# Sampling Networks from Modular Compression of Network Flows

**Christopher Blöcker**
Chair of Machine Learning for Complex Networks
University of Würzburg, Germany
Data Analytics Group, Department of Informatics
University of Zürich, Switzerland

**Jelena Smiljanić**
Integrated Science Lab, Department of Physics
Umeå University, Sweden

**Martin Rosvall**
Integrated Science Lab, Department of Physics
Umeå University, Sweden

**Ingo Scholtes**
Chair of Machine Learning for Complex Networks
University of Würzburg, Germany
Data Analytics Group, Department of Informatics
University of Zürich, Switzerland

## Abstract

Traditional benchmark models generate networks with given structural characteristics such as degree distribution, degree correlations, or community structure but do not consider dynamic processes on networks. Since dynamics are often superordinate to structure and guide network formation, we can learn about the structure from dynamics but lack methods for translating modular dynamics into network structure. To bridge this gap, we introduce a generative network model rooted in the modular compression of dynamic processes on a network as provided by the map equation, an information-theoretic method for community detection. We evaluate our approach by sampling networks according to the modular compression of network flows in empirical networks from different domains. We recover the original community structure and preserve the nodes' expected out-degrees, enabling benchmark networks by sampling from dynamic processes.

## 1 Introduction

Traditional benchmark models generate networks with given structural characteristics such as degree distribution, degree correlations, or community structure [1–5]. They enable testing and evaluating network analysis methods against known ground truth, for example, node membership in planted communities [6]. However, existing benchmark models do not consider dynamic processes on networks describing flow patterns that capture first or higher-order dependencies. But the structure of networks and dynamics on networks are interdependent and, as network structure constrains dynamics, we can learn about dynamics from structure and vice versa. Often, dynamics on networks are prior to structure and essential for our understanding of how networked entities behave and communicate. Dynamics extend beyond pairwise interactions and hold the key to understanding how distant parts can influence each other indirectly.

Flow-based community-detection methods such as Markov stability [7, 8] and the map equation [9] model dynamic processes on networks as random walks and identify communities as those sets of nodes that trap the random walker for a relatively long time. That is, they learn about the structure from the dynamics. Here, we focus on the map equation and reverse this approach: we propose a generative model based on the modular compression of network flows induced by the map equation and analyse emerging structures.

Early work in the study of random networks goes back to Erdős and Rényi [10] who proposed a random graph model where each link in a graph with $n$ nodes exists independently with uniform

C. Blöcker et al., Sampling Networks from Modular Compression of Network Flows (Extended Abstract). Presented at the Second Learning on Graphs Conference (LoG 2023), Virtual Event, November 27–30, 2023.

probability $p$. An emerging structural phenomenon in such Erdős-Rényi random graphs is the transition from disconnected to connected graphs at $p^* = \frac{\ln n}{n}$. Barabási and Albert [1] proposed a random graph model following the preferential attachment principle, reflecting that nodes in many real networks connect to existing nodes at a rate proportional to their degree, leading to networks with power-law distributed node degrees [11]. The preferential attachment mechanism has been found to explain properties of real networks, including clustering and degree correlations [2].

Lancichinetti et al. [3] proposed a method to generate networks with planted community structure where the user can choose the number $n$ of nodes in the networks, the mixing $\mu$, and the exponents for the power-law distributions of node degrees and community sizes. A generalisation of the approach generates networks with overlapping planted community structure [4]. Bazzi et al. [5] proposed a general framework for generating multilayer networks with planted community structure.

Recently, Bontorin et al. [12] formulated a generative model with spatial constraints whose optimisation leads to the emergence of features found in real traffic networks.

## 2 Background

Here, we review the map equation and a related node similarity score, MapSim, which we use as the basis for defining a generative model based on dynamic processes on networks.

### 2.1 The Map Equation

The map equation [9] is an information-theoretic objective function for community detection that uses random walks to model dynamics on networks, introducing a notion of flow. By exploiting the duality between compression and finding patterns in data, the map equation turns community detection into a compression problem, aiming to minimise the codelength: the per-step number of bits to describe a dynamical process represented as a random walk.

In the simplest case, we group all nodes into a single community, call it $\mathsf{M}_1$, and assign unique codewords to the nodes, for example using Huffman coding. The codelength $L(\mathsf{M}_1)$ is the Shannon entropy of the nodes' stationary visit rates, $L(\mathsf{M}_1) = \sum_u p_u \log_2 p_u$, where $p_u$ is $u$'s visit rate.

In networks with community structure (Fig. 1(b)), we can compress the codelength by partitioning nodes into disjoint modules $\mathsf{m} \in \mathsf{M}$ and assigning codewords that are unique within modules. This enables re-using the same codewords across different modules, resulting in shorter codewords on average. But for a unique description of the random walk, we need to add a module-exit codeword per module and an index-level codebook to encode transitions into modules. Then, the codelength is a weighted average of module-level and index-level codelengths, expressed by the two-level map equation: $L(\mathsf{M}) = qH(Q) + \sum_{\mathsf{m}\in\mathsf{M}} p_{\mathsf{m}} H(P_{\mathsf{m}})$. Here, $q$ is the rate at which the random walker uses the index-level codebook, and $Q$ is the set of module entry probabilities; $p_{\mathsf{m}}$ is the rate at which the random walker uses module $\mathsf{m}$'s codebook, including exiting, and $P_{\mathsf{m}}$ is the set of module-normalised node visit rates, including the exit rate.

Minimising the map equation is a search problem and means identifying the community structure that describes the random walker's movement patterns most efficiently; In practice, the map equation neither simulates random walks nor assigns codewords. Through recursion, the map equation can be generalised for networks with nested communities [13].

### 2.2 Map Equation Similarity

Map equation similarity, MapSim for short, is an information-theoretic node similarity measure based on the map equation's compression principles: MapSim interprets the modular structure of a network as an implicit embedding of its nodes in a non-metric latent space and relates the similarity between nodes $u$ and $v$ to the number of bits required to describe a random-walker step from $u$ to $v$ [14]. The key insight behind MapSim is that, while a network's topology constrains a random walker's steps, a coding scheme for describing transitions allows us to calculate similarities between any node pair, whether the nodes are connected or not. Applied to an unsupervised link-prediction task, MapSim was shown to outperform popular embedding methods in terms of the area under both the receiver-operator and precision-recall curves.

MapSim measures the similarity between nodes $u$ and $v$ as the rate at which a random walker transitions from $u$ to $v$, based on the network's modular structure. The rate at which a random walker

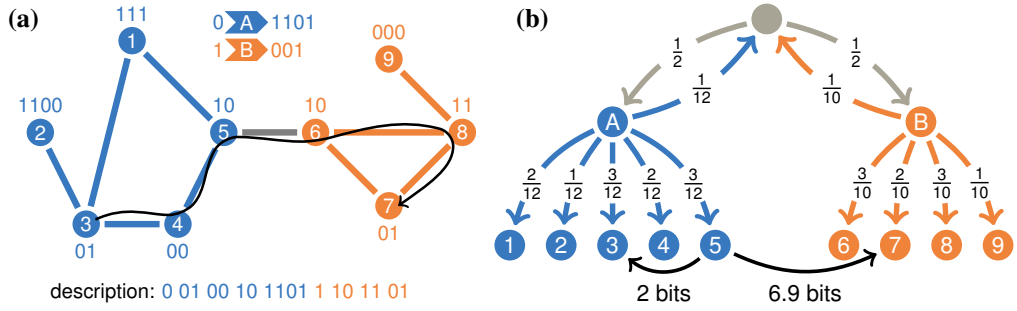


**Figure 1: (a)** A synthetic network with two communities for illustration. Each codeword is unique within modules, but the same codeword can be re-used across modules for a shorter average length. The description of the random-walk sequence on the network is shown at the bottom. **(b)** Representation of the network's community structure as a tree, annotated with transition rates for the random walker. The tree can be used to calculate description lengths for links as well as for non-links. Describing a random walker transition from node 5 to node 7 requires $-\log_2\left(\frac{1}{12}\cdot\frac{1}{2}\cdot\frac{2}{10}\right)\approx 6.9$ bits.

visits nodes within the current module is simply the respective target node's module-normalised visit rate. To find the transition rates to nodes in a different module, MapSim multiplies the module-normalised visit rates along the shortest path in the coding tree (Figure 1(b)).

## 3 Sampling from Modular Compression of Network Flows

We generalise MapSim and propose a sampling approach based on the modular flows in a given network. We interpret the similarity between nodes $u$ and $v$ as a distance by taking their logarithm with base 2: $d_{uv} = -\log_2 \operatorname{mapsim}(\mathsf{M}, u, v)$, and calculate distances between all node pairs. Then, we turn the distances into probabilities on a per-node basis using the softmax function,

$$p_{uv} \propto k_u^{\text{out}} \cdot \frac{2^{-\beta d_{uv}}}{\sum_{v\neq u} 2^{-\beta d_{uv}}}, \tag{1}$$

where $\beta$ is a temperature parameter that controls how peaked the resulting probability distribution is, and $k_u^{\text{out}}$ is node $u$'s out-degree. Finally, we sample links using the probabilities defined in Equation (1). Because link probabilities depend on the source and target nodes' degrees, generating a network requires considering $n^2$ many possible links, where $n$ is the number of nodes. For calculating MapSim values, we can exploit the coding scheme's modular structure for describing transitions. Since the similarity $\operatorname{mapsim}(\mathsf{M}, u, v)$ does not depend directly on node $u$, but on its module $\mathsf{m}_u$ [14], we only need to compute $m \cdot n$ similarities, where $m \ll n$ is the number of modules, typically scaling as $\sqrt{n}$ [15]. For softmax normalisation, we can precompute the values for the denominator, requiring in total $m$ normalisation values – one per module – resulting in $m \cdot n$ operations.

We calculate the expected in and out degrees resulting from our sampling approach by summing over all nodes. The expected out-degrees $k_u^{\text{out}}$ are preserved, $E\left[k_u^{\text{out}}\right] = \sum_v k_u^{\text{out}} \cdot \frac{2^{-\beta d_{uv}}}{\sum_v 2^{-\beta d_{uv}}} = k_u^{\text{out}}$, while the expected in-degrees $k_u^{\text{in}}$ are randomised $E\left[k_u^{\text{in}}\right] = \sum_v k_v^{\text{out}} \cdot \frac{2^{-\beta d_{uv}}}{\sum_v 2^{-\beta d_{uv}}} \neq k_u^{\text{in}}$. This does not hold in undirected networks because we cannot distinguish between in and out links.

**Table 1:** Data set of four networks where $|V|$ is the number of nodes, $|E|$ the number of edges, $\langle k\rangle$ the average degree, $|\mathsf{M}|$ the number of communities, and $\mu$ the mixing.

| Network | Ref | Type | $\lvert V\rvert$ | $\lvert E\rvert$ | $\langle k\rangle$ | $\lvert\mathsf{M}\rvert$ | $\mu$ |
|---|---|---|---|---|---|---|---|
| Interactome | [16] | undirected | 161 | 209 | 1.3 | 7 | 0.15 |
| Highschool | [17] | directed | 67 | 359 | 5.36 | 8 | 0.31 |
| Anybeat | [18] | directed | 8518 | 58799 | 6.9 | 542 | 0.53 |
| arXiv citation HepPh | [19] | directed | 12711 | 139981 | 11.01 | 270 | 0.38 |

By using the various generalisations of the map equation [20–23], we can incorporate various structural patterns in the community-detection stage, such as hierarchical and higher-order patterns

or node metadata. Because MapSim is based on the detected communities, these generalisations naturally integrate into sampling from MapSim to generate networks.

To analyse the structural properties of networks generated by our model, we consider dynamical processes on four empirical networks: a protein interaction network, a high school social network, an online social network, and a citation network (Table 1). For each network, we generate 100 samples for each $\beta \in [0, 3]$ with step size $0.1$ and show the average symmetrically normalised adjusted mutual information (AMI) between the generated and original network's community structure, the mixing $\mu$, as well as the average number of sampled links (Fig. 2). For detecting communities, both in the original and the generated networks, we use Infomap, an open-source software that seeks to minimise the map equation.

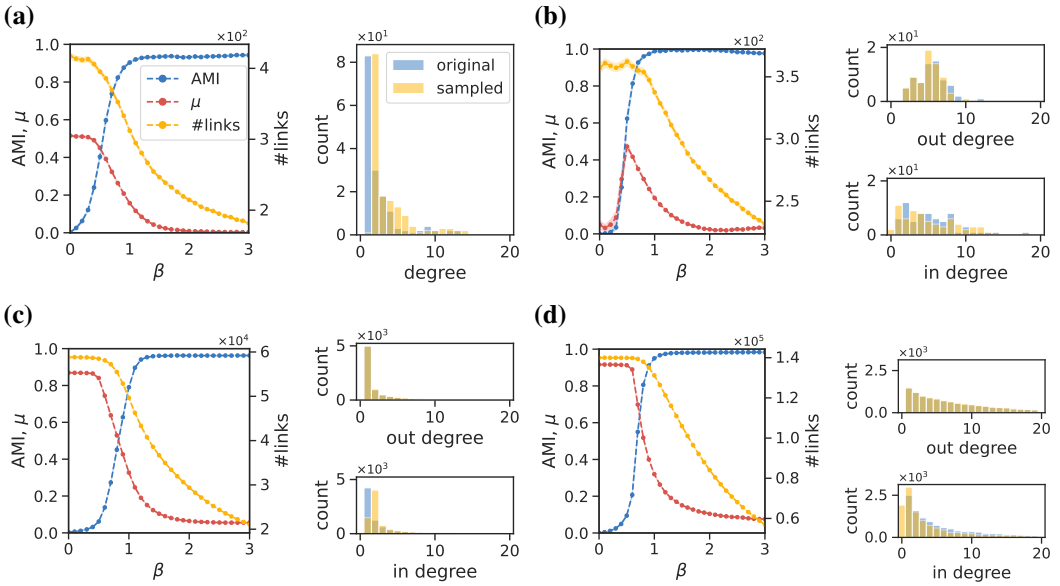


**Figure 2:** Sampling from the modular flows of four real networks. Each panel shows the average AMI, average mixing $\mu$, and the average number of links for 100 samples for each $\beta \in [0, 3]$, and the original and resulting degree distributions for $\beta = 1$. **(a)** Interactome, **(b)** Highschool, **(c)** Anybeat, **(d)** arXiv citation HepPh.

Our experiments confirm that we preserve the out-degree distribution in directed networks that are sufficiently large. For $\beta = 0$, links are uniformly distributed, resulting in AMI values of $0$ in all cases. Depending on whether random community structure emerges, initial mixing values $\mu$ are either relatively high or $0$. As we increase $\beta$, the sampled networks become sparser and fewer inter-community links manifest. Depending on the network, AMI scores increase sharply and tend to $1$ around $\beta = 1$. For large values of $\beta$, we recover the identified community structure because the softmax normalisation pushes links to be placed within communities while probabilities for inter-community become smaller.

## 4 Conclusion

By turning MapSim, an information-theoretic node similarity measure, into a generative model for networks, we show that mesoscopic network structures can be recovered from modular compression of dynamics on networks, connecting generative and descriptive notions of community detection. This connection opens new avenues for studying how dynamic processes influence the properties of the networks they generate, which we begin to explore by using our approach on a set of real networks. Furthermore, our sampling approach can be used to generate benchmark networks based on dynamical processes incorporating first or higher-order information to evaluate the performance of network analysis methods, adding to the scarce amount of current higher-order benchmark models.

In future work, we will investigate incorporating node metadata and higher-order dependencies into the network generation. Furthermore, we will explore alternative sampling approaches for scalability and efficient sampling from the modular compression of network regularities.

## References


[1] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999. 1, 2

[2] Alexei Vázquez. Growing network with local rules: Preferential attachment, clustering hierarchy, and degree correlations. *Phys. Rev. E*, 67:056104, May 2003. doi: 10.1103/PhysRevE.67.056104. URL https://link.aps.org/doi/10.1103/PhysRevE.67.056104. 2

[3] Andrea Lancichinetti, Santo Fortunato, and Filippo Radicchi. Benchmark graphs for testing community detection algorithms. *Phys. Rev. E*, 78:046110, Oct 2008. doi: 10.1103/PhysRevE.78.046110. URL https://link.aps.org/doi/10.1103/PhysRevE.78.046110. 2

[4] Andrea Lancichinetti and Santo Fortunato. Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Phys. Rev. E*, 80:016118, Jul 2009. doi: 10.1103/PhysRevE.80.016118. URL https://link.aps.org/doi/10.1103/PhysRevE.80.016118. 2

[5] Marya Bazzi, Lucas G. S. Jeub, Alex Arenas, Sam D. Howison, and Mason A. Porter. A framework for the construction of generative models for mesoscale structure in multilayer networks. *Phys. Rev. Res.*, 2:023100, Apr 2020. doi: 10.1103/PhysRevResearch.2.023100. URL https://link.aps.org/doi/10.1103/PhysRevResearch.2.023100. 1, 2

[6] Günce Keziban Orman, Vincent Labatut, and Hocine Cherifi. Towards realistic artificial benchmark for community detection algorithms evaluation. *International Journal of Web Based Communities*, 9(3):349–370, 2013. doi: 10.1504/IJWBC.2013.054908. URL https://www.inderscienceonline.com/doi/abs/10.1504/IJWBC.2013.054908. PMID: 54908. 1

[7] J.-C. Delvenne, S. N. Yaliraki, and M. Barahona. Stability of graph communities across time scales. *Proceedings of the National Academy of Sciences*, 107(29):12755–12760, 2010. doi: 10.1073/pnas.0903215107. URL https://www.pnas.org/doi/abs/10.1073/pnas.0903215107. 1

[8] Renaud Lambiotte, Jean-Charles Delvenne, and Mauricio Barahona. Random walks, markov processes and the multiscale modular organization of complex networks. *IEEE Transactions on Network Science and Engineering*, 1(2):76–90, 2014. doi: 10.1109/TNSE.2015.2391998. 1

[9] Martin Rosvall and Carl T Bergstrom. Maps of random walks on complex networks reveal community structure. *Proceedings of the national academy of sciences*, 105(4):1118–1123, 2008. 1, 2

[10] Paul Erdős and Alfréd Rényi. On the evolution of random graphs. *Publications of the Mathematical Institute of the Hungarian Academy of Sciences*, 5(1):17–60, 1960. 1

[11] Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Rev. Mod. Phys.*, 74:47–97, Jan 2002. doi: 10.1103/RevModPhys.74.47. URL https://link.aps.org/doi/10.1103/RevModPhys.74.47. 2

[12] Sebastiano Bontorin, Giulia Cencetti, Riccardo Gallotti, Bruno Lepri, and Manlio De Domenico. Emergence of complex network topologies from flow-weighted optimization of network efficiency, 2023. 2

[13] Martin Rosvall and Carl T. Bergstrom. Multilevel compression of random walks on networks reveals hierarchical organization in large integrated systems. *PLOS ONE*, 6(4):1–10, 04 2011. doi: 10.1371/journal.pone.0018209. URL https://doi.org/10.1371/journal.pone.0018209. 2

[14] Christopher Blöcker, Jelena Smiljanić, Ingo Scholtes, and Martin Rosvall. Similarity-based link prediction from modular compression of network flows. In *Proceedings of the First Learning on Graphs Conference*, volume 198, pages 52:1–52:18. PMLR, 09–12 Dec 2022. 2, 3

[15] Amir Ghasemian, Homa Hosseinmardi, and Aaron Clauset. Evaluating overfit and underfit in models of network community structure. *IEEE Transactions on Knowledge and Data Engineering*, 32(9):1722–1735, 2020. doi: 10.1109/TKDE.2019.2911585. 3

[16] Thijs Beuming, Lucy Skrabanek, Masha Y. Niv, Piali Mukherjee, and Harel Weinstein. PDZBase: a protein–protein interaction database for PDZ-domains. *Bioinformatics*, 21(6):827–828, 10 2004. ISSN 1367-4803. doi: 10.1093/bioinformatics/bti098. URL https://doi.org/10.1093/bioinformatics/bti098. 3

[17] James Samuel Coleman et al. Introduction to mathematical sociology. *Introduction to mathematical sociology.*, 1964. 3

[18] Michael Fire, Rami Puzis, and Yuval Elovici. *Link Prediction in Highly Fractional Data Sets*, pages 283–300. Springer New York, New York, NY, 2013. ISBN 978-1-4614-5311-6. doi: 10.1007/978-1-4614-5311-6_14. URL https://doi.org/10.1007/978-1-4614-5311-6_14. 3

[19] Johannes Gehrke, Paul Ginsparg, and Jon Kleinberg. Overview of the 2003 kdd cup. *SIGKDD Explor. Newsl.*, 5(2):149–151, dec 2003. ISSN 1931-0145. doi: 10.1145/980972.980992. URL https://doi.org/10.1145/980972.980992. 3

[20] Martin Rosvall, Alcides V. Esquivel, Andrea Lancichinetti, Jevin D. West, and Renaud Lambiotte. Memory in network flows and its effects on spreading dynamics and community detection. *Nature Communications*, 5(1):4630, Aug 2014. ISSN 2041-1723. doi: 10.1038/ncomms5630. URL https://doi.org/10.1038/ncomms5630. 3

[21] Manlio De Domenico, Andrea Lancichinetti, Alex Arenas, and Martin Rosvall. Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems. *Phys. Rev. X*, 5:011027, Mar 2015. doi: 10.1103/PhysRevX.5.011027. URL https://link.aps.org/doi/10.1103/PhysRevX.5.011027.

[22] Scott Emmons and Peter J. Mucha. Map equation with metadata: Varying the role of attributes in community detection. *Phys. Rev. E*, 100:022301, Aug 2019. doi: 10.1103/PhysRevE.100.022301. URL https://link.aps.org/doi/10.1103/PhysRevE.100.022301.

[23] Christopher Blöcker and Martin Rosvall. Mapping flows on bipartite networks. *Phys. Rev. E*, 102:052305, Nov 2020. doi: 10.1103/PhysRevE.102.052305. URL https://link.aps.org/doi/10.1103/PhysRevE.102.052305. 3