

Comparing Hierarchical Network Partitions Based on Relative Entropy

Learning on Graphs Conference 2024

Christopher Blöcker^{1,2} and Ingo Scholtes^{1,2}

`christopher.bloecker@uni-wuerzburg.de`

¹Chair of Machine Learning for Complex Networks, Center for AI and Data Science, University of Würzburg, Germany

²Data Analytics Group, Department of Informatics, University of Zurich, Switzerland



**Universität
Zürich**^{UZH}

too long; didn't listen

- Networks can be partitioned in many different ways
 - we often compare partitions
- Common partition-similarity scores ignore link patterns
 - Jaccard index, variants of mutual information, ...
- We propose *flow divergence*
 - link-aware partition-similarity score
 - description of random walks + principle behind Kullback-Leibler divergence

Comparing Partitions

Comparing Partitions

Jaccard index

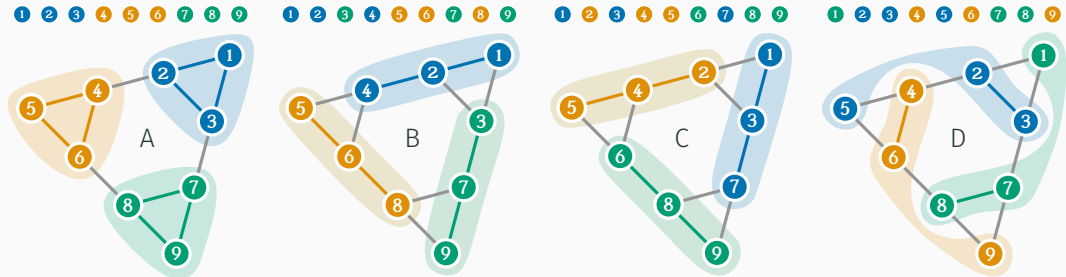
$$J(a, b) = \frac{|a \cap b|}{|a \cup b|}$$
$$J(A, B) = \sum_{a \in A} \frac{|a|}{n} \max_{b \in B} J(a, b)$$



Similar for mutual information etc.

Comparing *Network* Partitions

Comparing Network Partitions



$$J(A, B) = J(A, C) = J(A, D) = \frac{1}{2}$$

But what about the links?

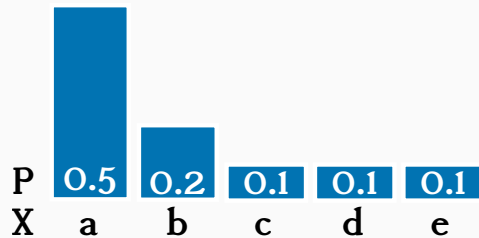
Idea:

KL divergence + random walks

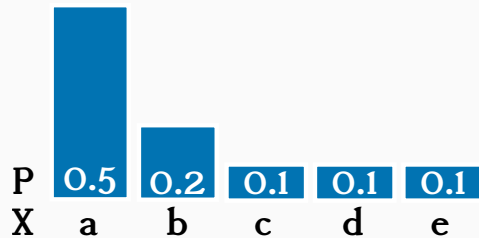
Kullback-Leibler divergence

X	a	b	c	d	e
---	---	---	---	---	---

Kullback-Leibler divergence

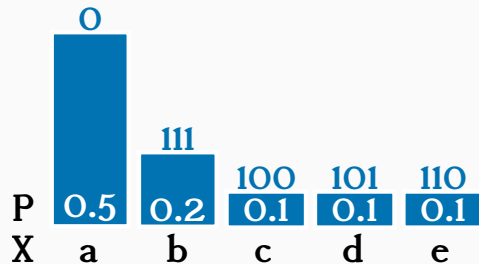


Kullback-Leibler divergence



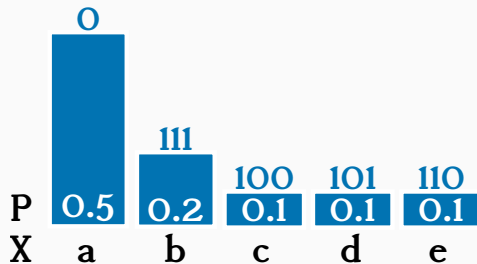
a a c d b b a b e c

Kullback-Leibler divergence



a a c d b b a b e c

Kullback-Leibler divergence

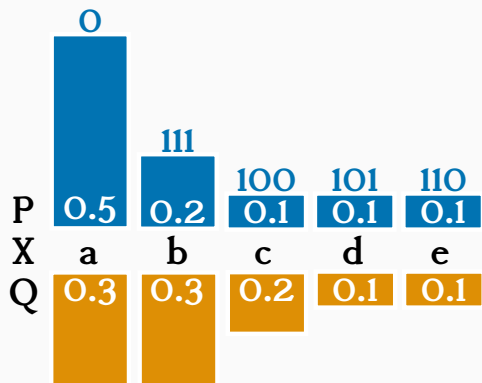


$$H(P) = - \sum_{x \in X} p_x \log_2 p_x$$

$$H(P) \approx 1.96 \text{ bits}$$

0 0 100 101 111 111 0 111 110 100
a a c d b b a b e c

Kullback-Leibler divergence

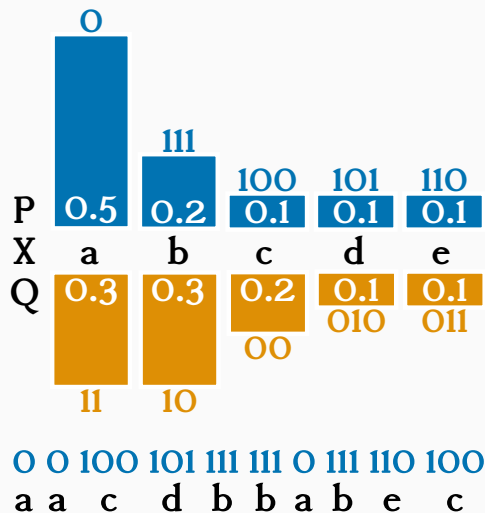


$$H(P) = - \sum_{x \in X} p_x \log_2 p_x$$

$$H(P) \approx 1.96 \text{ bits}$$

0 0 100 101 111 111 0 111 110 100
a a c d b b a b e c

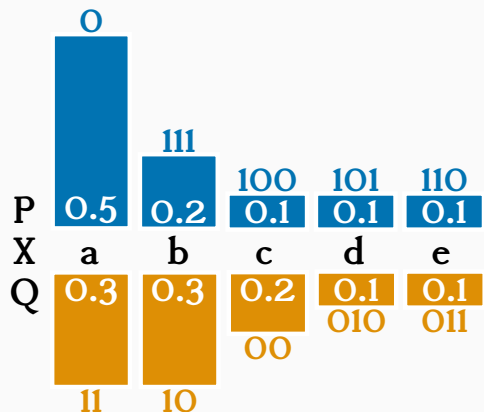
Kullback-Leibler divergence



$$H(P) = - \sum_{x \in X} p_x \log_2 p_x$$

$$H(P) \approx 1.96 \text{ bits}$$

Kullback-Leibler divergence



0 0 100 101 111 111 0 111 110 100
 a a c d b b a b e c
 11 11 00 010 10 10 11 10 011 00

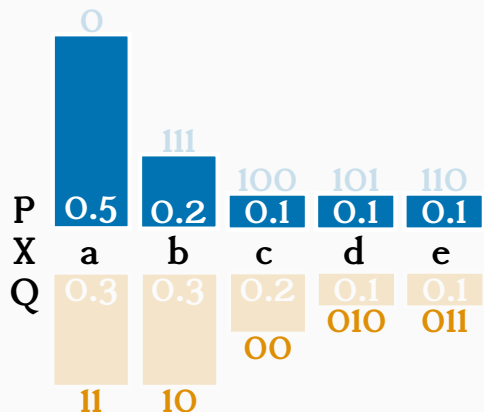
$$H(P) = - \sum_{x \in X} p_x \log_2 p_x$$

$$H(P) \approx 1.96 \text{ bits}$$

$$H(Q) \approx 2.17 \text{ bits}$$

$$D_{KL}(P || Q) = \sum_{x \in X} p_x \log_2 \frac{p_x}{q_x}$$

Kullback-Leibler divergence



0 0 100 101 111 111 0 111 110 100
 a a c d b b a b e c
 11 11 00 010101010 11 10 011 00

$$H(P) = - \sum_{x \in X} p_x \log_2 p_x$$

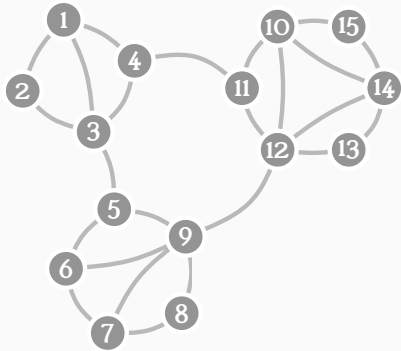
$$H(P) \approx 1.96 \text{ bits}$$

$$H(Q) \approx 2.17 \text{ bits}$$

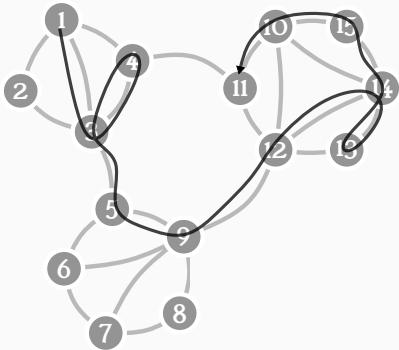
$$D_{KL}(P || Q) = \sum_{x \in X} p_x \log_2 \frac{p_x}{q_x}$$

$$D_{KL}(P || Q) \approx 0.15 \text{ bits}$$

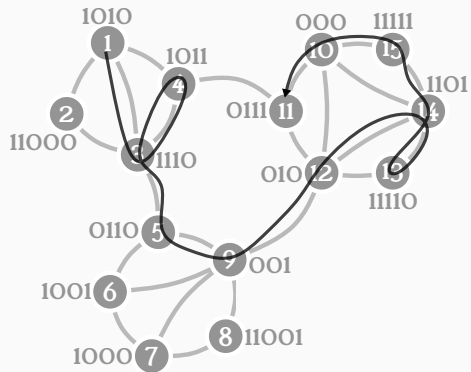
Random Walks



Random Walks

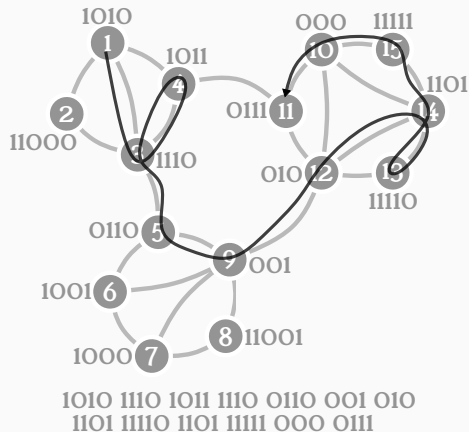


Random Walks



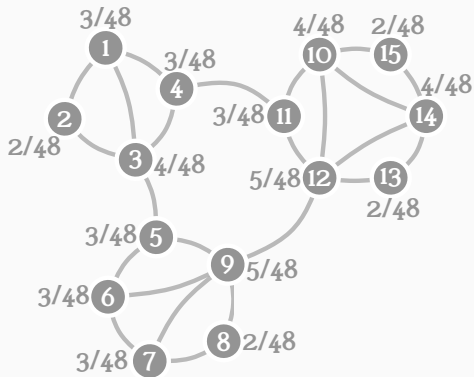
Random Walks

Description length



$$H(P) = - \sum_{v \in V} p_v \log_2 p_v \text{ bits} \quad (1)$$

Random Walks



Description length

$$H(P) = - \sum_{v \in V} p_v \log_2 p_v \text{ bits} \quad (1)$$

Visit rates

$$p_v = \sum_{u \in V} p_u t_{uv} \quad t_{uv} = \frac{w_{uv}}{\sum_{v \in V} w_{uv}} \quad (2)$$

Combine (1)+(2)

$$H(P) = - \sum_{u \in V} p_u \sum_{v \in V} t_{uv} \log_2 p_v$$

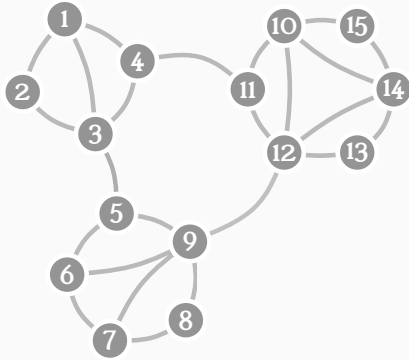
But we are interested in communities...

$$H(M) = - \sum_{u \in V} p_u \sum_{v \in V} t_{uv} \log_2 s(M, u, v)$$

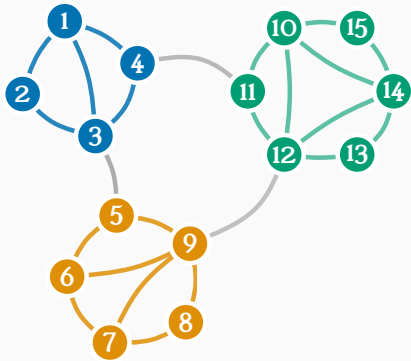
The Map Equation

→ Rosvall and Bergstrom; PNAS 2008

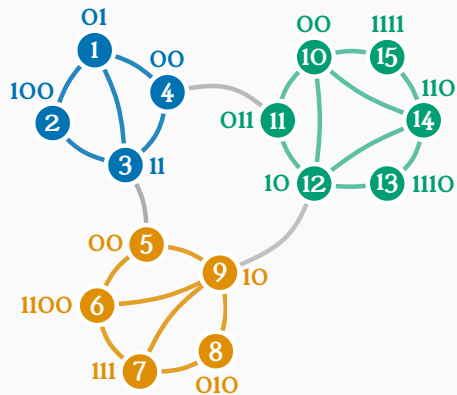
The Map Equation



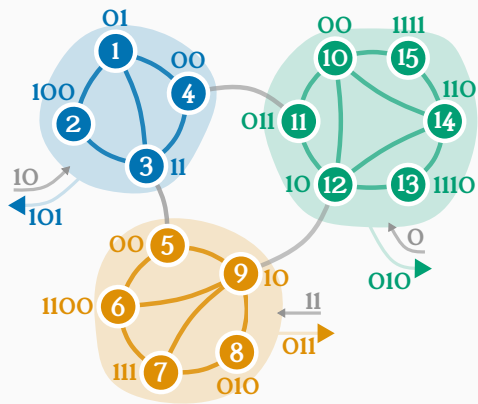
The Map Equation



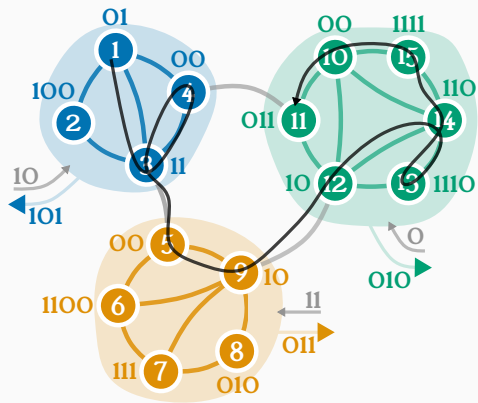
The Map Equation



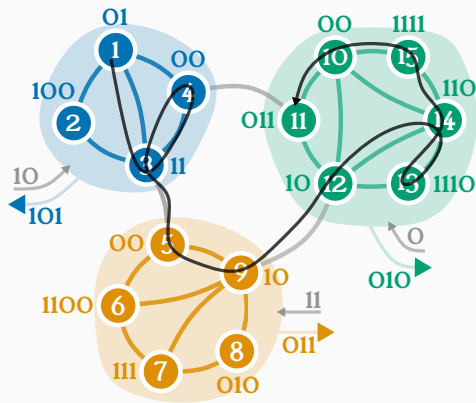
The Map Equation



The Map Equation



The Map Equation

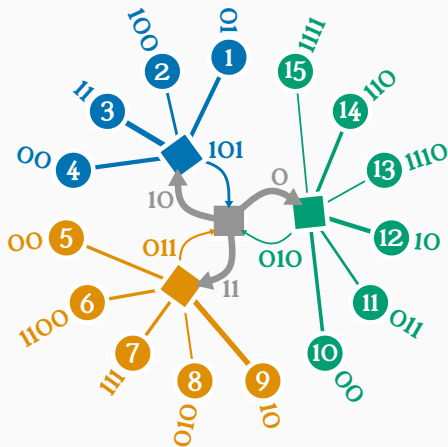


1001 11 00 11 1011 00 10 011 010
110 1110 110 1111 00 011

$$L(M) = qH(Q) + \sum_{m \in M} p_m H(P_m) \rightarrow \min$$

$$= - \sum_{u \in V} p_u \sum_{v \in V} t_{uv} \log_2 \text{mapsim}(M, u, v)$$

The Map Equation

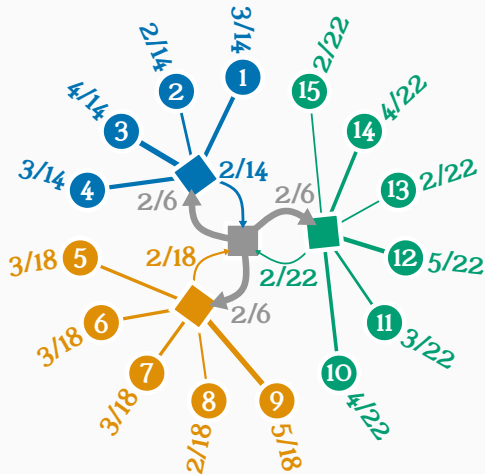


1001 11 00 11 1011 00 10 011 010
110 1110 110 1111 00 011

$$L(M) = qH(Q) + \sum_{m \in M} p_m H(P_m)$$

$$= - \sum_{u \in V} p_u \sum_{v \in V} t_{uv} \log_2 \text{mapsim}(M, u, v)$$

The Map Equation



$$L(M) = qH(Q) + \sum_{m \in M} p_m H(P_m)$$

$$= - \sum_{u \in V} p_u \sum_{v \in V} t_{uv} \log_2 \text{mapsim}(M, u, v)$$

$$\text{mapsim}(M, u, v) = \begin{cases} \frac{p_v}{p_{m_v}} & \text{if } m_u = m_v, \\ \frac{m_{u,\text{exit}}}{p_{m_u}} \cdot \frac{q_{m_v}}{q} \cdot \frac{p_v}{p_{m_v}} & \text{if } m_u \neq m_v, \end{cases}$$

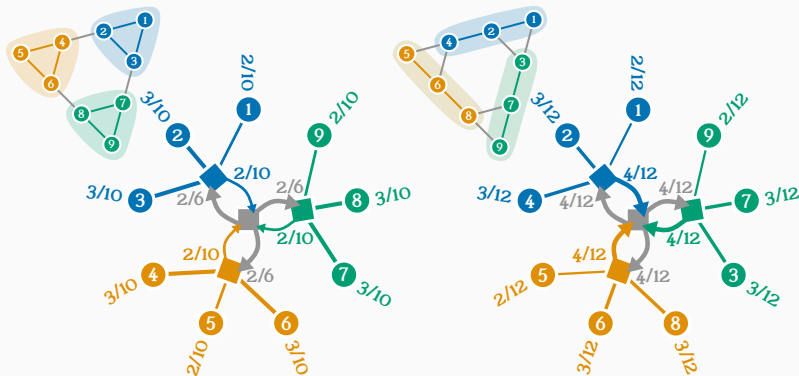
→ Blöcker et al.; PMLR 2022

Flow Divergence

Flow Divergence

Idea

Measure how many extra bits we need to describe a random walk using an “estimate” B of the networks “true” partition A.



Flow Divergence

$$D_{KL}(P \parallel Q) = \sum_{x \in X} p_x \log_2 \frac{p_x}{q_x} \qquad L(M) = - \sum_{u \in V} p_u \sum_{v \in V} t_{uv} \log_2 \text{mapsim}(M, u, v)$$

$$D(A \parallel B) = \sum_{u \in V} p_u \sum_{v \in V} t_{uv} \log_2 \frac{\text{mapsim}(A, u, v)}{\text{mapsim}(B, u, v)} \quad ?$$

$$\text{Not quite} \dots \quad D(A \parallel B) = L(A) - L(B)$$

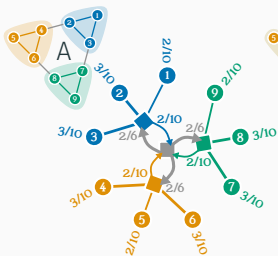
$$D_F(A \parallel B) = \sum_{u \in V} p_u \sum_{v \in V} t_{uv}^A \log_2 \frac{\text{mapsim}(A, u, v)}{\text{mapsim}(B, u, v)}$$

$$t_{uv}^A = \frac{\text{mapsim}(A, u, v)}{\sum_{v(\neq u)} \text{mapsim}(A, u, v)}$$

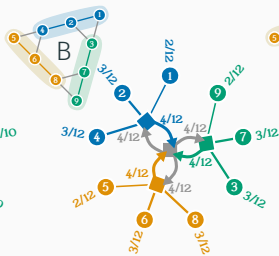
Naïvely, the complexity is $\mathcal{O}(n^2)$ because we consider all pairs of nodes
→ We're working on computing flow divergence more efficiently

Flow Divergence

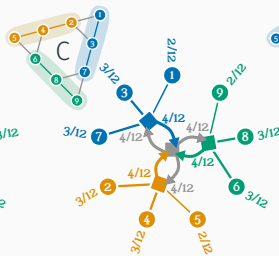
$L(A) \approx 2.86$ bits



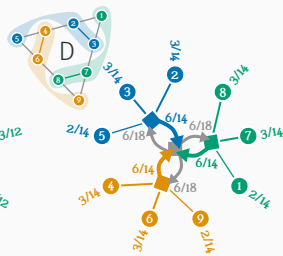
$L(B) \approx 3.73$ bits



$L(C) \approx 3.73$ bits



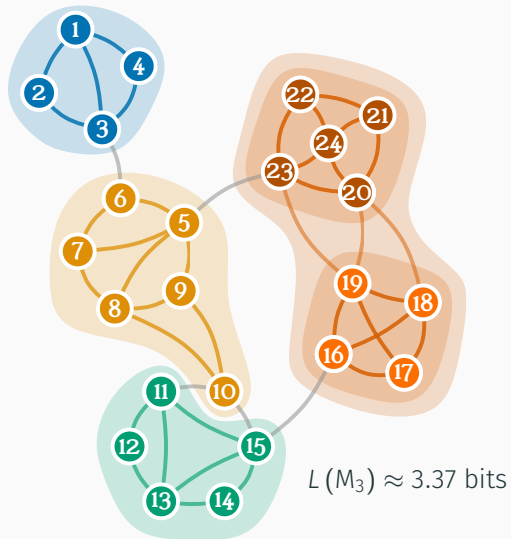
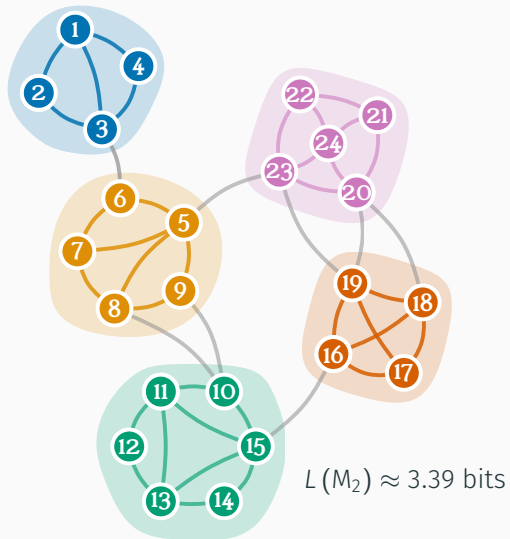
$L(D) \approx 4.47$ bits



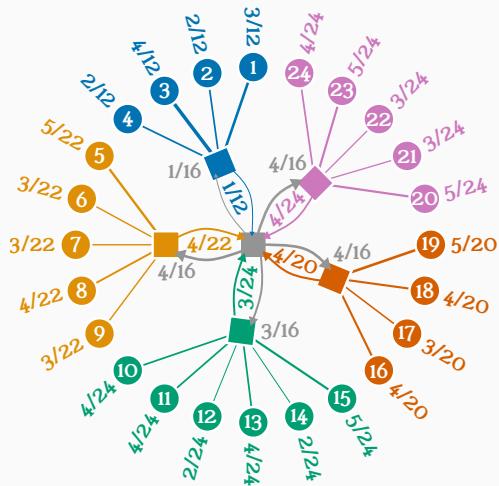
Other

Reference	A	B	C	D
A	0	1.92	1.92	1.5
B	1.8	0	1.17	1.99
C	1.8	1.17	0	1.48
D	1.14	1.78	1.25	0

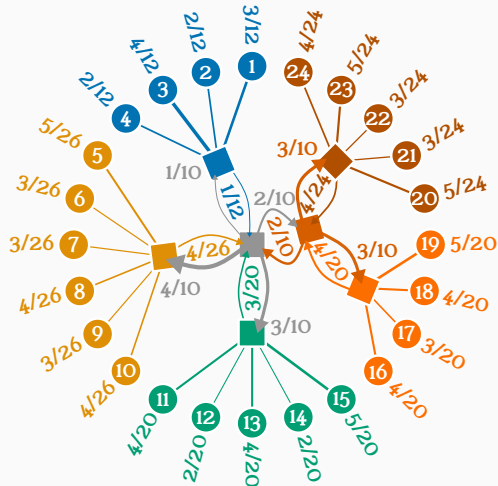
Flow Divergence



Flow Divergence



$$D_F(M_2 || M_3) \approx 0.11 \text{ bits}$$



$$D_F(M_3 \parallel M_2) \approx 0.01 \text{ bits}$$

Conclusion

Motivation

Communities are based on link patterns
but common partition-similarity scores ignore links.

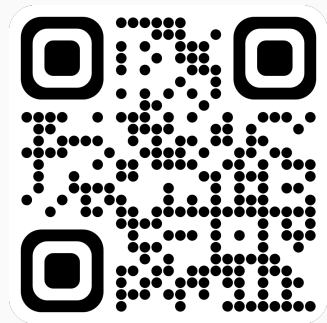
Our Solution

We combine the ideas behind describing random walks and the KL-divergence to develop a link-aware partition-similarity score.

- asymmetric
- does not require matching of communities between partitions
- naturally compares hierarchical partitions

Thank you for your attention!

Check out our preprint: [arXiv:2401.09052](https://arxiv.org/abs/2401.09052)

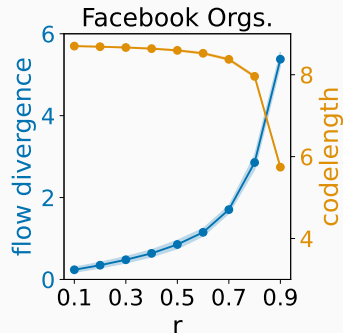
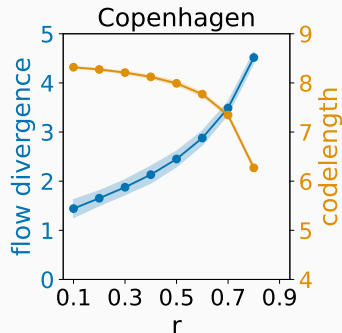
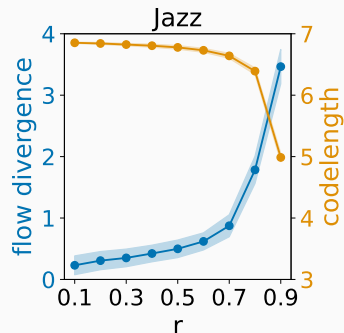


**Universität
Zürich**^{UZH}



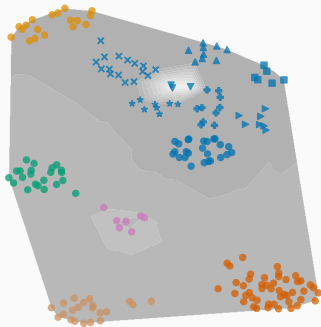
Appendix

Flow Divergence

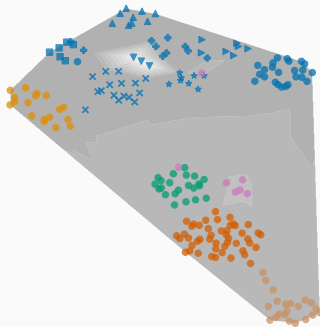


Flow Divergence

(a) Flow Divergence



(b) Jaccard Index



(c) Adjusted Mutual Information

