Exploring the solution landscape enables more reliable network community detection

Joaquín Calatayud,[1] Rubén Bernardo-Madrid,[2] Magnus Neuman,[1] Alexis Rojas,[1] and Martin Rosvall[1]

[1] *Integrated Science Lab, Department of Physics, Umeå University, Sweden*
[2] *Department of Conservation Biology, Estación Biológica de Doñana (EBD-CSIC), Spain*

(Dated: May 28, 2019)

To understand how a complex system is organized and functions, researchers often identify communities in the system's network of interactions. Because it is practically impossible to explore all solutions to guarantee the best one, many community-detection algorithms rely on multiple stochastic searches. But for a given combination of network and stochastic algorithm, how many searches are sufficient to find a solution that is good enough? The standard approach is to pick a reasonably large number of searches and select the network partition with the highest quality or derive a consensus solution based on all network partitions. However, if different partitions have similar qualities such that the solution landscape is degenerate, the single best partition may miss relevant information, and a consensus solution may blur complementary communities. Here we address this degeneracy problem with coarse-grained descriptions of the solution landscape. We cluster network partitions based on their similarity and suggest an approach to determine the minimum number of searches required to describe the solution landscape adequately. To make good use of all partitions, we also propose different ways to explore the solution landscape, including a significance clustering procedure. We test these approaches on synthetic and real-world networks, and find that different networks and algorithms require a different number of searches and that exploring the coarse-grained solution landscape can reveal noteworthy complementary solutions and enable more reliable community detection.

I. INTRODUCTION

Researchers in many disciplines across science use tools from network science to understand the structure, dynamics, and function of complex systems. For example, identifying possibly nested groups of densely connected nodes with community detection algorithms can highlight important network structures [1–3]. Most community detection algorithms seek to find the network partition that optimizes a quality score based on a specific definition of what constitutes a community. Because finding the best network partition is an NP-hard problem, many algorithms rely on stochastic search strategies and require multiple runs to avoid local minima with bad solutions [4–6]. However, all algorithms are more or less sensitive to degenerate solutions with similar quality scores for dissimilar partitions [7]. Moreover, small changes in a network due to noise can drastically change the best solution, and a weak community structure can worsen this degeneracy problem. Therefore, reliable community detection must be able to successfully deal with degenerate solutions.

To handle the degeneracy problem, consensus clustering seeks to combine information from multiple network partitions [8–10]. The aim is to summarize the partitions in a single and possibly new partition with graph-based, combinatorial, or statistical techniques. Various approaches include finding the median partition or the one that shares the most information with other partitions [8, 11], consolidating groups of partitions with hypergraph methods [8], and re-clustering a co-occurrence network with the same community detection algorithm–[9, 10]. Although consensus clustering can alleviate some degeneracy problems and give higher quality solutions, using a single consensus partition can also waste important information or lead to misleading solutions if the partitions are incompatible. Moreover, disregarding the partition qualities can aggravate these problems when the number of low-quality partitions outweighs the number of high-quality partitions (Fig. 1).

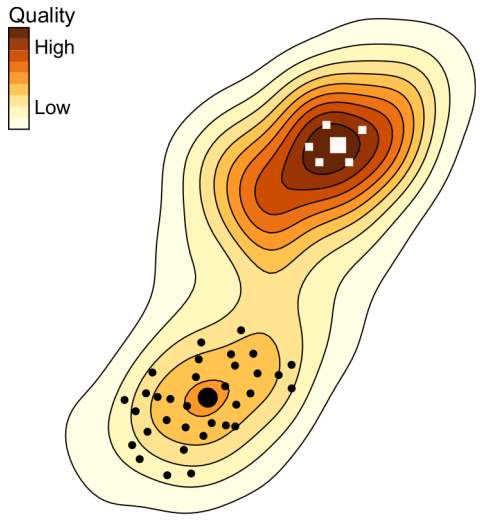

FIG. 1. A schematic solution landscape projected into a two-dimensional space with isolines for quality score. White squares and black circles represent two network partition clusters, with partitions distributed based on their partition distances. Large symbols represent cluster centers. A consensus solution biased toward the numerous partitions marked with a black circle may have a lower quality score than any of the detected partitions.

Studying the full solution landscape with all network

partitions and corresponding quality measures results in no wasted information. However, such approaches are in practice limited to approximate visual explorations and the qualitative assessment of degenerate solutions [7, 12]. Moreover, for a given network and community detection algorithm, it is unclear how many solutions are enough to describe the solution landscape adequately. As a result, we lack quantitative approaches that both highlight essential structures in the solution landscape and determine when it is safe to stop searching for new or better solutions. These challenges call for novel methods to comprehend and make use of the solution landscape to better understand the structure and dynamics of complex systems.

Here we present a partition clustering approach that explores the solution landscape of standard and multilevel community detection algorithms. To assess the completeness of the coarse-grained solution landscape, we cluster similar partitions together with a fast stream-clustering algorithm and estimate the probability that new partitions will fall within already defined partition clusters. For a coarse-grained solution landscape that meets a user-specified resolution, we propose different ways to explore the space of partitions, including visual explorations to reveal complementary solutions and a statistical test to identify significant communities. We validate our approach on synthetic networks as well as a real-world network of worldwide mammal occurrences.

II. DESCRIBING THE SOLUTION LANDSCAPE

A. Network partition distance

To describe the solution landscape, we first compute distances between partitions. While any of the many partition distance measures developed for different networks and research questions would work, most of them apply only to hard partitions that cannot capture hierarchical or overlapping community structures [13–15]. Because these types of community structures are common in many real-world networks [16–19], some distance measures have been generalized to capture either overlapping or hierarchical communities [16, 18, 20], but rarely both [21]. To capture different types of community structures and make it easy to interpret the results, we want a flexible and simple distance measure.

Because a community of nodes is the building block of all types of community structures, we base the partition distance measure on pairwise community comparisons, regardless of whether they are in different hierarchical levels or whether nodes belong to more than one community. Specifically, we measure the weighted average of the minimum Jaccard distance over all communities in partition P to a community in partition P', with the weight given by the fraction of node assignments. That is, for each community i in partition P with set of nodes C_i^P, we measure the minimum Jaccard distance to any community j in partition P' with set of nodes $C_j^{P'}$, and calculate the weighted average based on the number of nodes in C_i^P, $|C_i^P|$, and the number of community assignments in partition P, $\sum_k |C_k^P|$, such that the distance $d_{PP'}$ from partition P to partition P' is

$$d_{PP'} = \sum_i \min_j \left(1 - \frac{C_i^P \cap C_j^{P'}}{C_i^P \cup C_j^{P'}}\right) \frac{|C_i^P|}{\sum_k |C_k^P|}. \quad (1)$$

Because $d_{PP'}$ need not be equal to $d_{P'P}$, we calculate the average for a symmetric partition distance measure [22],

$$\bar{d}_{PP'} = \frac{1}{2} d_{PP'} + \frac{1}{2} d_{P'P}. \quad (2)$$

This partition distance works with hard, overlapping, and hierarchical communities. It is zero for identical partitions, and approaches 1 as they become completely dissimilar. Between these extremes, the partition distance gives the weighted average fraction of nodes that best-matching communities do not have in common.

B. Network partition clustering algorithm

Using the proposed network partition distance, we describe the solution landscape with clusters of similar network partitions. While many clustering algorithms can output such clusters, those algorithms generally involve NP-hard optimization problems in themselves. However, to identify dissimilar partitions with high quality, we do not need a solution landscape that optimizes some quality function. Instead, a fast and transparent deterministic approach that decides the number of clusters provides multiple advantages: First, a fast algorithm can run together with a stochastic community detection algorithm and decide when it is safe to stop to achieve a good result. Second, a deterministic algorithm that does not require a prespecified number of clusters evades the ambiguities that come with multiple solutions. Third, a transparent algorithm that produces interpretable clusters and a comprehensible solution landscape simplifies further analysis. Therefore, instead of relying on established clustering algorithms developed for other purposes, given a partition distance threshold $d_{\max}$, we perform the following steps:

1. Order all p network partitions from highest to lowest quality.

2. Let the highest quality network partition form cluster center 1.

3. Repeat until all network partitions have been clustered. Among the not yet clustered partitions, pick the one with the highest quality and assign it to the first of the m cluster centers that it is closer to than $d_{\max}$. If no such cluster center exists, let it form cluster center $m + 1$.

For example, in the schematic solution landscape in Fig. 1, the network partition clustering algorithm first lets the partition marked with a big square form the center of cluster 1. For an intermediate partition distance threshold, it then assigns the other partitions marked with squares to the same cluster before it lets the partition marked with a big circle form the center of cluster 2 and assigns the other partitions marked with circles to that cluster.

The partition distance threshold specifies the resolution of the coarse-grained solution landscape. Lowering the threshold gives more clusters with more similar network partitions and increasing the threshold gives fewer clusters with less similar network partitions.

We have implemented the partition clustering code in C++, which has worst-case time-complexity $\mathcal{O}(pm)$, and made it available for anyone to use at `https://github.com/mapequation/partition-validation`

C. Solution landscape completeness

We say that a solution landscape is complete when new network partitions at most marginally affect its coarse-grained description. Accordingly, when a solution landscape is complete, it is safe to stop searching for better network partitions. Intuitively, we need fewer partitions to describe the solution landscape of a network with a clear community structure than that of a network with a diffuse community structure because the former will have more similar partitions. Moreover, the required number of partitions will also depend on the variability of the search algorithm. In any case, for a sufficient number of partitions, the probability that a new partition will fit into existing clusters will be close to 1. We use this probability as a validation score to assess the solution landscape completeness. and to determine when to stop searching. For example, we stop the search algorithm when the validation score is higher than accuracy level 0.9. To avoid random effects caused by the search order of the stochastic community detection algorithm, we use repeated random sub-sampling validation and hold out 100 partitions, or $p/2$ when the number of partitions is fewer than 200, to estimate the validation score.

D. Solution landscape exploration

A complete coarse-grained solution landscape with clusters centered around locally high-quality partitions simplifies further analysis and makes the results more reliable. First, it indicates when it is safe to stop searching for a better solution because the accuracy level and partition distance threshold put a limit on the value of continuing. For example, when a solution landscape is complete at a high accuracy level for a small partition distance threshold, summary statistics based on all partitions will be reproducible and reliable. Second, it directly gives an idea about the spread of network partitions through the number of clusters for a given partition distance threshold. For more detailed analysis, alluvial diagrams can highlight qualitative pairwise differences between partitions [23] and various embedding techniques can depict the overall solution landscape [24]. Third, it can speed up further analysis with controlled information loss as comparing all pairs of cluster centers rather than all pairs of partitions reduces the computational complexity from $\mathcal{O}(p^2)$ to $\mathcal{O}(m^2)$.

Useful further analysis include finding communities or node assignments that are stable across many partitions. For example, in networks with partially clear community structure, distinguishing stable from unstable communities enables more reliable analysis. While approaches exist for assessing the significance of communities given a set of partitions [23, 25], these approaches only work for hard two-level partitions. Therefore, we propose an approach that also assesses the significance for hierarchical or overlapping communities. A straightforward approach to assessing the significance of a community would be to calculate the fraction of partitions in which the community appears. However, this significance test is overly demanding as communities with only slight variations in node composition would be considered non-significant. Consequently, we relax the demand for exact matching and reuse the minimum Jaccard distance of the network partition distance in Eq. (1) with a threshold. We measure the significance α_i^R of community i in the highest-quality or other reference partition R as the fraction of partitions that have a community with a smaller distance to i than a threshold τ,

$$\alpha_i^R = \frac{1}{p-1} \sum_{P \neq R} \Theta \left[\tau - \min_j \left(1 - \frac{C_i^R \cap C_j^P}{C_i^R \cup C_j^P} \right) \right], \quad (3)$$

where Θ is the Heaviside step function.

Stable communities can contain both stable and unstable node assignments, and we need a means to distinguish between them. Therefore, to measure the community-assignment significance η_v^R of node v in reference partition R, we calculate the fraction of partitions in which v appears in the community that is most similar to v's community in the reference partition. Using the Kronecker delta function δ, the community-assignment significance can be written

$$\eta_v^R = \frac{1}{p-1} \sum_{P \neq R} \delta \left(c_v^P, c_v^{RP} \right), \quad (4)$$

where c_v^P is the community index of node v in partition P, and $c_v^{RP} = \arg\max_j C_{c_v^R}^R \cap C_j^P / C_{c_v^R}^R \cup C_j^P$ is the community index of the community in partition P that is most similar to the community of v in partition R.

III. RESULTS AND DISCUSSION

A. Solution landscape of synthetic networks

We tested our approach on LFR benchmark networks with different intercommunity link probabilities μ [26]. We generated networks with 500 nodes, of average degree 10 and maximum degree 20, with community sizes distributed between 20 and 100 nodes, and with four different intercommunity link probabilities, $\mu = 0.1$, 0.2, 0.3 and 0.4, for less and less pronounced communities. To account for the internal variability of the LFR benchmark networks, we generated 25 synthetic networks for each μ. We analyzed these networks with two popular and contrasting stochastic algorithms for community detection: Infomap [4, 27] and Bayesian inference of the stochastic blockmodel (BSBM) as implemented in the graph-tool library [6, 28]. While both algorithms optimize information-theoretic objective functions, Infomap seeks to compress dynamics on a network whereas BSBM seeks to compress the network itself. Moreover, BSBM can handle partition uncertainty based on sampling from the posterior distribution [12], but the solution landscape nevertheless contains useful information about the variability of the partitions. We ran each algorithm 500 times on a given network in incremental steps of first 50 and then 100 times. After each step, we ran the partition clustering algorithm and validated 100 times on 100 subsampled hold-out partitions when $p \geq 200$ and on $p/2$ partitions otherwise.

The benchmark tests show that Infomap tends to generate simpler landscapes than BSBM. That is, for the synthetic networks, Infomap requires fewer partitions to obtain complete solution landscapes. Nevertheless, both methods require more partitions for networks with higher intercommunity link probabilities (Fig. 2). Thus, a less pronounced community structure requires a larger number of searches to obtain a complete solution landscape. The choice of partition distance threshold $d_{\max}$ also influences the required number of searches. To exemplify this, we used two threshold values for validation, $d_{\max} = 0.025$ and $d_{\max} = 0.05$. With the higher threshold, more hold-out partitions fit in clusters such that the validation score increases (Fig. 2). Therefore, the choice of partition distance threshold should reflect a compromise between accuracy and efficiency and depend on the particular problem at hand.

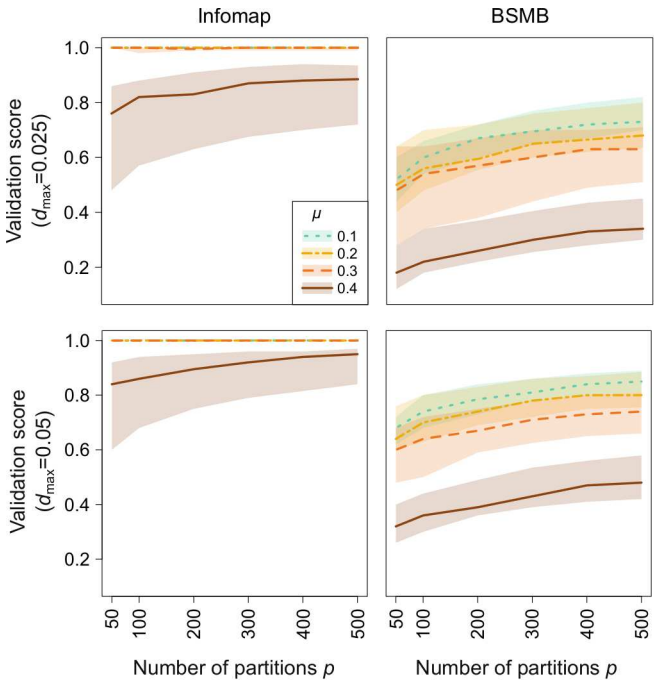

FIG. 2. Infomap and BSBM solution landscape completeness for synthetic networks generated with four intercommunity link probabilities μ. The validation scores with solid medians and shaded regions between the 25th and 75th quantiles for different numbers of partitions with partition distance thresholds $d_{\max} = 0.025$ and $d_{\max} = 0.05$. Infomap requires fewer partitions than BSMB for complete solution landscapes. Both methods require more partitions for less pronounced communities.

B. Solution landscape of a mammal occurrence network

We further explored the solution landscape in a real-world case using a terrestrial mammal occurrence network. This bipartite network consists of 4999 mammal species and 10775 grid cells of 1 degree that cover the surface of the Earth [29]. A link exists between a species and a grid cell if the species occurs in the grid cell. The resulting communities form global-scale areas that share similar species called bioregions. We analyzed the community structure with the multilevel versions of Infomap [17] and BSBM [30] by generating 1500 partitions with each algorithm. We chose $d_{\max} = 0.2$, which roughly corresponds to partition differences that cover up to 20% of the Earth's surface. Higher partition distances indicate major changes in the bioregional configuration, which require separate examination. Nevertheless, to illustrate the effect of different thresholds, we also used three smaller values, $d_{\max} = 0.025$, 0.05, and 0.1. To validate the solution landscape under different numbers of runs, we used 200–1500 partitions with 100 hold-out partitions sub-sampled 100 times.

The results on the real networks resemble those on the synthetic networks. Compared with Infomap, BSBM again generated partitions with higher variability and a more complex solution landscape. Because the distance was higher than $d_{\max} = 0.2$ between each pair of partitions, each partition formed its own cluster such that the validation score was 0. Therefore, we only explored the results of Infomap, for which we achieved complete solution landscapes with validation scores above 0.9 for all tested threshold values $d_{\max}$ (Fig. 3). For example, for the lowest tested $d_{\max} = 0.025$, the validation score was

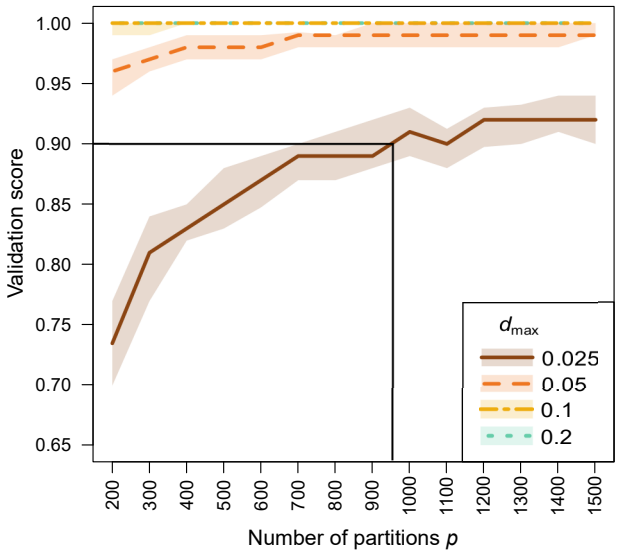

FIG. 3. Validation score for landscape completeness of the terrestrial mammal occurrence network under four partition distance threshold values $J_{\max}$ (0.2, 0.1, 0.05, 0.025).

higher than 0.9 when we used more than 900 partitions, which formed 188 clusters (Fig. 3). In contrast, for the highest tested $d_{\max} = 0.2$, the validation score was higher than 0.9 already at 200 partitions (and likely before), and the 1500 partitions formed two clusters with 970 and 530 partitions, respectively. The cluster centers have similar qualities, 10.689 and 10.695, which Infomap measures as code lengths in bits. Indeed, the clusters have partitions with overlapping code lengths (from 10.695 and 10.697 at the 25th percentile to 10.700 for both clusters at the 75th percentile), which call for further analysis of the degenerate solution landscape.

To explore the qualitative differences between the clusters, techniques such as alluvial diagrams can give a visual overview of major changes between the cluster centers (Fig. 4(a)). In our particular case, however, we can visualize the geographic projection of the spatially explicit grid cells (Fig. 4(b)). At the highest hierarchical level, the major difference is that the second cluster center splits Africa and a southeastern portion of Asia from a large region that encompasses Eurasia and Africa in the first cluster center. At lower hierarchical levels, the first cluster center further subdivides the North American region whereas the second cluster center further subdivides regions in Africa and central Asia. These results show the rich information contained in different partitions, which can reveal meaningful patterns. For instance, the subdivision of Sub-Saharan Africa closely coincides with the Köppen climate classification [31].

Finally, we applied the significance clustering procedure both at the community and node level with the overall highest quality partition as a reference. We used community distance threshold $\tau = 0.2$ to calculate the community significance α_i^R. The community significance is largely in agreement with the previous qualitative visual assessment. The region including Africa and Eurasia is weakly supported, which is also true for the North American and Central Asian regions (Fig. 5). Also, the node significance η_i^A agrees with these results, but provides further information. For instance, the weakly supported African Euro-Asiatic region in the first level appears to hold a significant core of nodes coinciding with northern Eurasia. Moreover, nodes with low significance tend to be placed along regional borders such as the Sahel border and the border separating southern and northern South America. Beyond methodological stochasticity, this result shows that some nodes are inherently more difficult to assign to particular communities.

IV. CONCLUSIONS

We have introduced a fast network partition clustering algorithm to describe the often degenerate solution landscape of stochastic community detection algorithms in coarse-grained form. Our approach establishes a criterion for when it is safe to stop searching for better solutions and start exploring the solution landscape. We further illustrate with visualizations of new statistical tests of communities and node assignments, which give insights into the underlying causes of the solution landscape degeneracy. The validation on real-world as well as synthetic networks highlights how focusing on a single network partition can waste useful information. In contrast, using the entire solution landscape enables more reliable community detection and a better understanding of the organization of complex systems. Beyond community detection, our approach works with any stochastic search with outputs that have measurable distances.

ACKNOWLEDGMENTS

We thank Eloy Revilla for stimulating discussions and Anton Eriksson for helping us to construct the alluvial diagram in Fig. 4. J.C. was supported by the the Carl Trygger Foundation. R.B.-M. was supported by the Spanish Ministry of Economy, Industry and Competitiveness, grant BES-2013-065753. M.N. and A.R. were supported by the Olle Engkvist Byggmästare Foundation. M.R. was supported by the Swedish Research Council, grant 2016-00796.

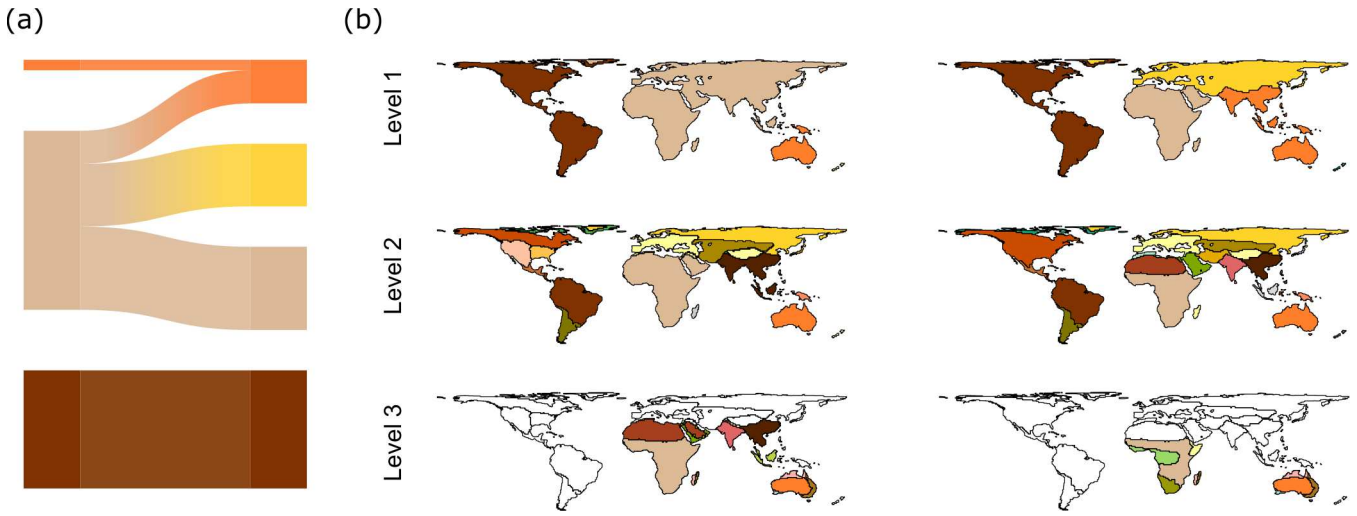

FIG. 4. World bioregions from communities in the two best partition cluster centers.(a) Alluvial diagram showing the differences between the two partitions at the highest hierarchical level. (b) Geographic projection of nodes representing grid cells. In all cases, we obtained three hierarchical levels. The differences show the rich information contained in separate partitions, even when they have similar quality.

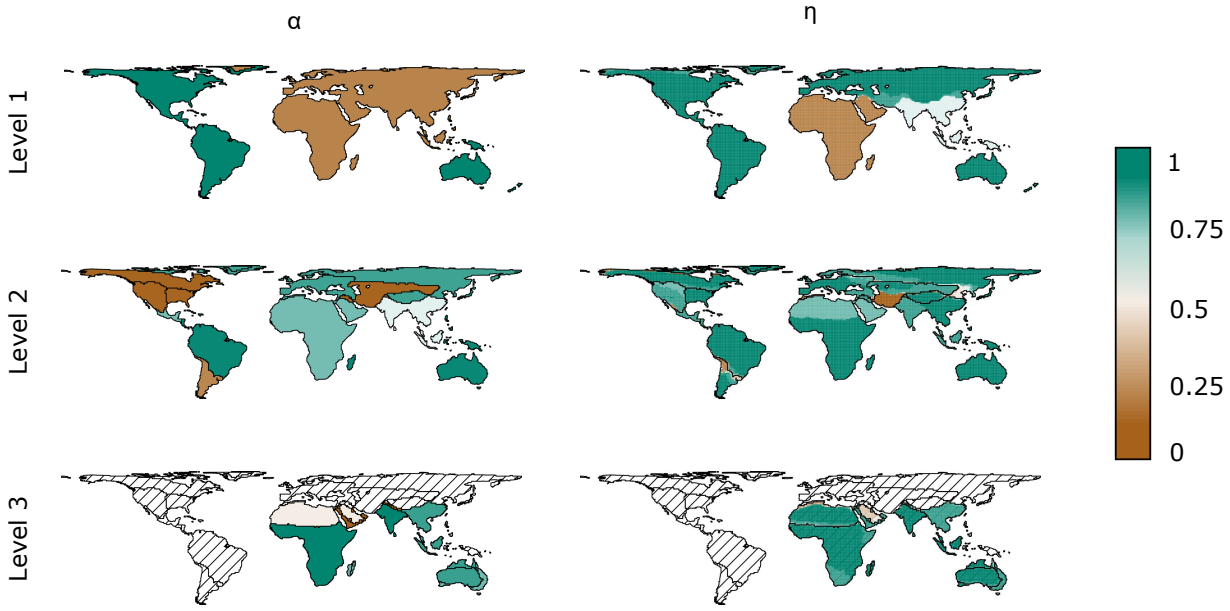

FIG. 5. The fraction of partitions having a community more similar than 0.2 to the reference community (left). The fraction of partitions where a node belongs to the most similar community (right). We see for example that the weakly supported African Euro-Asiatic region in the first level appears to hold a significant core of nodes coinciding with the north of Eurasia, while less significant nodes tend to be placed along bioregional borders. The striped areas correspond to regions that were not further subdivided in the third hierarchical level.

[1] D. Deritei, W. C. Aird, M. Ercsey-Ravasz, and E. R. Regan, Scientific reports **6**, 21957 (2016).

[2] C. Y. Baldwin and K. B. Clark, Managing in the modular age: Architectures, networks, and organizations **149**, 84 (2003).

[3] J. Grilli, T. Rogers, and S. Allesina, Nature communications **7** (2016).

[4] M. Rosvall and C. T. Bergstrom, Proceedings of the National Academy of Sciences **105**, 1118 (2008).

[5] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, Journal of statistical mechanics: theory and experiment **2008**, P10008 (2008).

[6] T. P. Peixoto, Physical Review E **89**, 012804 (2014).

[7] B. H. Good, Y.-A. de Montjoye, and A. Clauset, Physical Review E **81**, 046106 (2010).

[8] A. Strehl and J. Ghosh, Journal of machine learning research **3**, 583 (2002).

[9] A. Lancichinetti and S. Fortunato, Scientific reports **2**, 336 (2012).

[10] A. Tandon, A. Albeshri, V. Thayananthan, W. Alhalabi, and S. Fortunato, Phys. Rev. E **99**, 042301 (2019).

[11] A. Topchy, A. K. Jain, and W. Punch, IEEE transactions on pattern analysis and machine intelligence **27**, 1866 (2005).

[12] T. P. Peixoto, arXiv preprint arXiv:1705.10225 (2017).

[13] L. Danon, A. Diaz-Guilera, J. Duch, and A. Arenas, Journal of Statistical Mechanics: Theory and Experiment **2005**, P09008 (2005).

[14] L. Hubert and P. Arabie, Journal of classification **2**, 193 (1985).

[15] W. M. Rand, Journal of the American Statistical association **66**, 846 (1971).

[16] A. Lancichinetti, S. Fortunato, and J. Kertész, New Journal of Physics **11**, 033015 (2009).

[17] M. Rosvall and C. T. Bergstrom, PloS one **6**, e18209 (2011).

[18] J. I. Perotti, C. J. Tessone, and G. Caldarelli, Physical Review E **92**, 062825 (2015).

[19] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, Nature **435**, 814 (2005).

[20] L. M. Collins and C. W. Dent, Multivariate Behavioral Research **23**, 231 (1988).

[21] A. J. Gates, I. B. Wood, W. P. Hetrick, and Y.-Y. Ahn, arXiv preprint arXiv:1706.06136 (2017).

[22] M. K. Goldberg, M. Hayvanovych, and M. Magdon-Ismail, in *Social Computing (SocialCom), 2010 IEEE Second International Conference on* (IEEE, 2010) pp. 303–308.

[23] M. Rosvall and C. T. Bergstrom, PloS one **5**, e8694 (2010).

[24] L. v. d. Maaten and G. Hinton, Journal of machine learning research **9**, 2579 (2008).

[25] B. Karrer, E. Levina, and M. E. Newman, Physical review E **77**, 046119 (2008).

[26] A. Lancichinetti, S. Fortunato, and F. Radicchi, Physical review E **78**, 046110 (2008).

[27] D. Edler and M. Rosvall, *The Infomap Software Package* (2019).

[28] T. P. Peixoto, *The Infomap Software Package* (2014).

[29] R. Bernardo-Madrid, J. Calatayud, M. González-Suarez, M. Rosvall, P. M. Lucas, M. Rueda, A. Antonelli, and E. Revilla, Ecology Letters (2019).

[30] T. P. Peixoto, Physical Review X **4**, 011047 (2014).

[31] M. Kottek, J. Grieser, C. Beck, B. Rudolf, and F. Rubel, Meteorologische Zeitschrift **15**, 259 (2006) .