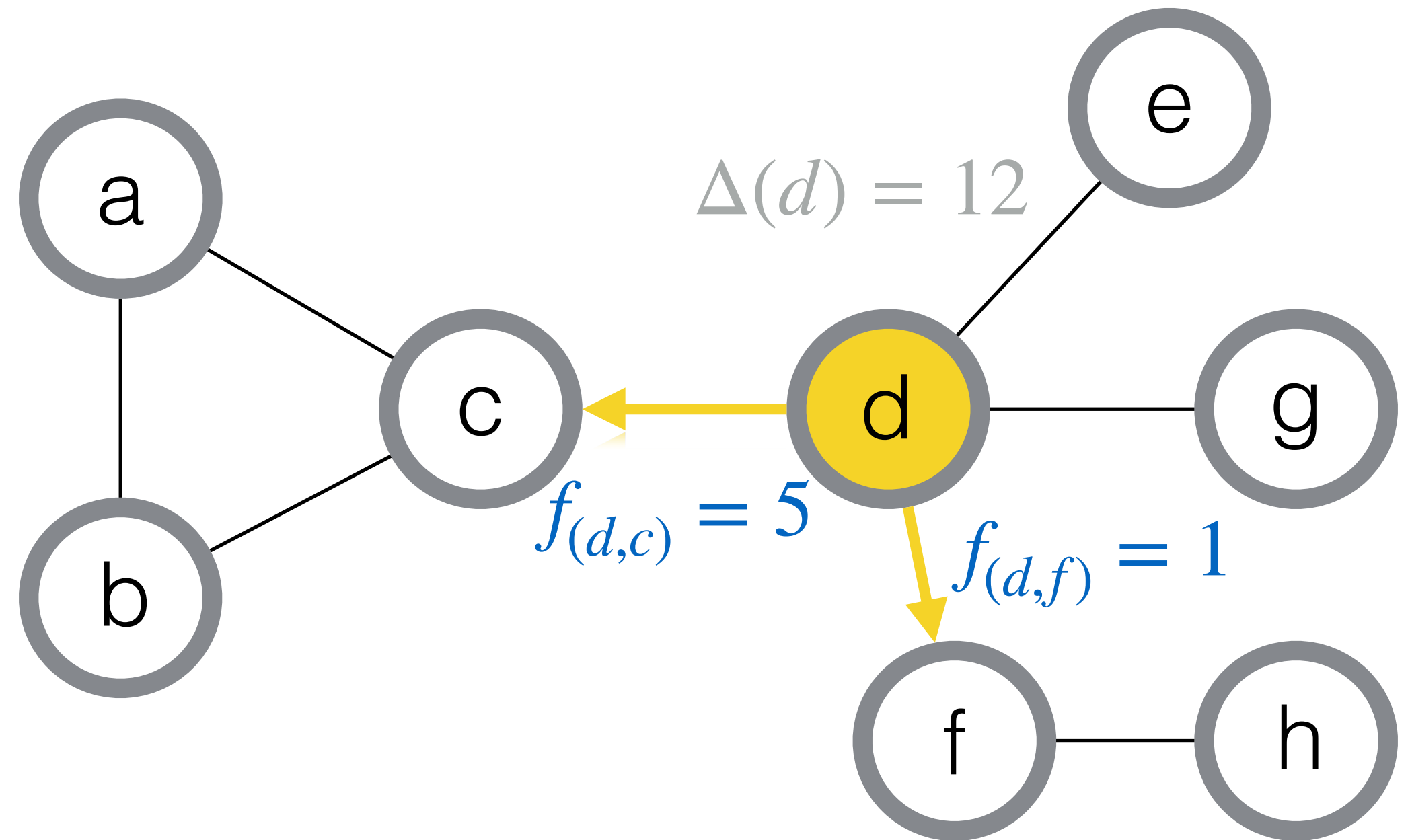
# Notations and definitions

-  $\Delta \in \mathbb{R}_+^{|V|}$  specifies **initial mass** on nodes.



# Notations and definitions

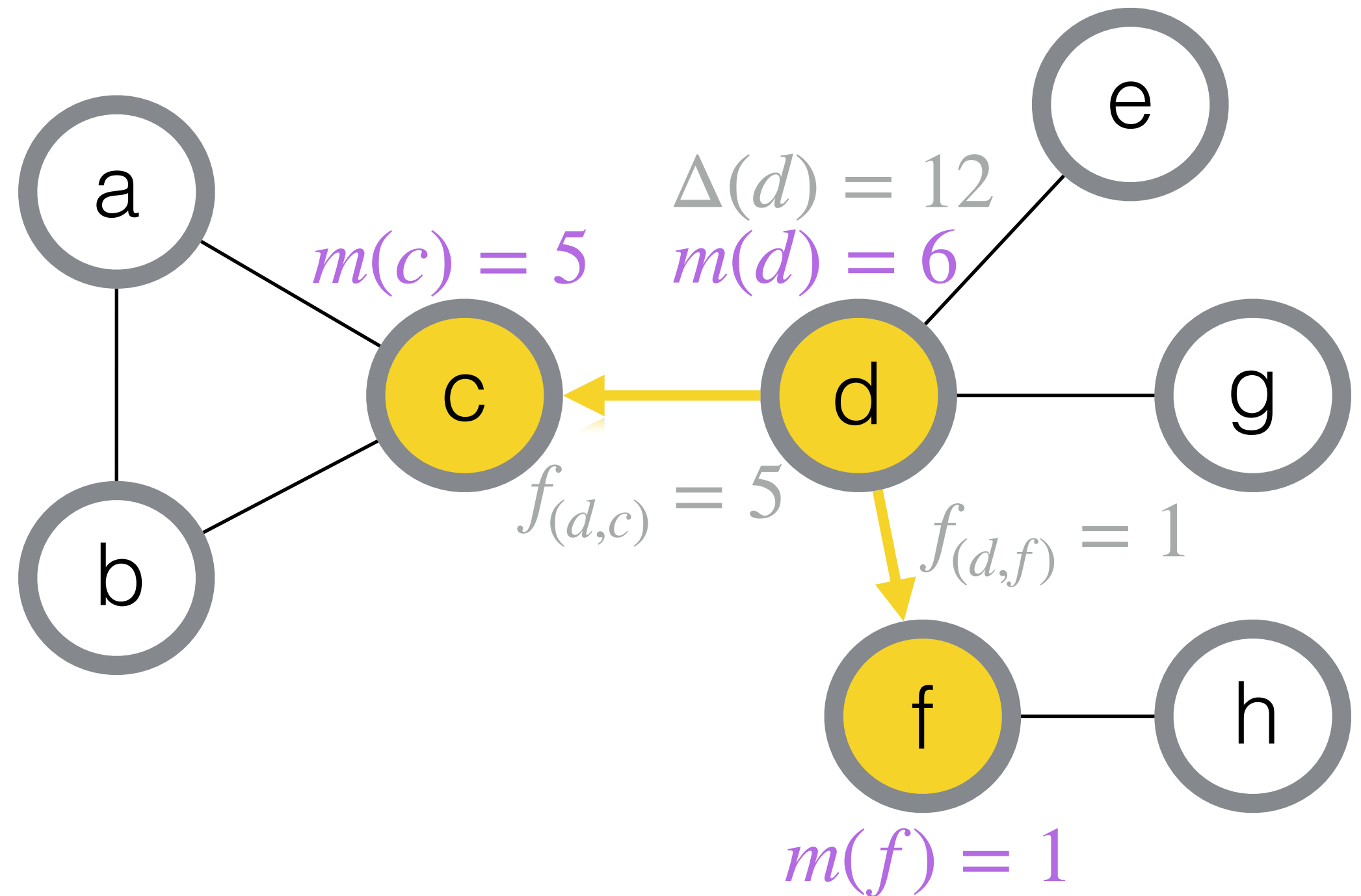
- $\Delta \in \mathbb{R}_+^{|V|}$  specifies initial mass on nodes.
- $f \in \mathbb{R}^{|E|}$  specifies the **amount of flow**.





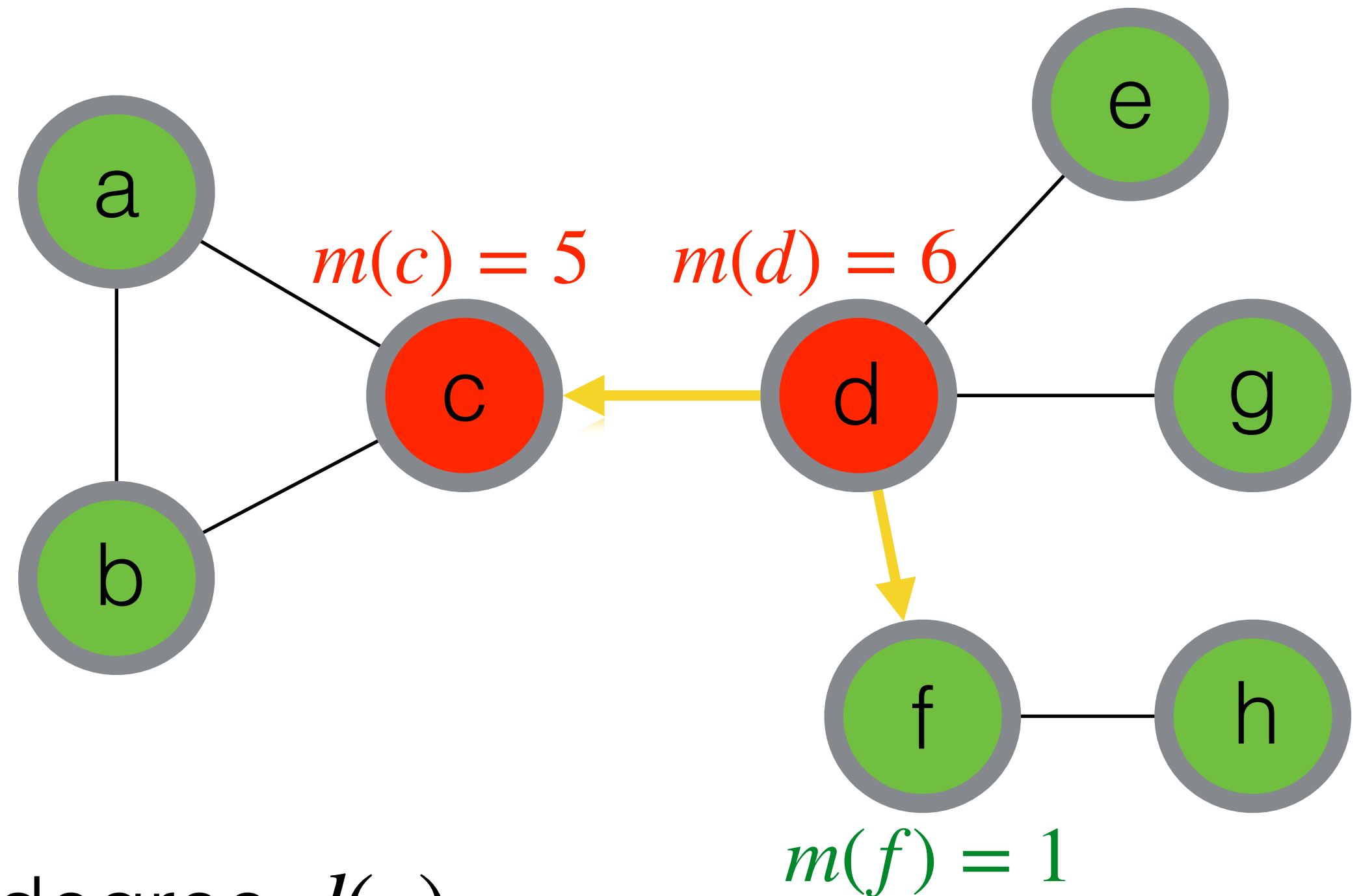
# Notations and definitions

- $\Delta \in \mathbb{R}_+^{|V|}$  specifies initial mass on nodes.
- $f \in \mathbb{R}^{|E|}$  specifies the amount of flow.
- $m := B^\top f + \Delta$  specifies **net mass** on nodes.



# Notations and definitions

- $\Delta \in \mathbb{R}_+^{|V|}$  specifies initial mass on nodes.
- $f \in \mathbb{R}^{|E|}$  specifies the amount of flow.
- $m := B^\top f + \Delta$  specifies net mass on nodes.
- Each node  $v$  has **capacity** equal to its degree  $d(v)$ .
- A flow  $f$  is **feasible** if  $[B^\top f + \Delta](v) \leq d(v), \forall v$ .



# $p$ -Norm flow diffusions - problem formulation

- We formulate **diffusion process on graph as optimization**:

$$\begin{array}{ll} \text{minimize } \|f\|_p & \xrightarrow{\text{green}} \text{Nonlinear} \text{ 😊} \\ \text{subject to: } B^\top f + \Delta \leq d & \xrightarrow{\text{green}} \text{Nonlinear} \text{ 😊} \\ & \searrow \text{red} \text{ Only one tuning parameter 😊} \end{array}$$

- Out of all feasible flows  $f$ , we are interested in the one having minimum  $p$ -norm, where  $p \in [2, \infty)$ .

# $p$ -Norm flow diffusions - problem formulation

- We formulate diffusion process on graph as optimization:

$$\begin{aligned} & \text{minimize } \|f\|_p \\ & \text{subject to: } B^\top f + \Delta \leq d \end{aligned}$$

- **Versatility:** different  $p$ -norm flows explore different structures in a graph

- **Locality:**  $\|f^*\|_0 \leq |\Delta| := \sum_{v \in V} \Delta(v)$

# $p$ -Norm flow diffusions - problem formulation

- We formulate diffusion process on graph as optimization:

$$\begin{aligned} & \text{minimize } \|f\|_p \\ & \text{subject to: } B^\top f + \Delta \leq d \end{aligned}$$

- The **dual** problem provides node embeddings

$$\begin{aligned} & \text{minimize } x^\top (d - \Delta) \longrightarrow \text{Biased towards seed node} \\ & \text{subject to: } \|Bx\|_q \leq 1 \\ & \quad \quad \quad x \geq 0 \end{aligned}$$

$1/p + 1/q = 1$

- Obtain a cluster by applying **sweep cut** on  $x$

# $p$ -Norm flow diffusions - local clustering guarantees

- Conductance of target cluster  $C$

$$\phi(C) = \frac{|\{(u, v) \in E : u \in C, v \notin C\}|}{\min \{\mathbf{vol}(C), \mathbf{vol}(V \setminus C)\}} \quad \text{where } \mathbf{vol}(C) := \sum_{v \in C} d(v)$$

- Seed set  $S := \text{supp}(\Delta)$ .

- Assumption (sufficient overlap):  
 $\mathbf{vol}(S \cap C) \geq \beta \mathbf{vol}(S)$   
 $\mathbf{vol}(S \cap C) \geq \alpha \mathbf{vol}(C)$       $\alpha, \beta \geq \frac{1}{\log^t \mathbf{vol}(C)}$  for some  $t$

- The output cluster  $\tilde{C}$  satisfies

$$\phi(\tilde{C}) \leq \tilde{\mathcal{O}}(\phi(C)^{1-1/p})$$

- Cheeger-type bound  $\phi(\tilde{C}) \leq \tilde{\mathcal{O}}(\sqrt{\phi(C)})$  for  $p = 2$

- Constant approximate  $\phi(\tilde{C}) \leq \tilde{\mathcal{O}}(\phi(C))$  for  $p \rightarrow \infty$



# $p$ -Norm flow diffusions - local clustering guarantees

- Conductance of target cluster  $C$

$$\phi(C) = \frac{|\{(u, v) \in E : u \in C, v \notin C\}|}{\min \{\text{vol}(C), \text{vol}(V \setminus C)\}} \quad \text{where } \text{vol}(C) := \sum_{v \in C} d(v)$$

- Seed set  $S := \text{supp}(\Delta)$ .

- Assumption (sufficient overlap):  $\begin{array}{l} \text{vol}(S \cap C) \geq \beta \text{vol}(S) \\ \text{vol}(S \cap C) \geq \alpha \text{vol}(C) \end{array} \quad \alpha, \beta \geq \frac{1}{\log^t \text{vol}(C)} \text{ for some } t$

- The output cluster  $\tilde{C}$  satisfies

$$\phi(\tilde{C}) \leq \tilde{\mathcal{O}}(\phi(C)^{1-1/p})$$

- Cheeger-type bound  $\phi(\tilde{C}) \leq \tilde{\mathcal{O}}(\sqrt{\phi(C)})$  for  $p = 2$

- Constant approximate  $\phi(\tilde{C}) \leq \tilde{\mathcal{O}}(\phi(C))$  for  $p \rightarrow \infty$

Proof based on analysis of primal and dual objective and constraints.

Larger  $p$  penalizes more on the flows that cross “bottleneck” edges, leading to less leakage.

# $p$ -Norm flow diffusions - simple strongly local algorithm

- Solve an **equivalent penalized** dual formulation by a variant of randomized coordinate descent.

**Initially** each node has a net mass equals the initial mass.

**Iterate:**

Pick a node  $v$  whose net mass exceeds its capacity.

Send excess mass to its neighbors.

Update net mass.

# $p$ -Norm flow diffusions - simple strongly local algorithm

- Solve an equivalent penalized dual formulation by a variant of randomized coordinate descent.

**Initially** each node has a net mass equals the initial mass.

**Iterate:**

Pick a node  $v$  whose net mass exceeds its capacity.

Send excess mass to its neighbors.

Update net mass.

- Worst-case running time  $\mathcal{O} \left( |\Delta| \left( \frac{|\Delta|}{\epsilon} \right)^{\boxed{2/q-1}} \log \frac{1}{\epsilon} \right)$ .

Natural tradeoff between  
speed and robustness to noise

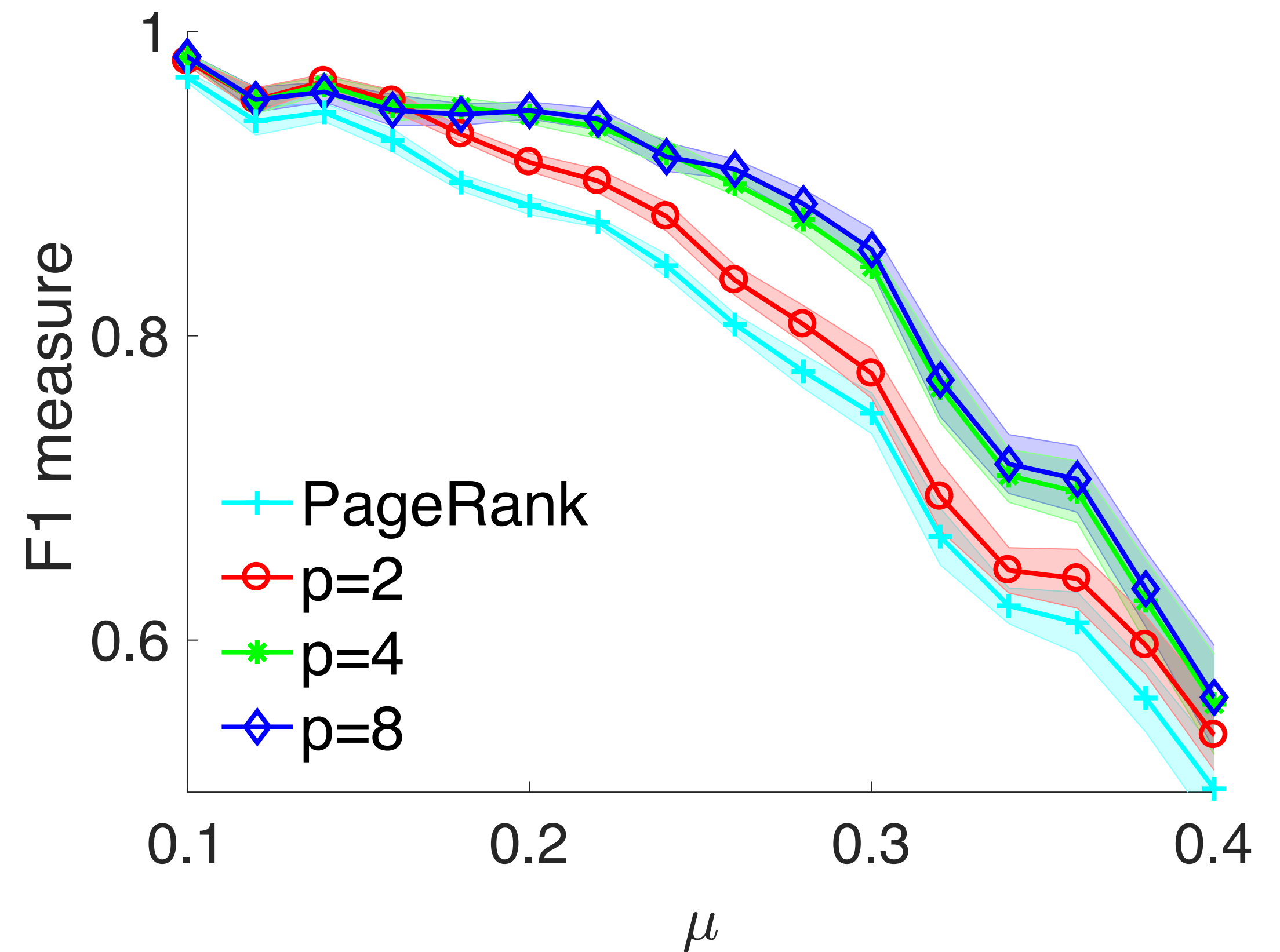
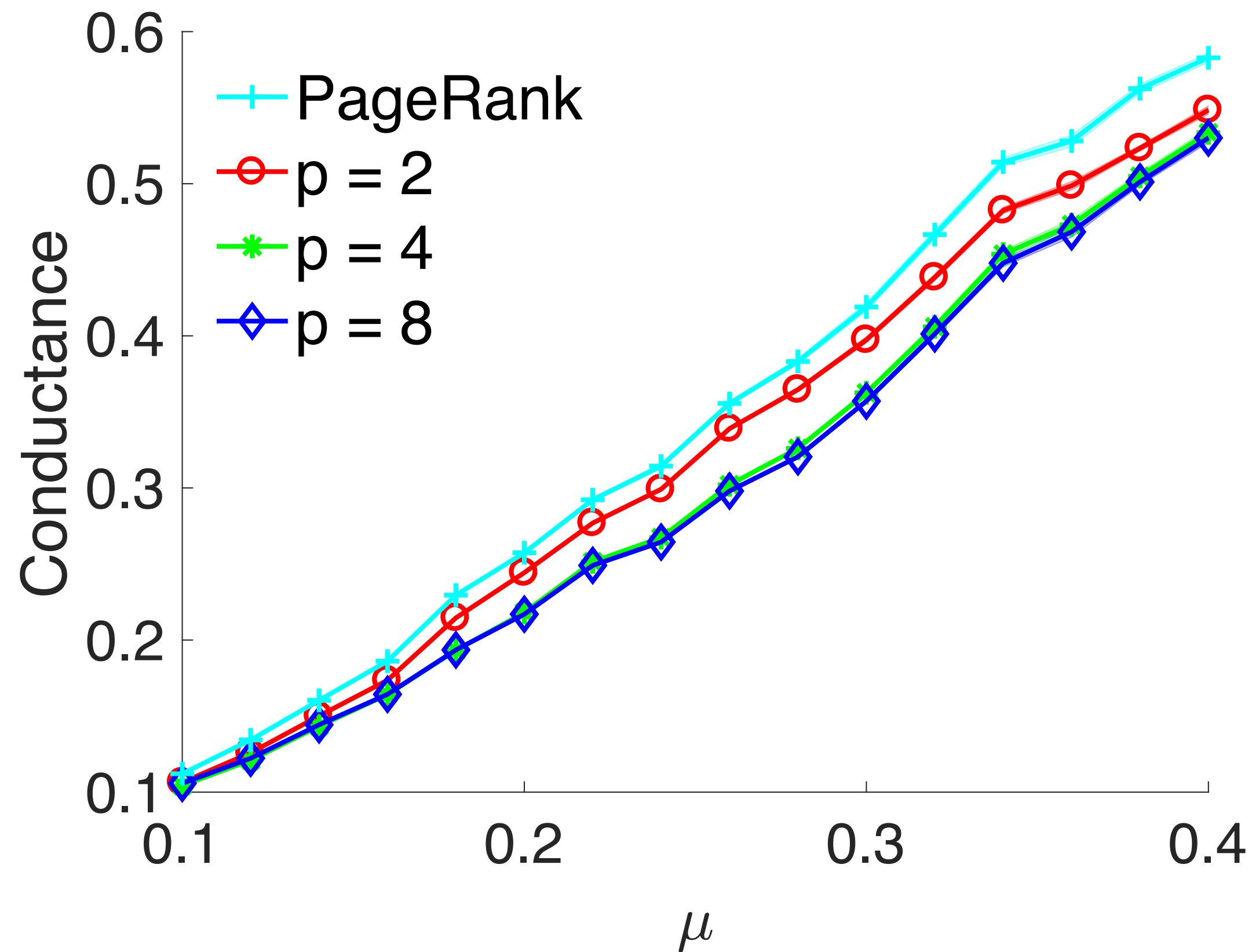
Total amount of initial mass

- Linear convergence when  $q = 2$ .

# $p$ -Norm flow diffusions - empirical performance

-LFR synthetic model

- $\mu$  is a parameter that controls noise, the higher the more noise.



# $p$ -Norm flow diffusions - empirical performance

- Facebook social network for Colgate University, students in Class of 2009

|             | PageRank | $p = 2$ | $p = 4$     |
|-------------|----------|---------|-------------|
| Conductance | 0.13     | 0.13    | <b>0.12</b> |
| F1 measure  | 0.96     | 0.96    | <b>0.97</b> |

*very clean  
ground  
truth*

- Facebook social network for Johns Hopkins University, students of the same major

|             | PageRank | $p = 2$ | $p = 4$     |
|-------------|----------|---------|-------------|
| Conductance | 0.25     | 0.23    | <b>0.22</b> |
| F1 measure  | 0.83     | 0.85    | <b>0.87</b> |

*average  
ground  
truth*

- Orkut, large-scale on-line social network, user-defined group

|             | PageRank | $p = 2$ | $p = 4$     |
|-------------|----------|---------|-------------|
| Conductance | 0.37     | 0.35    | <b>0.33</b> |
| F1 measure  | 0.66     | 0.71    | <b>0.73</b> |

*very noisy  
ground  
truth*

# Julia implementation: **pNormFlowDiffusion** on **GitHub**

- Includes demonstrations and visualizations on LFR and Facebook social networks.
- Contains all code to reproduce the results in our paper, “p-Norm flow diffusion for local graph clustering”, ICML 2020.

|  | Local<br>running time,<br><b>fast computation</b> | Good<br><b>theoretical<br/>guarantee</b> | Simple algorithm,<br><b>less tuning</b> |
|--|---|--|---|
| <b>Spectral diffusion</b><br>(e.g. PageRank) | ✓✓  | ✗  | ✗                                       |
| <b>Combinatorial diffusion</b><br>(e.g. CRD) | ✓   | ✓  | ✗✗                                      |
| <b>p-Norm flow diffusion</b>                 | ✓✓  | ✓  | ✓                                       |



**Thank you!**