Mapping flows on hypergraphs

Anton Eriksson (anton.eriksson@umu.se)
Umeå University https://orcid.org/0000-0001-5859-4073
Daniel Edler
Umeå University
Alexis Rojas
Umeå University
Martin Rosvall
Umeå University https://orcid.org/0000-0002-7181-9940

Article

Keywords: mapping flows, hypergraphs

Posted Date: January 7th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-134751/v1

Version of Record: A version of this preprint was published at Communications Physics on June 11th, 2021. See the published version at https://doi.org/10.1038/s42005-021-00634-z.

Mapping flows on hypergraphs

Anton Eriksson,* Daniel Edler, Alexis Rojas, and Martin Rosvall

Integrated Science Lab,
Department of Physics,
Umeå University, SE-901 87 Umeå,
Sweden

(Dated: December 23, 2020)

Hypergraphs offer an explicit formalism to describe multibody interactions in complex systems. To connect dynamics and function in systems with these higher-order interactions, network scientists have generalised random walk models to hypergraphs and studied the multibody effects on flow-based centrality measures. But mapping the large-scale structure of those flows requires effective community detection methods. We derive unipartite, bipartite, and multilayer network representations of hypergraph flows and explore how they and the underlying random walk model change the number, size, depth, and overlap of identified multilevel communities. These results help researchers choose the appropriate modelling approach when mapping flows on hypergraphs.

Researchers model and map flows on networks to identify important nodes and detect significant communities[1,2,3,4]. From small to large system scales, random walk-based methods help to uncover the inner workings of the systems the networks represent[5,6]. When standard network models fail to adequately represent a system's interactions, researchers turn to higher-order models of complex systems[7,8], including multilayer networks[9,10,11] for multitype interactions, non-Markovian networks[12,13,14] for multistep interactions, and combinatorial models such as simplicial complexes[15,16,17,18] and hypergraphs[19,20,21,22] with nodes in hyperedges for multibody interactions.

While several methods can identify flow-based communities in multilayer[9,23,24] and memory[12,13,14] networks with non-Markovian dynamics, researchers have just begun to unravel the large-scale systemic effects of multibody interactions captured by hypergraphs[22]. However, different systems and research questions call for different random walk and hypergraph models: Random walks can be lazy and able to visit the same node multiple times in a row or non-lazy and forced to move on. Hyperedges can have arbitrary weights, and nodes can have hyperedge-dependent weights. Because these and other models can be represented with different network types – bipartite, unipartite, and multilayer networks – the questions multiply: How do different hypergraph random walk models combined with different network representations change the flow dynamics at scales captured by communities?

For example, random walks on hypergraphs can model flows of ideas in co-authorship networks. A node represents an author, and a hyperedge connects all authors of a paper. In the simplest dynamics, a random walker on a node picks a random hyperedge among those that contain the node and steps to a random node of the picked hyperedge. Then repeats. Excluding author self-links for non-lazy walks or including hyperedge weights from paper citations or using hyperedge-dependent node weights for varying author contributions are natural model variations that generate different dynamics[20,21]. How does the organisation of authors in nested communities from research groups to research areas change with random-walk model and representation?

For lazy random walks on hypergraphs with self-links and hyperedge-independent node weights, random walks on weighted, undirected networks generate equivalent dynamics[20]. Each hyperedge becomes a clique with properly adjusted link weights. This projection enables standard flow-based methods developed for weighted networks to identify communities where random walks stay for a long time. Non-lazy walks or walks with hyperedge-dependent node weights require directed networks[20]. A bipartite representation provides hyperedge assignments, and a multilayer representation enables overlapping communities.

Representing hypergraphs with bipartite networks requires weighted, directed links between two sets of nodes: one for the nodes and one for the hyperedges. Picking a random hyperedge becomes an explicit step to a hyperedge node. Non-lazy walks on the hypergraph require non-backtracking walks on the bipartite network[25]. With proper normalisation, the node-visit rates stay the same. Though unipartite and bipartite representations give identical node flows, the bipartite representation's link flows from nodes to hyperedge nodes and back to nodes can induce more flows between communities and alter the optimal community composition. The community-detection algorithm must also assign more nodes, which implies more degrees of freedom and a larger search space.

Multilayer networks represent the hyperedges as layers with fully connected groups of nodes. Each node is present in each of its hyperedge layers. Hyperedge weights become layer weights, and hyperedge-dependent node weights become layer-dependent node weights. Though the node visit rates aggregated over layers remain the same, multilayer networks multiply the degrees of freedom and enable new models. Reducing the inter-layer link weights increases the time a random walker spends within a hyperedge before moving to another. Reducing the inter-layer link weights only between dissimilar layers reinforces flows within similar layers. The search space expands when nodes can belong to multiple overlapping communities.

* anton.eriksson@umu.se

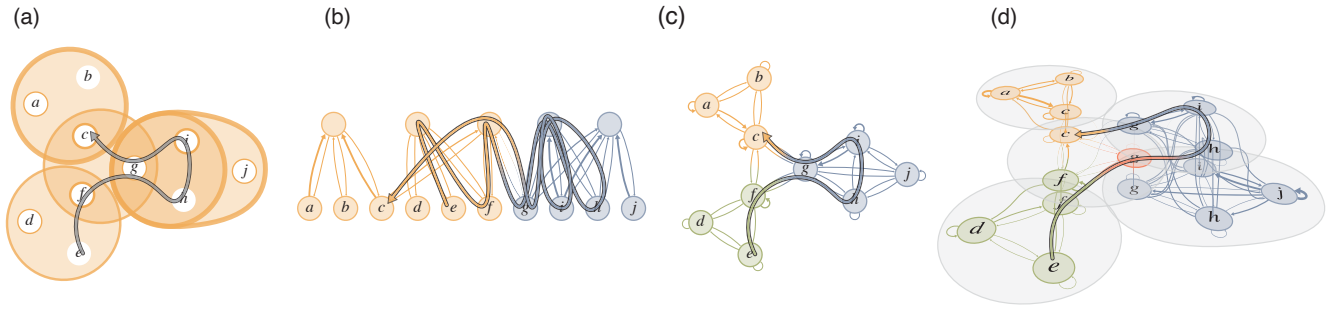

Fig. 1. A schematic hypergraph represented with three types of networks. (a) The schematic hypergraph with weighted hyperedges and hyperedge-dependent node weights. Thin borders for weight 1 and thick borders for weight 3. A lazy random walk on the schematic hypergraph represented on: (b) a bipartite network, (c) a unipartite network, and (d) a multilevel network. The colours indicate optimised module assignments, in (d) for hyperedge-similarity walks.

The many combinations of random walk models and representations available to address specific research problems call for the question: For different data and questions, which model and representation is best?

To address which combination of model and representation is best for answering different questions about various hypergraph data, we derive unipartite, bipartite, and multilayer network representations of hypergraph flows with identical node-visit rates for the same random-walk model. For unique node-visit rates when a representation requires directed links, we apply an unrecorded teleportation scheme robust to changes in the teleportation rate and that preserves the node-visit rates when teleportation is superfluous in undirected networks[26]. The information-theoretic and flow-based community detection method Infomap[27] allow us to explore how different hypergraph random walk models and network representation change the number, size, depth, and overlap of identified multilevel communities.

By analysing schematic and real hypergraphs, we find that the bipartite network representation requires the fewest links and enables the fastest community detection. A multilayer network representation that reinforces flows within similar layers give the deepest modular structures with most overlapping communities but at a high computational cost. The unipartite network representation provides a trade-off with intermediate compactness, speed, and detectable modular regularities.

Results and Discussion

Modelling flows on hypergraphs. We model flows on hypergraphs with random walks. We use hypergraphs with nodes V, hyperedges E with weights ω, and hyperedge-dependent node weights γ. Each hyperedge e has a weight $\omega(e)$. Each node u with incident hyperedges $E(u) = \{e \in E : u \in e\}$ has a weight $\gamma_e(u)$ for each incident hyperedge e. To simplify the notation when normalising weights into probabilities, we denote node u's total incident hyperedge weight $d(u) = \sum_{e \in E(u)} \omega(e)$ and hyperedge e' total node weight $\delta(e) = \sum_{u \in e} \gamma_e(u)$[20]. With these weights, a lazy random walker moves from node u at time t to node v at time $t+1$ in three steps by[20]:

1. Picking hyperedge e among node u's hyperedges $E(u)$ with probability $\frac{\omega(e)}{d(u)}$.
2. Picking one of the hyperedge e's nodes v with probability $\frac{\gamma_e(v)}{\delta(e)}$.
3. Moving to node v.

Variations include non-lazy walks, which never visit the same node twice in a row with a modified second step

2. Picking one of the hyperedge e's nodes $v \neq u$ with probability $\frac{\gamma_e(v)}{\delta(e)-\gamma_e(u)}$,

and teleporting walks, which jump to a random node at some rate to ensure that all nodes can be reached from any node in a finite number of moves, so-called ergodic walks. We pick the next hyperedge based on the similarity with the previously picked hyperedge in hyperedge-similarity walks, which are useful for modelling flows that tend to stay among similar hyperedges such as among research papers with similar author lists and likely similar topics. These walks require memory and correspond to a higher-order Markov chain model because they depend on the previously picked hyperedge.

The bipartite, unipartite, and multilayer network representations have different advantages and limitations (Fig. 1). A weighted, undirected network suffices for memoryless lazy random walks without hyperedge-dependent node weights, hyperedge-dependent node weights require directed networks, and hyperedge-similarity walks require multilayer networks.

Bipartite networks offer the most direct representation of the three-step random walk process above. We represent the hyperedges with hyperedge nodes, and the three steps become a two-step walk between the nodes at the bottom and the hyperedge nodes at the top in Fig. 1b. For simplicity, we refer to them as nodes and hyperedge nodes. First a step from a node u to a hyperedge node e,

$$P_{ue} = \frac{\omega(e)}{d(u)}, \tag{1}$$

and then a step from the hyperedge node to a node v,

$$P_{ev} = \frac{\gamma_e(v)}{\delta(e)}. \tag{2}$$

By starting the random walk on the nodes and taking two steps at a time, corresponding to Markov time two[28], hyperedge nodes are only intermediate stops with zero flow when the random walk is back on the nodes after two steps. The stationary distribution of the random walk is concentrated to the nodes. For non-lazy walks represented with bipartite networks, we use so-called state nodes[27] in the hyperedge nodes. One state node for each incoming link has out-links to all nodes in the hyperedge except the incoming link's source ensures that the walks are not backtracking (Fig. 2).

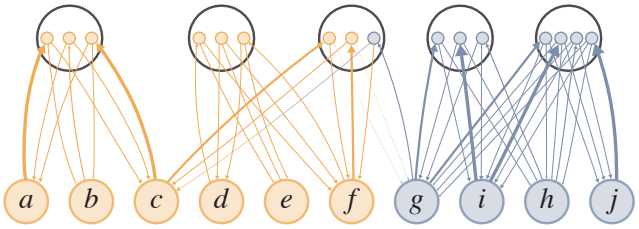

Fig. 2. Bipartite network with state nodes for non-lazy random walks. To prevent random walks on bipartite networks to visit the same node at the bottom twice in a row by backtracking from the hyperedge node at the top, we use state nodes in the hyperedge nodes. Each hyperedge node requires one state node for each node in the hyperedge. The state nodes have one incoming link from its source node and outgoing links to all other nodes in the hyperedge. Colours indicate the optimised partition in Fig. 3(b).

To represent the random walk on a unipartite network, we project the three-step random walk process down to a one-step process between the nodes and describe it with the transition rate matrix

$$P_{uv} = \sum_{e \in E(u,v)} P_{ue} P_{ev} = \sum_{e \in E(u,v)} \frac{\omega(e)}{d(u)} \frac{\gamma_e(v)}{\delta(e)}, \tag{3}$$

where $E(u,v) = \{e \in E : u \in e, v \in e\}$ is the set of hyperedges incident to both nodes u and v. Each hyperedge forms a fully connected group of nodes (Fig. 1c). Unipartite networks for non-lazy walks have no self-links. Compared with the bipartite representation, the unipartite representation with fully connected groups of nodes requires more links.

To represent the random walk on a multilayer network, we project the three-step random walk process down to a one-step process on state nodes in separate layers α for each hyperedge e. A state node u^α represents u in each layer $\alpha \in E(u)$ that contains the node. All state nodes in the same layer form a fully connected set (Fig. 1d). The transition rate between state node u^α in layer α and state node v^β in layer β is

$$P_{uv}^{\alpha\beta} = \frac{\omega(\beta)}{d(u)} \frac{\gamma_\beta(v)}{\delta(\beta)} \text{ for } \beta \in E(u,v). \tag{4}$$

Node u's state node visit rates in different layers sum to u's visit rate in the unipartite and bipartite representations. With one state node per hyperedge layer that contains the node, the multilayer representation requires the most nodes and links to describe the walk. But this cost comes with benefits. The multilayer representation can describe higher-order Markov chains, which can capture more regularities in the data.

For example, a useful variant of the basic hypergraph random walk is to pick a hyperedge not only proportional to its weight but also proportional to how similar it is to the hyperedge picked in the previous step. To include hyperedge-dependent node weight information in the similarity measure, we use one minus the Jensen-Shannon divergence (JSD) between the transition rate vectors $\mathbf{P}_{\alpha v}$ and $\mathbf{P}_{\beta v}$ to nodes at layers α and β as the hyperedge coupling strength,

$$\begin{aligned} D_u^{\alpha\beta} &= \omega(\beta)\left[1 - JSD(\alpha,\beta)\right] \\ &= \omega(\beta)\left[1 - H\left(\frac{1}{2}\mathbf{P}_{\alpha v} + \frac{1}{2}\mathbf{P}_{\beta v}\right)\right. \\ &\qquad \left. + \frac{1}{2}H\left(\mathbf{P}_{\alpha v}\right) + \frac{1}{2}H\left(\mathbf{P}_{\beta v}\right)\right] \end{aligned} \tag{5}$$

for $\beta \in E(u,v)$. With node u's total incident hyperedge weight in layer α

$$S_u^\alpha = \sum_{\beta \in E(u)} D_u^{\alpha\beta}, \tag{6}$$

the hyperedge-similarity walk has the transition rates

$$P_{uv}^{\alpha\beta} = \frac{D_u^{\alpha\beta}}{S_u^\alpha} \frac{\gamma_\beta(v)}{\delta(\beta)} \text{ for } \beta \in E(u,v). \tag{7}$$

Because the transition rates at a node depend on the current layer, the random walks generate non-Markovian dynamics that a unipartite or bipartite network representation cannot capture.

To ensure ergodic node-visit rates, we derived an unrecorded teleportation scheme that leaves the node-visit rates unchanged when teleportation is superfluous for hypergraphs with hyperedge-independent node weights, robust to changes in the teleportation rate when teleportation is needed[26], and independent of the representation (see Methods).

Mapping flows on hypergraphs. To identify flow-based communities or modules in hypergraphs, we seek to compress a modular description of random walks on the network representations guided by their links. We cast the problem of finding flow-based communities in hypergraphs as a minimum-description length problem with the map equation framework[3]. With this compression-based framework, we can compare how much the different representations compress modular flows.

When detecting communities, the representation matters because bipartite, unipartite, and multilayer networks provide the community-detection algorithm Infomap with different degrees of freedom[27]. Infomap assigns only nodes to communities in a unipartite network, but also hyperedge nodes in a bipartite network. The multilayer network, with a state node for each hyperedge a

Table I. Optimal flow-based communities of the schematic hypergraph in Fig. 1 represented with different networks. The number of nodes includes state nodes for the multilevel representations and the bipartite non-lazy representation. We measure the overlap as the perplexity of the optimal solutions (see Methods).

Representation	Nodes	Links	Modules	Codelength (bits)	Overlap
Lazy					
Bipartite	15	32	2	2.90	–
Unipartite	10	40	3	2.35	–
Multilayer	16	98	3	2.35	1.00
Multilayer h-s[a]	16	98	4	2.28	1.09
Non-lazy					
Bipartite	26	52	2	3.00	–
Unipartite	10	30	3	2.63	–
Multilayer	16	68	3	2.62	1.10
Multilayer h-s[a]	16	68	4	2.32	1.29

[a] hyperedge-similarity

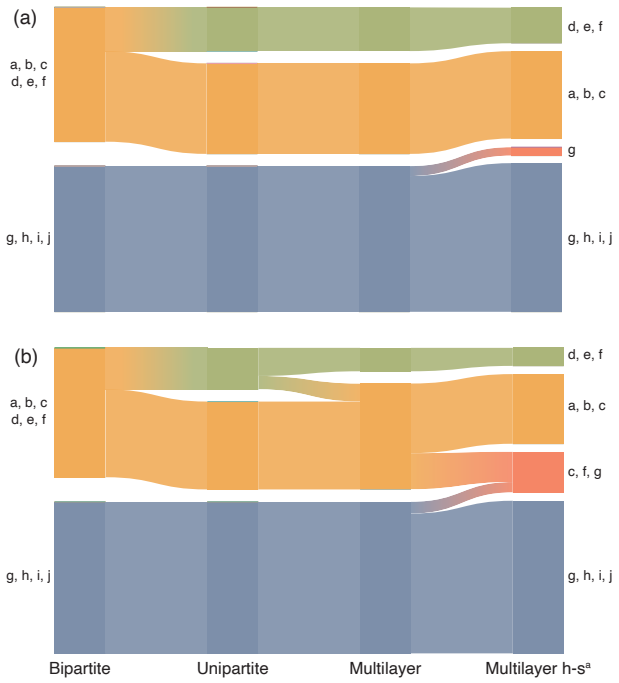

Fig. 3. Alluvial diagrams of optimal partitions for the schematic hypergraph in Fig. 1. (a) Optimal partitions for lazy walks represented with the networks in Fig. 1(b-d). (b) Optimal partitions for non-lazy walks.

node belongs to, implies even more node assignments and possibly overlapping communities.

When mapping flows modelled by lazy and non-lazy random walks on the schematic network in Fig. 1, the optimal partitions of the bipartite networks have two communities, whereas the unipartite and multilayer networks have three communities (Table I and Fig. 3). The bipartite network favours fewer modules – using the optimal three-module partition of the unipartite network on the bipartite network gives code length 3.29 bits instead of 2.90 bits for two modules — because the random walker transitions more frequently between modules when they include hyperedges: Even if a hyperedge node contains no flows at the end of each two-step walk from node through hyperedge node to node, assigning it to a module costs extra bits when it has nodes in multiple modules. For example, if nodes a, b, and c in the bipartite network in Fig. 1(b) would belong to a third green module as in the optimal unipartite solution, and the random walker at node c would return to the hyperedge it comes from before revisiting node c, it would first need to exit the green module and enter the orange module, then exit the orange module and enter the green module. The corresponding walk on the unipartite network stays within the green module. As a result, the unipartite network representation favours more, smaller modules than the bipartite network representation for lazy and non-lazy walks (Table I).

Multilayer networks enable further compression with overlapping modules. But for this small network, only non-lazy walks give overlapping modules with 0.01 bits compression gain (Table I). With walks that preferentially move to similar hyperedges, the optimal partitions of the multilayer hyperedge-similarity network representations for lazy and non-lazy random walks both have more overlap in four modules (Table I and Fig. 3). The hyperedge-similarity walks favour these overlapping modules because they stay longer within them than the regular walks.

For a given random-walk model, the representations give equivalent node-visit rates but alter the link flows. And with different link flows, the optimal partition can change. The bipartite network representation favours partitions with fewer modules than the unipartite network representation because hyperedge nodes assigned to modules implies encoding more transitions between modules. Multilayer representations, especially with walks that spend longer time among similar hyperedges, favour more overlapping modules. The random-walk model determines how much the multilayer network modules overlap. Non-lazy and hyper-edge similarity walks favour overlap because they lead to longer persistence times among nodes in possibly overlapping groups.

Experiments. To illustrate how the network representation affects detected communities in real hypergraphs, we generated a collaboration hypergraph from the 734 references in *Networks beyond pairwise interactions: Structure and dynamics* by F. Battiston et al.[8] We modelled the referenced articles as hyperedges and their authors as nodes. Authors with multiple articles form connections between the hyperedges. We analysed the largest connected component with $|V| = 361$ author nodes in $|E| = 220$ hyperedges. The median number of authors in a hyperedge is 3, and the authors have contributed to 2.2 articles on average though most have only contributed to one.

We assigned the relative importance of references by their number of citations c in December 2020. Some references had no citations and some were highly cited. One such example is *Diffusion of innovations* by Everett M. Rogers, with more than 120, 000 citations. To avoid disproportionally large or small hyperedge weights $\omega(e)$, we weighed the edges by the logarithm of the number of citations and added unit constants to avoid the zero citation

Table II. Optimised flow-based multilevel communities of the collaboration hypergraph represented with different networks. The number of nodes includes state nodes for the multilevel representations and the bipartite non-lazy representation. Shortest codelength of 100 trials with the variance in parenthesis. We measure the overlap as the perplexity of the optimised solutions (see Methods).

Representation	Nodes	Links	Modules				Codelength
			Top	Leaf	Levels	Overlap	(bits)
Lazy							
Bipartite	581	1,560	4	23	3	–	5.178(1)
Unipartite	361	2,607	9	69	4	–	3.82557(2)
Multilayer	780	17,193	9	76	4	1.003	3.82730(2)
Multilayer h-s[a]	780	17,193	8	90	4	1.127	3.54939(3)
Non-lazy							
Bipartite	1,141	3,548	5	25	3	–	5.1733(2)
Unipartite	361	2,246	7	49	4	–	4.25104(8)
Multilayer	780	12,843	7	54	4	1.098	4.16349(8)
Multilayer h-s[a]	780	12,843	9	66	4	1.181	3.70432(1)

[a] hyperedge-similarity

problem,

$$\omega(e) = \ln(c+1) + 1. \tag{8}$$

We modelled the authors' different contributions to articles by assigning higher weights to the first and last author[20]. We used the edge-dependent node weights

$$\gamma_e(v) = \begin{cases} 2 & \text{if node } v \text{ is first or last author,} \\ 1 & \text{otherwise.} \end{cases} \tag{9}$$

We assumed equal contribution for alphabetically sorted authors, and assigned all of them weight $\gamma(v) = 1$. This model ranks co-corresponding author's contributions lower than corresponding authors.

To study how hypergraph representations and random walk models affect the community structure, we generated bipartite, unipartite, and multilayer representations for lazy and non-lazy random walks on the collaboration network. We identified nested hierarchical partitions in each network with Infomap, using 100 independent searches for each network. Infomap's running time depends on the number of nodes, links, and solution levels: The bipartite and unipartite representations finished 3–7 times faster than the multilayer representations. The non-lazy bipartite representation with many state nodes ran almost as long.

The optimised partitions for the lazy and non-lazy representations behave like the schematic example: The bipartite representations have the fewest leaf modules and highest codelengths, and the multilayer hyperedge-similarity representations have the most leaf modules and shortest codelengths with the unipartite and the regular multilayer representations in between (Table II). Except for the non-lazy bipartite representation with its many state nodes, the lazy representations have more leaf modules and shorter code lengths than their corresponding non-lazy representations because the lazy random walk is more confined than the non-lazy random walk.

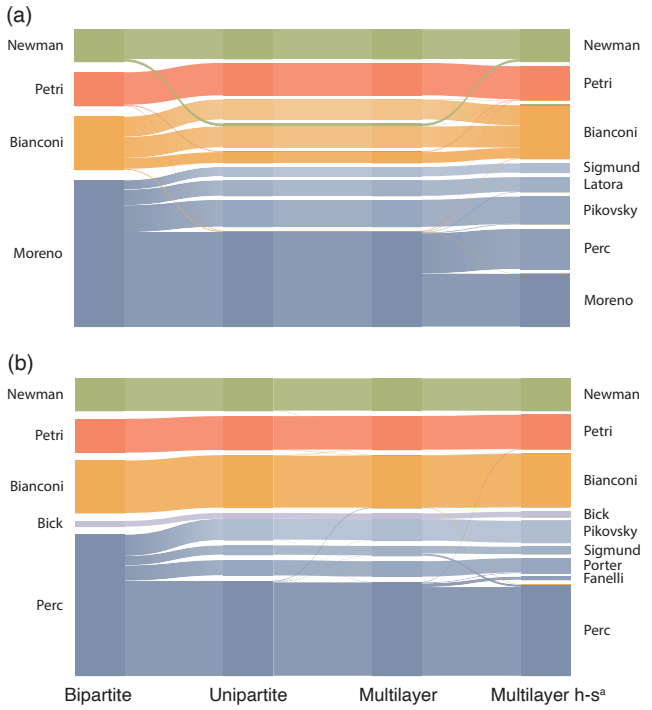

Fig. 4. Alluvial diagrams of optimised partitions for different representations of the collaboration hypergraph . Lazy walks in (a) and non-lazy walks in (b). Module names from the top-ranked author within each module.

With more nodes than in the schematic example, the solutions have more depth. The bipartite solutions have three, and the unipartite and multilayer solutions have four hierarchical levels. The unipartite and multilayer solutions also have more top modules. With non-lazy dynamics, they split the largest top module, and in the lazy dynamics, they split the two largest top modules. But the second-largest top module reunites in the hyperedge-similarity representation with stronger connections between similar hyperedges (Fig. 4 and Fig. 7 in Appendix A). The unipartite and multilayer solutions are also most similar at the leaf level (Fig. 8 in Appendix A).

In this larger example, the multilayer hyperedge-similarity representations give more overlap. The non-lazy representations result in higher average overlap because random walkers visiting a node must continue to other nodes, often in the same or a similar hyperedge layer. When random walkers from dissimilar hyperedges come together at a node, they tend to return to where they came from and favour overlapping modules. The non-lazy representations also result in higher max overlap with the same authors topping all representations (Fig. 5).

In line with the information-theoretic duality between finding regularities in data and compressing those data, representations that enable deeper solutions with more modules have shorter codelengths (Table II). The lazy multilayer representation is an exception. Its optimised codelength is bound above by the lazy unipartite representation's codelength – they have the same codelength for the same hard partition – and overlapping modules can potentially reduce the codelength. Infomap's best codelength was

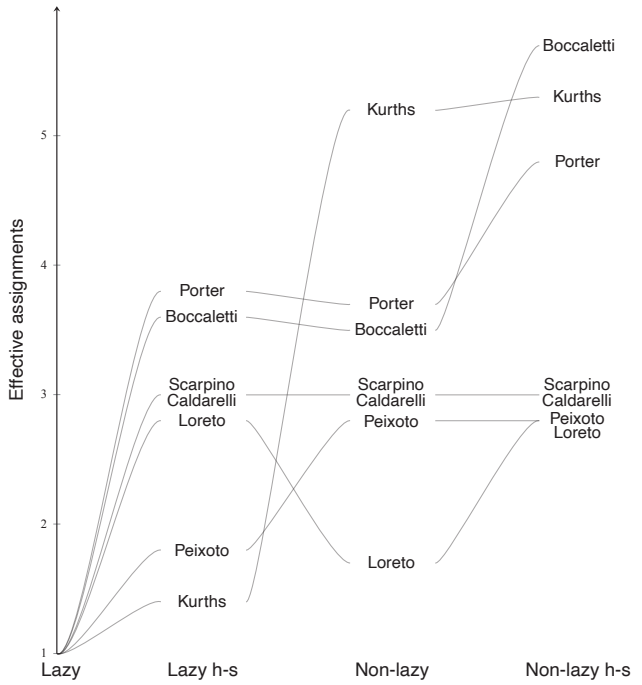

Fig. 5. Authors in the collaboration hypergraph with the highest average effective number of assignments in the lazy and non-lazy multilayer representations (see Methods).

instead 0.05 percent longer than for the lazy unipartite representation. Multilayer representations with their many state nodes and links aggravate the search problem, and Infomap could not find a better solution with 100 attempts. But the gain from overlapping modules is higher for the non-lazy multilayer representation and Infomap finds a solution with a significantly shorter codelength.

A case study on fossil data. Palaeontologists classify major groups of marine animals archived in the fossil record into global-scale faunas that change over time[29]. They have used different network representations to understand the macroevolutionary pattern of marine biodiversity[30,31]. However, it is still unclear how such an organisation of marine animals into modules representing global faunas changes with random-walk model and network representation. To illustrate how the network representation of the underlying paleontological data affects empirical estimates of this macroevolutionary pattern, we generated a hypergraph from genus-level fossil occurrences presented in ref. 30 and retrieved from the PaleoDB[32]. We restricted our analysis to fossil occurrences from the Cambrian (541 MY) to the Cretaceous period (66 MY) and modelled 77 geological stages as hyperedges and 13,276 genera as nodes. Genera occurring in multiple geological stages form connections between hyperedges. We weighted the hyperedges by dividing the number of samples where a genus occurs in a given geological stage by the total number of samples recorded at the stage, a procedure modified from ref. 33. We generated bipartite, unipartite, and multilayer network representations for lazy and non-lazy random walks from the underlying palaeontology data and identified optimised partitions in the assembled networks using Infomap.

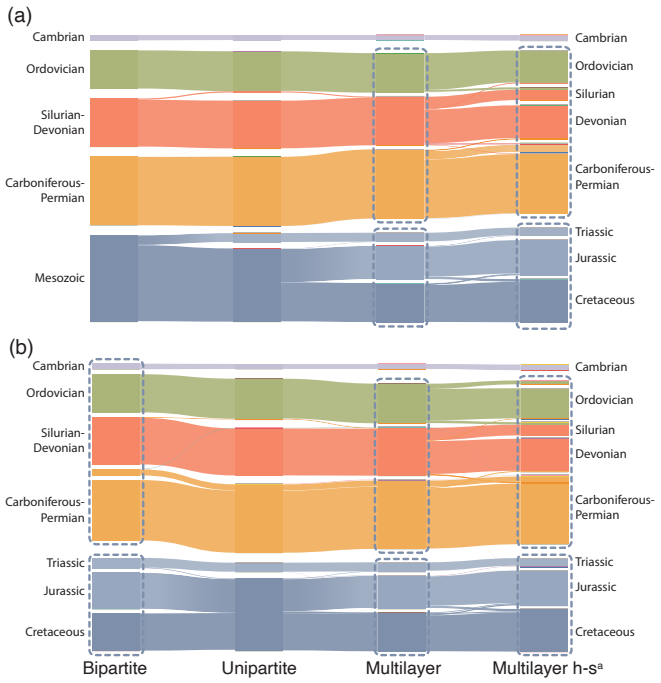

Fig. 6. Alluvial diagrams of optimised partitions for the fossil hypergraph represented with different networks. Lazy walks in (a) and non-lazy walks in (b). We show top modules when a partition lacks deeper levels and leaf modules marked with dashed lines when they exist. Module names from the geological period or era represented by the fauna assemblage.

For lazy random walks, Infomap partitioned only the multilayer representations into multilevel communities: three modules at the first hierarchical level [Fig. 6(a)]. Similar to the schematic example and the collaboration hypergraph, the bipartite representation for the lazy random walks has the fewest leaf modules and the highest codelength. The multilayer hyperedge-similarity representation has the most leaf modules and the shortest codelength (Table III).

For non-lazy random walks, Infomap partitioned the bipartite representation into a multilevel solution with shorter codelength than the unipartite representation and the standard multilevel representation [Fig. 6(b)]. The multilayer hyperedge-similarity representation once more provides the most leaf modules and highest overlap.

The multilayer network representations, including lazy and non-lazy random walks, reproduce modules reminiscent of the Cambrian, Paleozoic, and modern evolutionary faunas widely used in macroevolutionary research[29]. Also, leaf modules in the multilayer representations capture subfaunas from specific geological periods as nested modules such as Silurian, Triassic, Jurassic, and Cretaceous. Infomap applied to the bipartite representation of the non-lazy random walks identified similar subfaunas but combined Cambrian and Paleozoic faunas into a single top module, obscuring the large-scale pattern. Overall, our results indicate some advantages of using multilayer over bipartite and unipartite representations of fossil occurrence data to quantify the marine biodiversity's macroevolutionary patterns, with lazy and non-lazy random walks providing similar solutions.

Table III. Optimised flow-based multilevel communities of the fossil hypergraph represented with different networks. The number of nodes includes state nodes for the multilevel representations and the bipartite non-lazy representation. The number of non-trivial top and leaf modules. Average number of levels weighted by the flow volume. We measure the overlap as the perplexity of the optimised solutions (see Methods). Shortest codelength of 20 trials with the variance in parenthesis.

Representation	Nodes ($\times 10^3$)	Links ($\times 10^3$)	Modules Top	Leaf	Levels	Overlap	Codelength (bits)	Time (hh:mm:ss)
Lazy								
Bipartite	13	79	5	8	2.02	–	10.50927(5)	00:00:06
Unipartite	13	16,155	6	13	2.02	–	10.3953503(1)	00:13:24
Multilayer	40	174,490	3	17	3.00	1.011	10.39819(1)	09:08:43
Multilayer h-s[a]	40	174,490	3	19	3.28	1.135	9.84170(1)	14:19:39
Non-lazy								
Bipartite	53	25,937	2	15	3.02	–	10.34889(3)	01:14:25
Unipartite	13	16,141	6	12	2.02	–	10.4031798(6)	00:13:04
Multilayer	40	174,209	3	15	3.00	1.010	10.406141(9)	08:55:03
Multilayer h-s[a]	40	174,209	3	16	3.00	1.135	9.84912(1)	13:23:13

[a] hyperedge-similarity

Conclusions

We have derived unipartite, bipartite, and multilayer network representations of hypergraph flows with different advantages. We used the information-theoretic and flow-based community detection method Infomap to explore how different hypergraph random walk models and network representation change the number, size, depth, and overlap of identified multilevel communities. By identifying flow-based communities in a schematic and real hypergraphs – a small collaboration hypergraph of researchers working on networks beyond pairwise interactions and a large faunal hypergraph of sampled species across geological stages – we found that the bipartite network representation is most compact and enables the fastest community detection. A multilayer network representation that reinforces flows within similar layers – one for each hyperedge – gave the deepest modular structures with most module overlap. But the modular detection gain comes at a high computational cost: Combining fully connected layers with other layers requires many more nodes and links than the bipartite network representation. If the research question does not require hyperedge assignments or overlapping modules, the unipartite network representation provides a trade-off with intermediate compactness, speed, and ability to reveal modular regularities. Among the random-walk models, lazy walks typically give more modules in deeper nested structures, and non-lazy walks higher modular overlap. Our methods and results help researchers model and map flows on hypergraphs to study the effects of multibody interactions in complex systems.

Methods

Unrecorded teleportation. With hyperedge-independent node weights where $\gamma_e(u) = \gamma(u)$ for all hyperedges $e \in E(u)$, undirected weighted networks can represent the dynamics, and the stationary distribution of the random walk π_u is proportional to the product of node u's total incident hyperedge weight $d(u)$ and weight $\gamma(u)$. With normalised node-visit rates[20],

$$\pi_u = \frac{d(u)\gamma(u)}{\sum_{v \in V} d(v)\gamma(v)}. \tag{10}$$

For the multilayer network representation, the node-visit rates split between layers based on the node u's incident hyperedge weight per layer state node

$$\pi_u^\alpha = \frac{\omega(\alpha)\gamma(u)}{\sum_{v \in V} d(v)\gamma(v)}. \tag{11}$$

With hyperedge-dependent node weights $\gamma_e(u)$, only directed weighted networks can represent the dynamics. We use random teleportation to ensure ergodic walks when deriving the node-visit rates with the power-iteration method. Unrecorded teleportation to links minimises the distortion[26]: In each iteration of the power-iteration method, we distribute a fraction $\tau = 0.15$ of each node's flow volume among all nodes proportional to their out-link weights. The remaining flow volume moves on the links proportional to their weights. In the last iteration, we move all flows on the links proportional to their weights and record all flows on links and nodes to obtain the ergodic node- and link-visit rates with unrecorded teleportation. This procedure gives equivalent visit rates as simulating a random walker that only records moves on links: With probability $1 - \tau$, the random walker moves to a node by following the links proportional to their weights and records the link and the target node. With probability τ, the random walker teleports without recording to the link's source node proportional to the link weight. The normalised number of recordings of each node and link gives the visit rates.

We want teleportation applied to undirected networks – where it is unnecessary – to leave the node- and link-visit rates unchanged. We achieve this smooth teleportation by scaling the transition rates from nodes by the node-visit rates: Then unrecorded teleportation proportional to nodes' total out-link weights followed by recorded moves on the links proportional to their weights distribute on the

nodes according to the ergodic visit rates on undirected networks[26]. For the general case when the node weights can depend on the hyperedge, and the network may be directed, we use Eq. 10 without assuming $\gamma_e(u) = \gamma(u)$ as an approximation of the node-visit rates:

$$\tilde{\pi}_u = \frac{\sum_{e \in E(u)} \omega(e)\gamma_e(u)}{\sum_{v \in V, e \in E(v)} \omega(e)\gamma_e(v)} \tag{12}$$

for nodes and

$$\tilde{\pi}_u^{\alpha} = \frac{\omega(\alpha)\gamma_\alpha(u)}{\sum_{v \in V, e \in E(v)} \omega(e)\gamma_e(v)} \text{ for } \alpha \in E(u) \tag{13}$$

for state nodes. With exact node-visit rates, we would obtain the stationary flow volumes on links by multiplying the transition rates by the source nodes' visit rates. With approximate node-visit rates, instead, we obtain the link weights

$$w_{ue} = \tilde{\pi}_u P_{ue} \tag{14}$$

for bipartite networks,

$$w_{uv} = \tilde{\pi}_u P_{uv} \tag{15}$$

for unipartite networks, and

$$w_{uv}^{\alpha\beta} = \tilde{\pi}_u^{\alpha} P_{uv}^{\alpha\beta} \text{ for } \beta \in E(u, v) \tag{16}$$

for multilayer networks. With unrecorded teleportation proportional to these link weights, modelling flows on hypergraphs give node-visit rates robust to changes in the teleportation rate and independent of the representation.

Overlap metric. Modules overlap when Infomap assigns a node's state nodes in the multilayer network representations to different modules. Measuring the overlap through the absolute number of assignments is misleading because the overlap is 2 regardless of the number of state nodes assigned to a different module than the rest. Instead, we used the effective number of assignments. If a fraction f of node u's state nodes is assigned to the mth module in u's module assignment set, the mth element of u's assignment vector is $a_m^u = f$ and the effective number of assignments measured by the perplexity of u's module assignments is

$$o_u = 2^{H(\mathbf{a}^u)}. \tag{17}$$

The effective number of assignments is one if all u's state nodes are in one module, and it is equal to the number of assignments when the state nodes are divided evenly among u's module assignments. We averaged over all nodes for the partition overlap.

Data and code availability

All data and source code are available on GitHub: `http://github.com/mapequation/mapping-hypergraphs`.

References

1. Brin, S. & Page, L. The anatomy of a large-scale hypertextual web search engine. *Comput. Networks ISDN* **30**, 107–117 (1998).
2. Simonsen, I., Eriksen, K. A., Maslov, S. & Sneppen, K. Diffusion on complex networks: a way to probe their large-scale topological structures. *Physica A* **336**, 163–173 (2004).
3. Rosvall, M. & Bergstrom, C. T. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences* **105**, 1118–1123 (2008).
4. Delvenne, J., Yaliraki, S. & Barahona, M. Stability of graph communities across time scales. *Proc. Natl. Acad. Sci. USA* **107**, 12755–12760 (2010).
5. Boccaletti, S., Latora, V., Moreno, Y., Chavez, M. & Hwang, D.-U. Complex networks: Structure and dynamics. *Physics reports* **424**, 175–308 (2006).
6. Fortunato, S. Community detection in graphs. *Physics reports* **486**, 75–174 (2010).
7. Lambiotte, R., Rosvall, M. & Scholtes, I. From networks to optimal higher-order models of complex systems. *Nature physics* **15**, 313–320 (2019).
8. Battiston, F. *et al.* Networks beyond pairwise interactions: structure and dynamics. *Physics Reports* (2020).
9. Mucha, P. J., Richardson, T., Macon, K., Porter, M. A. & Onnela, J.-P. Community structure in time-dependent, multiscale, and multiplex networks. *science* **328**, 876–878 (2010).
10. Kivelä, M. *et al.* Multilayer networks. *Journal of complex networks* **2**, 203–271 (2014).
11. De Domenico, M., Granell, C., Porter, M. A. & Arenas, A. The physics of spreading processes in multilayer networks. *Nature Physics* **12**, 901–906 (2016).
12. Rosvall, M., Esquivel, A. V., Lancichinetti, A., West, J. D. & Lambiotte, R. Memory in network flows and its effects on spreading dynamics and community detection. *Nature communications* **5**, 1–13 (2014).
13. Scholtes, I. *et al.* Causality-driven slow-down and speed-up of diffusion in non-markovian temporal networks. *Nature communications* **5**, 1–9 (2014).
14. Xu, J., Wickramarathne, T. L. & Chawla, N. V. Representing higher-order dependencies in networks. *Science Advances* **2**, e1600028 (2016).
15. Parzanchevski, O. & Rosenthal, R. Simplicial complexes: spectrum, homology and random walks. *Random Structures & Algorithms* **50**, 225–261 (2017).
16. Salnikov, V., Cassese, D. & Lambiotte, R. Simplicial complexes and complex systems. *European Journal of Physics* **40**, 014001 (2018).
17. Iacopini, I., Petri, G., Barrat, A. & Latora, V. Simplicial models of social contagion. *Nature communications* **10**, 1–9 (2019).
18. Schaub, M. T., Benson, A. R., Horn, P., Lippner, G. & Jadbabaie, A. Random walks on simplicial complexes and the normalized hodge 1-laplacian. *SIAM Review* **62**, 353–391 (2020).
19. Zhou, D., Huang, J. & Schölkopf, B. Learning with hypergraphs: Clustering, classification, and embedding. In *Advances in neural information processing systems*, 1601–1608 (2007).
20. Chitra, U. & Raphael, B. J. Random walks on hypergraphs with edge-dependent vertex weights. In *36th International Conference on Machine Learning, ICML 2019*, 2002–2011 (International Machine Learning Society (IMLS), 2019).
21. Carletti, T., Battiston, F., Cencetti, G. & Fanelli, D. Random walks on hypergraphs. *Physical Review E* **101**, 022308 (2020).
22. Carletti, T., Fanelli, D. & Lambiotte, R. Random walks and community detection in hypergraphs. *arXiv:2010.14355* (2020).
23. De Domenico, M., Lancichinetti, A., Arenas, A. & Rosvall, M. Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems. *Physical Review*

X **5**, 011027 (2015).
24. Jeub, L. G., Mahoney, M. W., Mucha, P. J., Porter, M. A. *et al.* A local perspective on community structure in multilayer networks. *Network Science* **5**, 144–163 (2017).
25. Alon, N., Benjamini, I., Lubetzky, E. & Sodin, S. Non-backtracking random walks mix faster. *Communications in Contemporary Mathematics* **9**, 585–603 (2007).
26. Lambiotte, R. & Rosvall, M. Ranking and clustering of nodes in networks with smart teleportation. *Phys. Rev. E* **85**, 056107 (2012).
27. Edler, D., Bohlin, L. *et al.* Mapping higher-order network flows in memory and multilayer networks with infomap. *Algorithms* **10**, 112 (2017).
28. Kheirkhahzadeh, M., Lancichinetti, A. & Rosvall, M. Efficient community detection of network flows for varying markov times and bipartite networks. *Physical Review E* **93**, 032309 (2016).
29. Sepkoski, J. J. A factor analytic description of the Phanerozoic marine fossil record. *Paleobiology* **7**, 36–53 (1981). URL https://www.cambridge.org/core/product/identifier/S0094837300003778/type/journal_article.
30. Rojas, A., Calatayud, J., Kowalewski, M., Neuman, M. & Rosvall, M. A multiscale view of the phanerozoic fossil record reveals the three major biotic transitions. preprint, Paleontology (2019). URL http://biorxiv.org/lookup/doi/10.1101/866186.
31. Muscente, A. D. *et al.* Quantifying ecological impacts of mass extinctions with network analysis of fossil communities. *Proceedings of the National Academy of Sciences* **115**, 5217–5222 (2018). URL http://www.pnas.org/lookup/doi/10.1073/pnas.1719976115.
32. Peters, S. E. & McClennen, M. The Paleobiology Database application programming interface. *Paleobiology* **42**, 1–7 (2016). URL http://www.journals.cambridge.org/abstract_S0094837315000391.
33. Rojas, A., Patarroyo, P., Mao, L., Bengtson, P. & Kowalewski, M. Global biogeography of Albian ammonoids: A network-based approach. *Geology* **45**, 659–662 (2017). URL https://pubs.geoscienceworld.org/geology/article/45/7/659-662/207876.

Acknowledgments

We thank Christopher Blöcker, Manlio De Domenico, Michael Schaub, and Jelena Smiljanić for valuable comments that helped us improve the manuscript. A.E was supported by the Swedish Foundation for Strategic Research, Grant No. SB16-0089. A.R., D.E. and M.R. were supported by the Swedish Research Council, Grant No. 2016-00796.

Author contributions

A.E. and M.R. conceived the study. A.E., A.R. and D.E. performed the numerical experiments and analysed the results. A.E. and M.R. wrote the manuscript.

Competing interests

The authors declare no competing interests.

A. Appendix

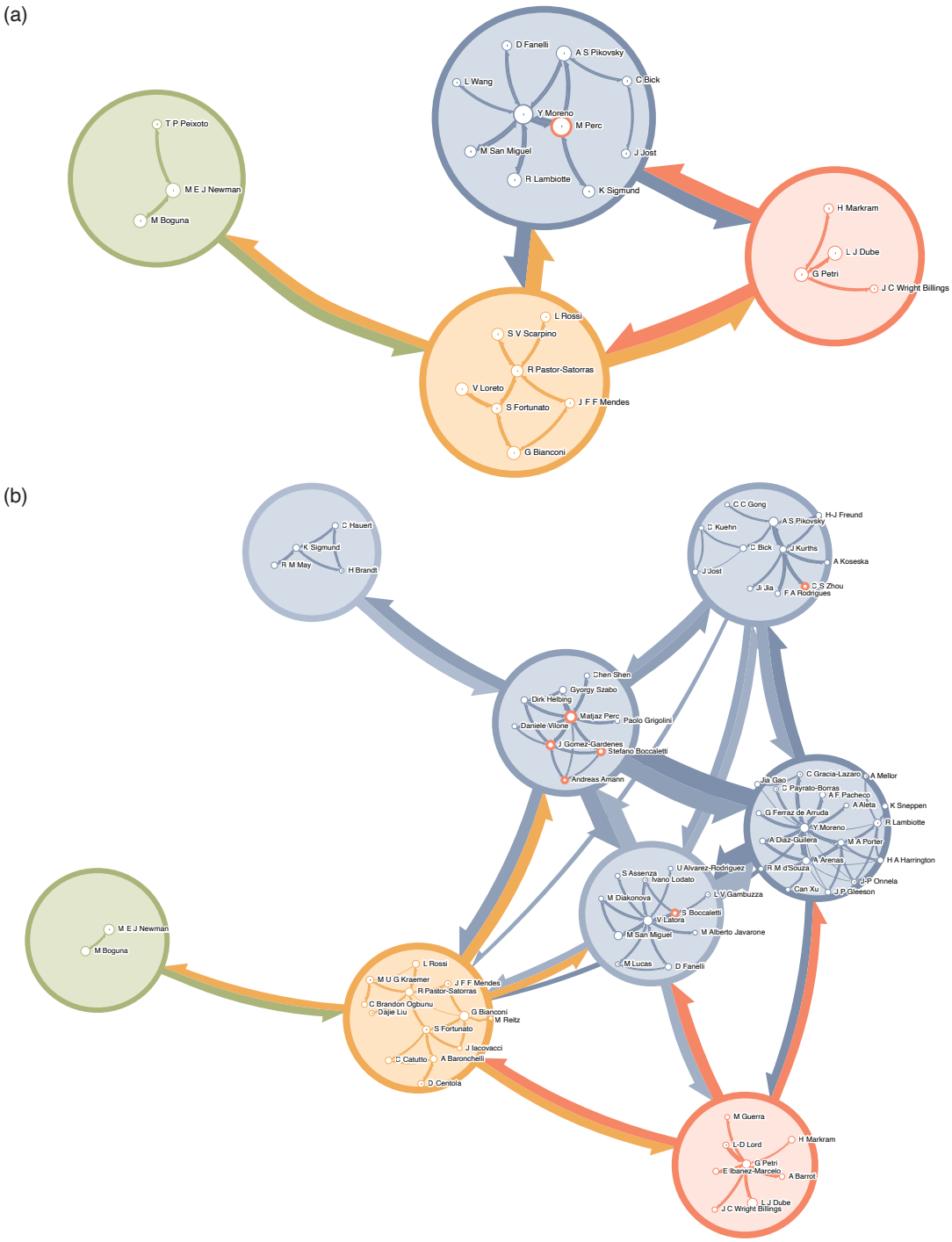

Fig. 7. Hierarchical maps of the collaboration hypergraph using (a) the bipartite representation and (b) the multilayer hyperedge-similarity representation. Module colours are the same as in Fig. 4(a). Aggregated inter-module links with sizes proportional to the exiting flow volume and length inversely proportional to the flow volume. White sub-modules are labelled with the top-ranked author. The largest blue top module in (a) contains ten sub-modules. In (b), the partition assigns those nodes to five top modules containing more sub-modules. S. Boccaletti, one of the most overlapping authors and highlighted in red, is assigned to one module in (a) and three top modules and six sub-modules in (b).

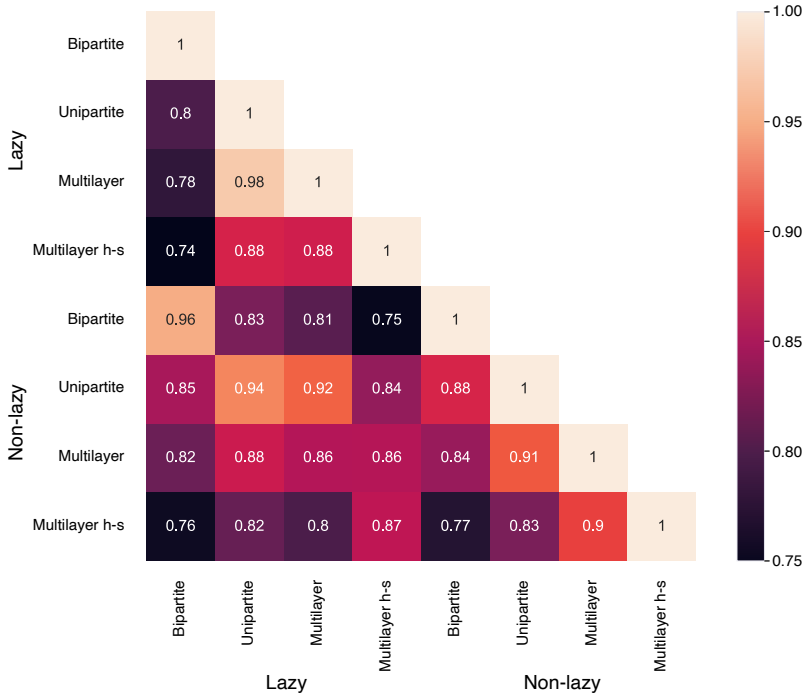

Fig. 8. Leaf module assignments' adjusted mutual information for different random walk dynamics and hypergraph representations. The bipartite representations differ the most from the other representations, and the unipartite and multilayer representations are most similar at the leaf level.

Figures

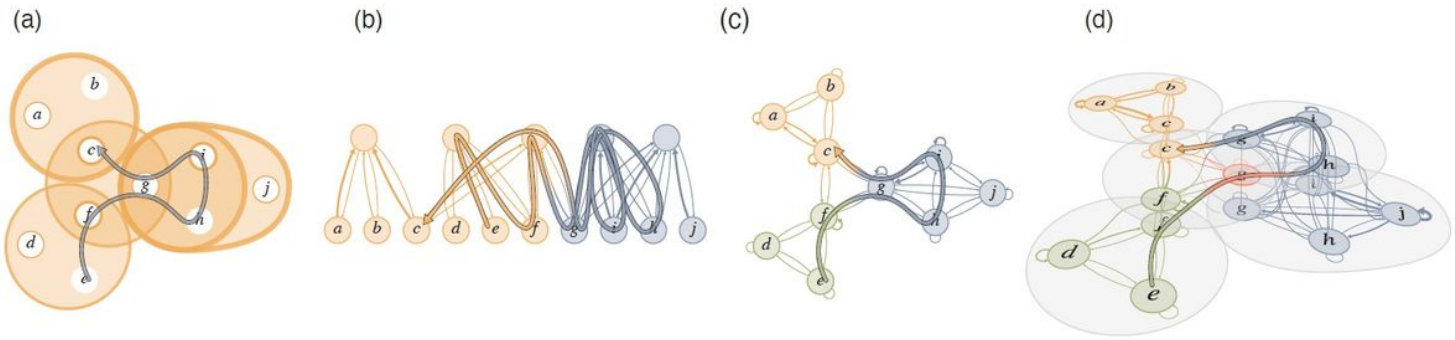

Figure 1

A schematic hypergraph represented with three types of networks. (a) The schematic hypergraph with weighted hyperedges and hyperedgedependent node weights. Thin borders for weight 1 and thick borders for weight 3. A lazy random walk on the schematic hypergraph represented on: (b) a bipartite network, (c) a unipartite network, and (d) a multilevel network. The colours indicate optimised module assignments, in (d) for hyperedge-similarity walks.

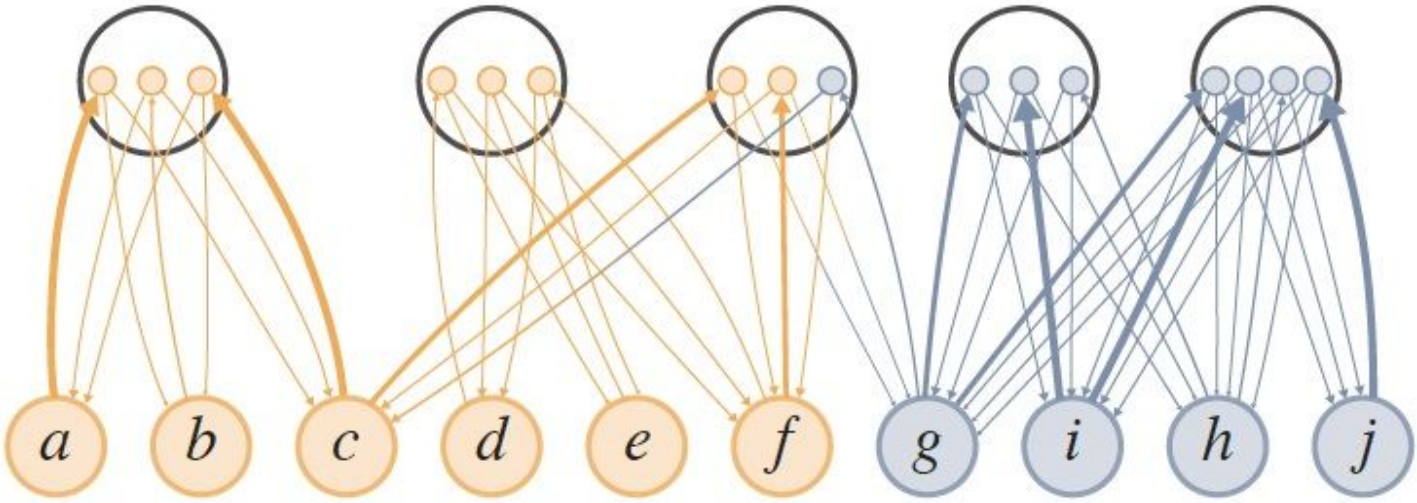

Figure 2

Bipartite network with state nodes for non-lazy random walks. To prevent random walks on bipartite networks to visit the same node at the bottom twice in a row by backtracking from the hyperedge node at the top, we use state nodes in the hyperedge nodes. Each hyperedge node requires one state node for each node in the hyperedge. The state nodes have one incoming link from its source node and outgoing links to all other nodes in the hyperedge. Colours indicate the optimised partition in Fig. 3(b).

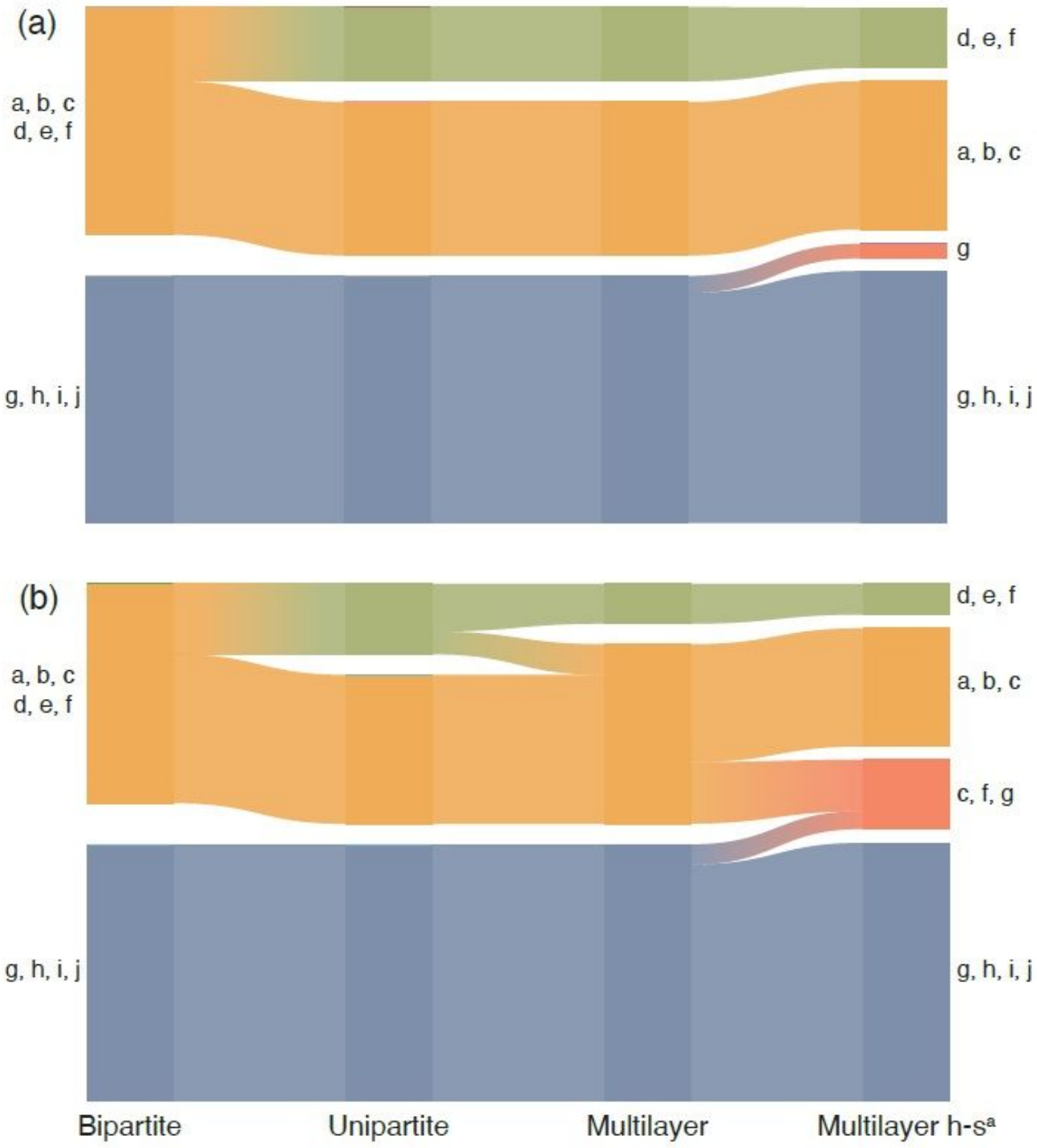

Figure 3

Alluvial diagrams of optimal partitions for the schematic hypergraph in Fig. 1. (a)Optimal partitions for lazy walks represented with the networks in Fig. 1(b-d). (b) Optimal partitions for non-lazy walks.

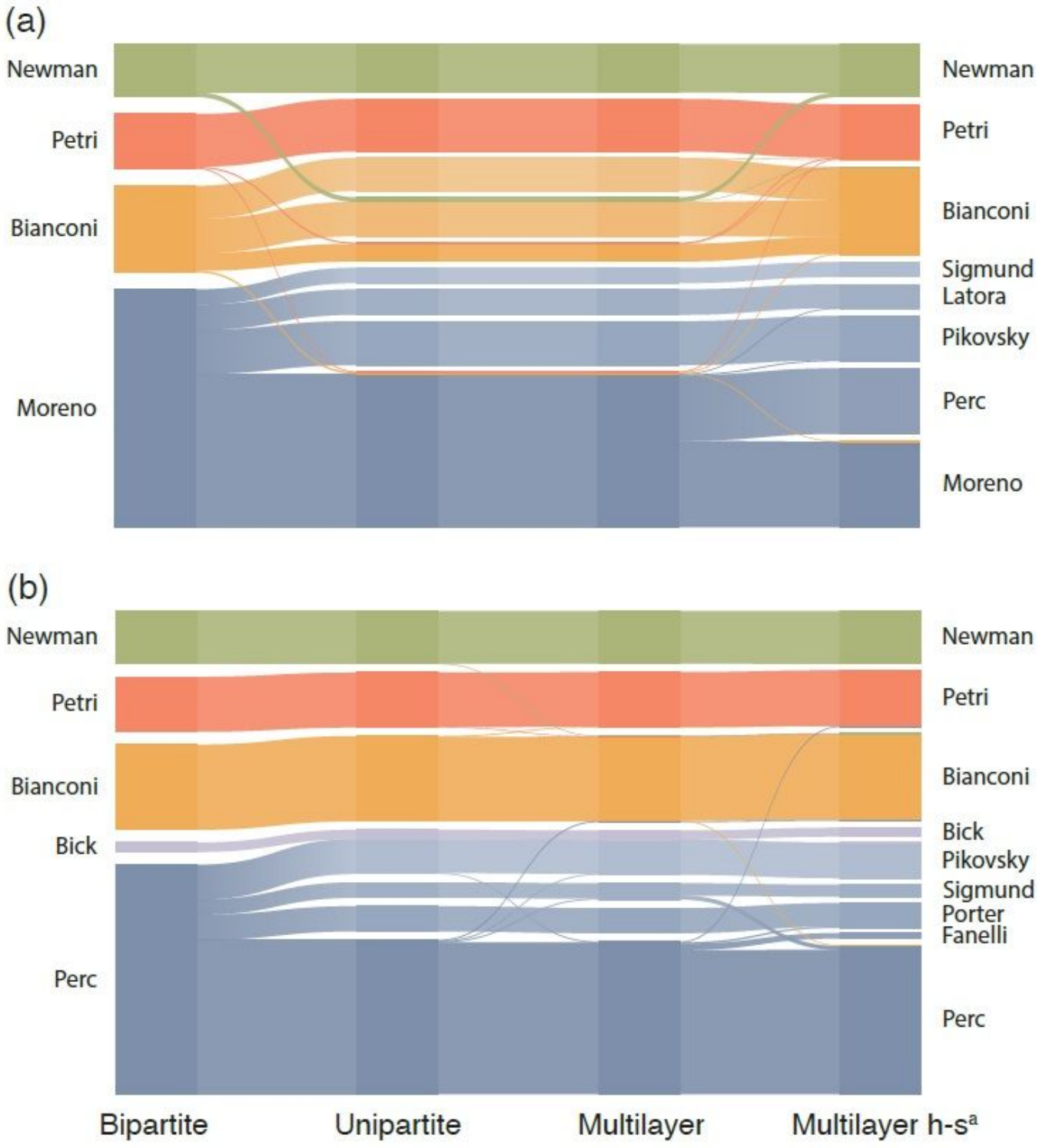

Figure 4

Alluvial diagrams of optimised partitions for different representations of the collaboration hypergraph . Lazywalks in (a) and non-lazywalks in (b). Module names from the top-ranked author within each module.

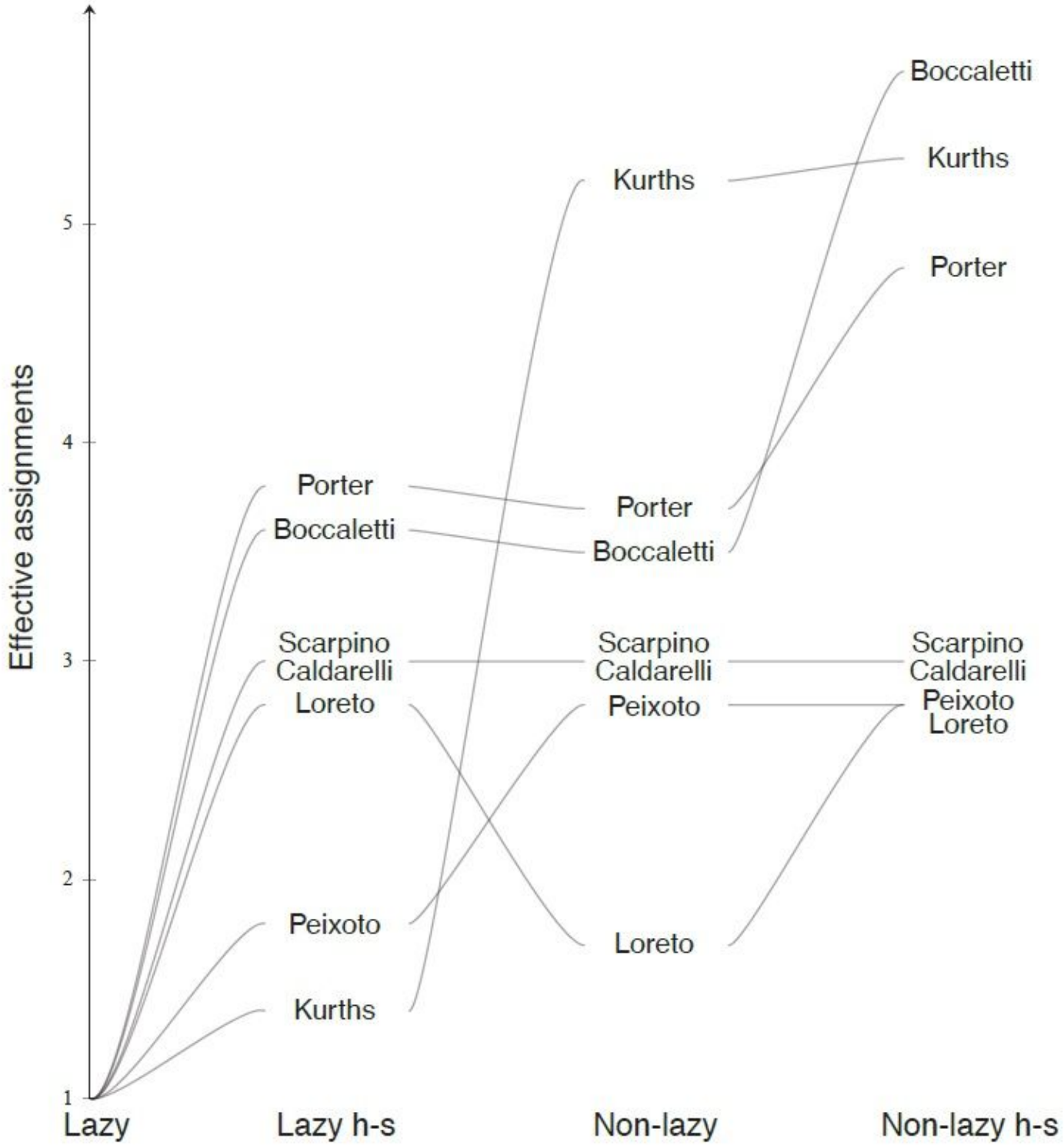

Figure 5

Authors in the collaboration hypergraph with the highest average effective number of assignments in the lazy and non-lazy multilayer representations (see Methods).

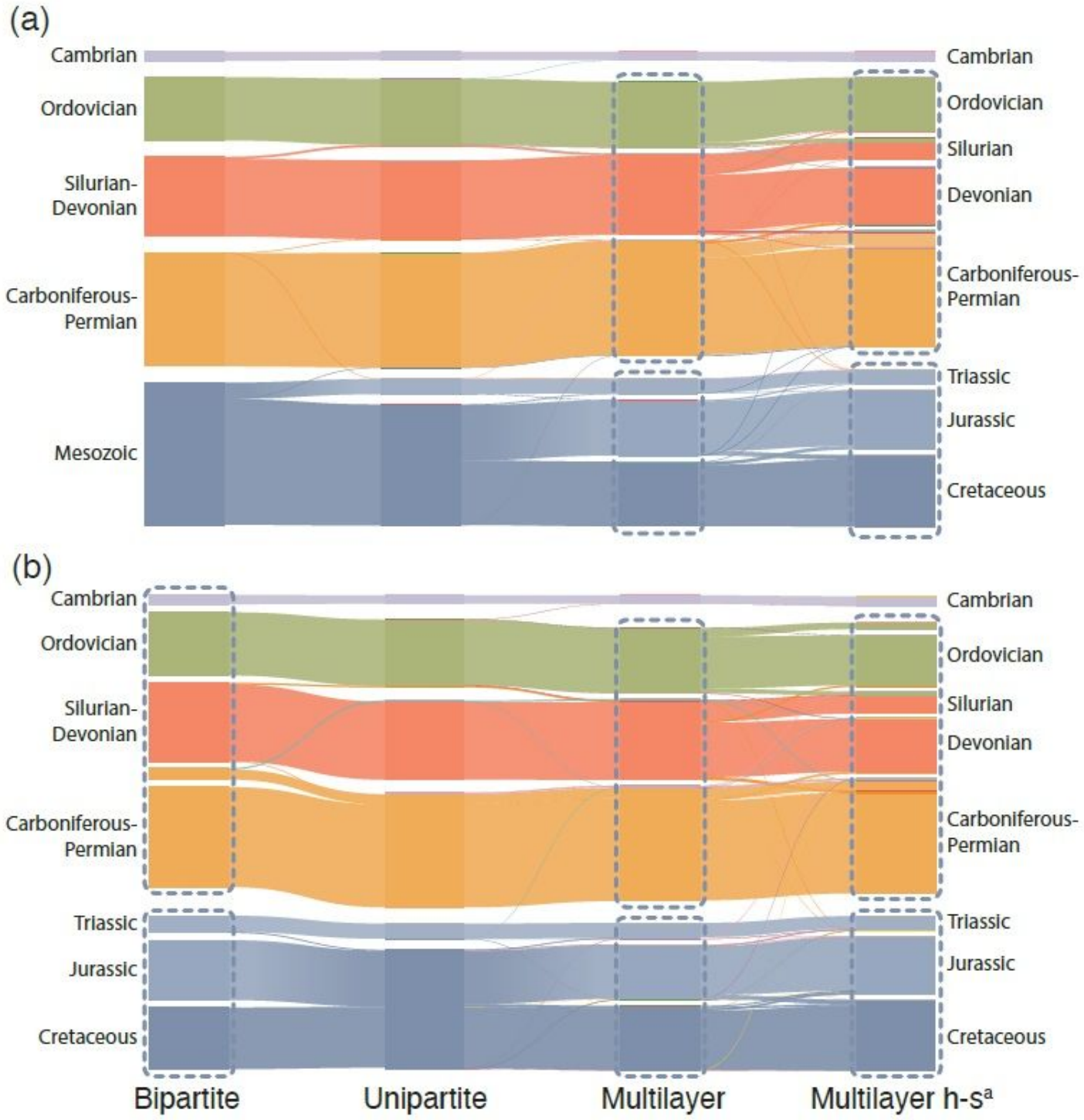

Figure 6

Alluvial diagrams of optimised partitions for the fossil hypergraph represented with different networks. Lazy walks in (a) and non-lazy walks in (b).We show top modules when a partition lacks deeper levels and leaf modules marked with dashed lines when they exist. Module names from the geological period or era represented by the fauna assemblage.

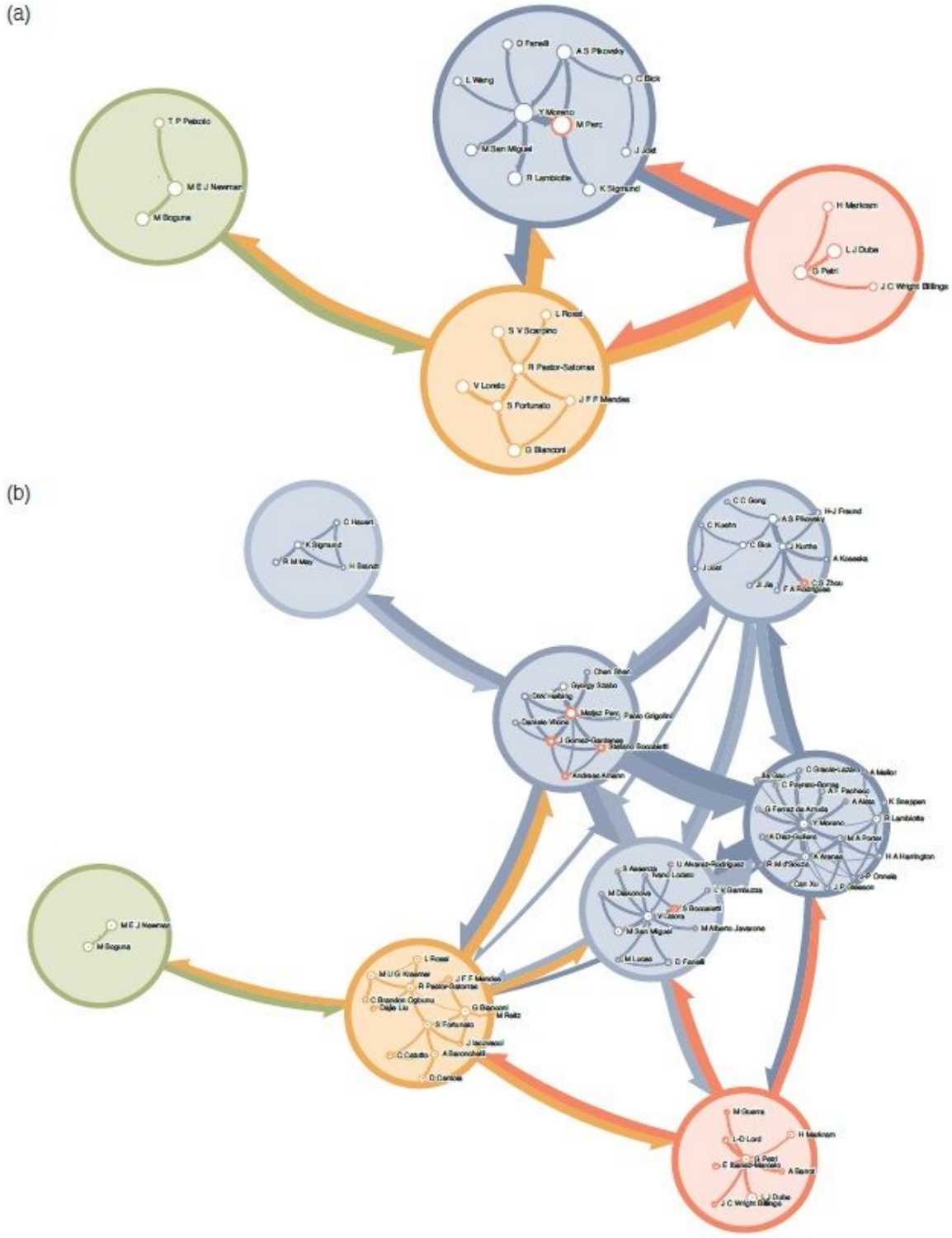

Figure 7

Hierarchical maps of the collaboration hypergraph using (a) the bipartite representation and (b) the multilayer hyperedge-similarity representation. Module colours are the same as in Fig. 4(a). Aggregated inter-module links with sizes proportional to the exiting flow volume and length inversely proportional to the flow volume. White sub-modules are labelled with the top-ranked author. The largest blue top module in (a) contains ten sub-modules. In (b), the partition assigns those nodes to five top modules containing

more sub-modules. S. Boccaletti, one of the most overlapping authors and highlighted in red, is assigned to one module in (a) and three top modules and six sub-modules in (b).

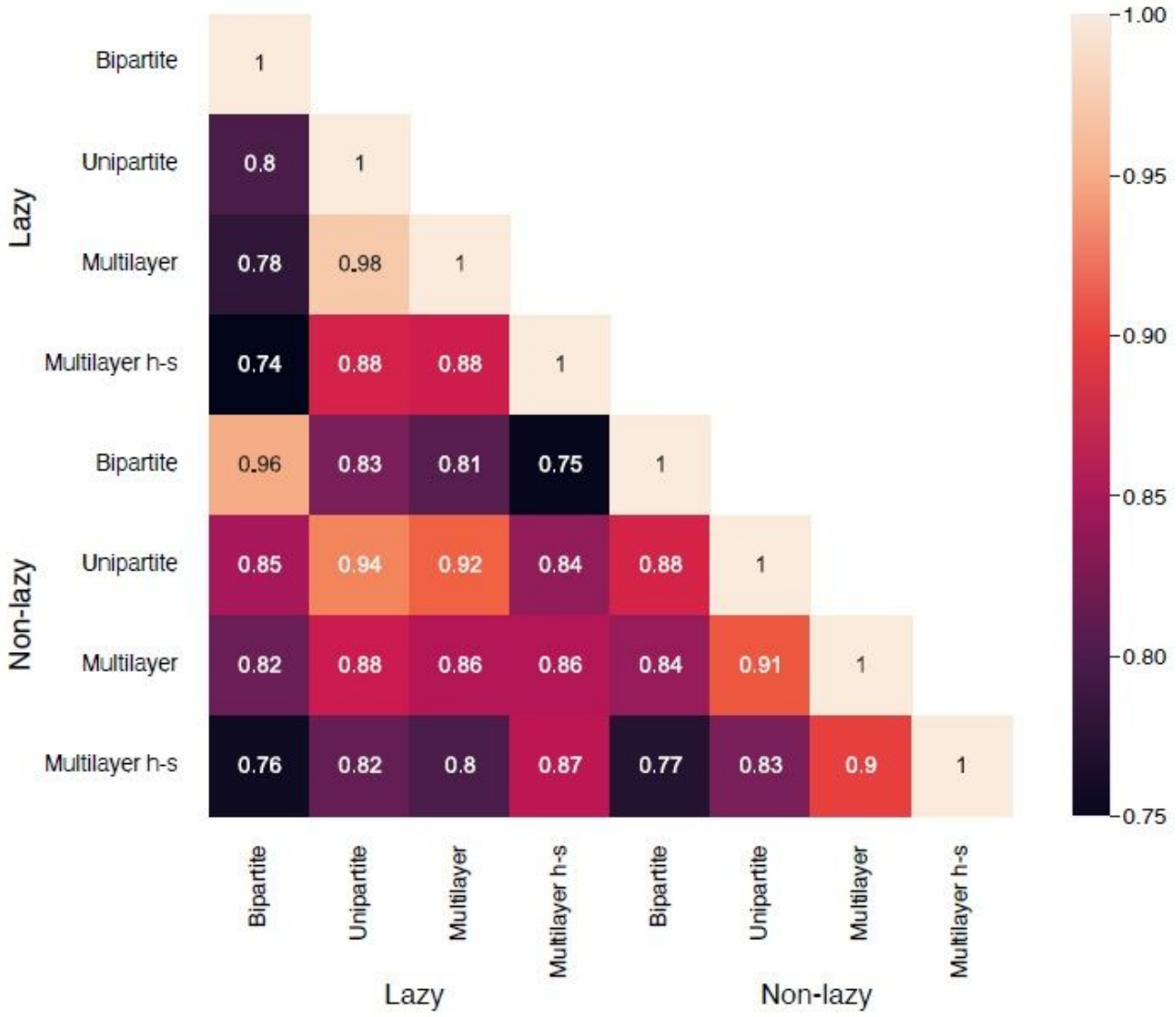

Figure 8

Leaf module assignments' adjusted mutual information for different random walk dynamics and hypergraph representations. The bipartite representations differ the most from the other representations, and the unipartite and multilayer representations are most similar at the leaf level.