nature portfolio

Peer Review File

Mapping bipartite networks into multidimensional hyperbolic spaces

Corresponding Author: Professor Marian Boguña

This file contains all reviewer reports in order by version, followed by all author rebuttals in order by version.

Version 0:

Reviewer comments:

Reviewer #1

(Remarks to the Author)
The manuscript extends previous work on generating bipartite networks in hyperbolic space by relaxing the two-dimensional constraint of the bipartite S^1/H^2 model. Specifically, it introduces the S^D/H^{D+1} model, which operates in hyperbolic spaces of dimension D+1>=2. At the center of the contribution lies the B-Mercator embedding algorithm, which can be regarded as an adaptation of the earlier D-Mercator embedding method to bipartite networks. The algorithm combines Laplacian Eigenmaps with likelihood maximization under the corresponding hyperbolic network generation model. While earlier bipartite hyperbolic embedding techniques were designed to produce strictly two-dimensional layouts, the B-Mercator approach generalizes this setting by treating the dimensionality as a tunable parameter. The authors validate the performance of B-Mercator on both synthetic and real datasets. Although the adjustable embedding dimension appears to be the main innovation, the necessity of controlling the embedding dimension is not sufficiently motivated in the text, the range of dimensions explored remains rather limited (from 2D to 6D), and in most measurements the results are only presented for two- and three-dimensional embeddings.

Regarding the abstract, since two-dimensional geometric models of bipartite networks were already described in Refs. [12], [27] (using the S^1 representation) and [15] (describing the connection between the S^1 and H^2 representations), I would suggest mentioning in the third sentence that the main innovation of the proposed S^D/H^{D+1} model lies in the tunable number of dimensions.

In the Introduction, it would be helpful to emphasize why it is important that the proposed model and embedding method are multi-dimensional. What are the limitations of restricting network generation and embedding to two-dimensional spaces? It could also be made more explicit what the authors intended to achieve with the new model in comparison to previous bipartite geometric models. For example, such clarification might be added after the following sentence: “In recent years, network geometry [21] ... has also been extended to bipartite settings, proposing that nodes of both types can indeed lie in a shared latent space with connection probabilities governed by their mutual distances [12, 27]”. In addition, the final paragraph of the Introduction might note that not only one-mode projections have been embedded previously: Ref. [12], for instance, already provided actual bipartite embeddings, though only in two dimensions.

I find Section I/A relatively difficult to follow, as the descriptions of embedding and network generation are somewhat intertwined. It might also be helpful to add a brief statement about how the network nodes are arranged angularly, rather than requiring the reader to search for this information in the Methods section.

In Section I/B, the description of how the reproducibility of the topological properties was tested (also in the caption of Fig. S6 of the Supplementary Information) is rather hard to read.

In Fig. 2, I find it confusing that panels a–b (concerning the S^1/H^2 model) show only a single (angular) coordinate, while panels c–d (concerning the S^2/H^3 model) show three Cartesian coordinates. This creates the impression that panels a–b rely on the S^1 representation but panels c–d on the H^3 representation, even though the caption refers only to the S^1 and S^2 terms. I would recommend avoiding mixing different representations within a single figure and either showing only angular coordinates in both cases, or show Cartesian coordinates in both. Furthermore, in the caption of Fig. 2, the authors state that they applied a transformation to minimize the average angular distance between the original and inferred coordinates “since the inferred coordinates for D=2 might be rotated”. Was this transformation not used for D=1? If not, why?

Also, I believe that instead of citing Section III (“Algorithm’s time complexity”) of the Supplementary Information of Ref. [30], Section IV (“Transformation of unit D-sphere”) would be more appropriate.
In addition, Fig. S2 of the Supplementary Information appears to duplicate panels a–b of Fig. 2 in the main text. This could be noted in the caption of Fig. S2, or alternatively, Fig. S2 could present results for a different parameter setting (similar to how Fig. S3 differs from Fig. 2c–d). In general, the rationale behind the parameter choices of the experiments comparing true and inferred node coordinates is not made clear. For example, in Fig. 2 not only the number of dimensions is increased between panels a–b and c–d, but in the meantime, the temperature is decreased, strengthening the coupling between the generated network topology and the underlying metric space. It therefore remains unclear whether the higher-dimensional node arrangements of the network generation process are easier to estimate by the embedding, or whether the observed effect is due to changes in other parameters. Similarly, in Figs. S2–S4 the dimensionality varies, but other parameters also change simultaneously, making it difficult to isolate the effect of D itself.

In Fig. S5 of the Supplementary Information, the use of D_in and D_out in the caption but only D (without subscript) in the figure is confusing.

With respect to greedy routing (Section I/C, the end of Section I/D, and Supplementary Sections 5–6), my main concern is that its importance is not validated on real networks. The authors “introduce bipartite greedy routing (BGR) as a practical tool to infer the network’s effective dimension”, stating that the performance of greedy routing on a bipartite graph can be considered as an indicator “effectively allowing us to determine the dimension that best captures its structure”. On synthetic networks generated by the S^D/H^{D+1} model, the results indeed show that “the performance of BGR is optimal when B-Mercator is used with the same dimension that was used to generate the network”. In Table S2 of the Supplementary Information, three real bipartite networks are examined from the viewpoint of greedy routing; however, the corresponding choice of the number of embedding dimensions has not been underpinned by futher investigations. Section I/D in the main text focuses on the setting D=1 for the three real networks. Figures S15, S18 and S22 of the Supplementary Information examine the reproducibility of the topological properties of the three real networks based on their multi-dimensional embeddings (using D = 1, 2, 3 and 4), but while Table S2 identifies D=1 as the optimal setting for the metabolic and flavor networks, Figs. S18 and S22 do not show a clear superiority of this setting. Moreover, although the first paragraph of Section I/D mentions link prediction, such task is not examined for these three real networks, whereas for the real datasets studied e.g. from the viewpoint of link prediction in Section I/E, no results are reported regarding the BGR-based choice of D (and only D=1 and D=2 is tested). Overall, while stating that BGR can be used to determine the optimal embedding dimension, the manuscript does not seem to actually verify that the BGR-based choice of D stands out among the possible settings for real networks.

In addition, it is not clear why the four variants of greedy routing (with source-target node pairs of type A–A, A–B, B–A, and B–B) are treated separately. Namely, Fig. 3b reports the proportion of successful greedy routes specifically with type-A source nodes and type-B target nodes and Supplementary Table S2 deals only with the A-A variant, even though both the success rate and the mean stretch could have been evaluated considering all four types of greedy paths simultaneously. For synthetic networks, Figs. S11-12 show that the performance of the four different greedy routing variants are similar (according to the end of Section I/C: the "results are consistent across all variants of BGR"), but is this also true for the real networks examined in Supplementary Table S2? What was the rationale for restricting Supplementary Table S2 to the A–A variant?

In Section I/E, it would be helpful to clarify already in the first paragraph that node features and node labels are different concepts. In the second paragraph, selecting “embedding dimensions D=1 and D=2” should be justified. Since all the reference methods are evaluated up to 16 dimensions, why were the hyperbolic cases restricted to 2D and 3D? In the fourth paragraph (and in the caption of Fig. 6 and in the Supplementary Information), I believe the intended training–test ratio was 80/20 rather than 20/80. If this was not a typographical error, the authors should explain the reason for such an unusual choice.

While link prediction was performed using Euclidean distances for Euclidean methods and hyperbolic distances for hyperbolic methods, in the node classification task hyperbolic methods were evaluated using angular distances (which is reasonable as here the role of the angular coordinates is relatively clearly "separated" from that of the radial coordinates, but this remains unexplained here in the text), whereas Euclidean methods were evaluated using Euclidean distances. Was it tested whether Euclidean distance is indeed the best choice for node classification in Euclidean embeddings, or could one justify theoretically that angular distances reflect the node classes more poorly compared to Euclidean distances?

In Table S3 of the Supplementary Information, I believe that the value N_I for the Wisconsin dataset should be 5 rather than 7, since the main text states that there are five categories here. In the caption of Fig. S31, the phrasing “The remaining existing links form the training set” is confusing; “training network” (as in the main text) would be clearer.

In the final paragraph of Section I/E, the authors state that “B-Mercator in low dimensions (D=1 and D=2) consistently ranks among the top approaches for the NC task and outperforms all feature-based methods for the LP task, for which D-Mercator is the best method in dimension D=1”, and emphasize the “efficiency of our low-dimensional embeddings compared to high-dimensional alternatives”. However, it is not clear from the results whether B-Mercator’s performance might improve further by increasing the number of embedding dimensions. Given the size of the datasets, embeddings up to 16 dimensions cannot really be considered “high-dimensional”; even 16-dimensional node coordinate vectors represent a substantial dimension reduction compared to the adjacency matrixes. Though for graph visualization, 2D and 3D embeddings are obviously desirable, in machine learning tasks such as link prediction and node classification, why is it preferred to use e.g. a 3-dimensional embedding over 4D, 8D, or 16D ones? According to Fig. 6d, the 4-dimensional MUSAE embedding seems

to perform the best, and its rankings are much better than for the 2-dimensional MUSAE embeddings. Why is it not expected that the performance of B-Mercator also could be increased with further increase in D, or why were higher values of D (that were tested for the reference methods) not tested for B-Mercator?

In Fig. 6b, the legend would avoid overlapping with the data if placed in the lower left corner. In addition, although the caption refers to diagonal hatching, I do not see it in Fig. 6a–b (unlike the analogous Supplementary figures). Besides, the meaning of the star notation for MUSAE and FN in panels a, b and d of Fig. 6 is not explained.

In Section II (Discussion), the authors state that “B-Mercator offers a powerful and interpretable way to capture community structure, hierarchical organization, and topological relationships”, and mention “studying network hierarchies” as a potential application. Since the topic of hierarchy was not discussed earlier in the manuscript, its appearance in the Discussion seems unmotivated. Moreover, the final sentences (B-Mercator provides “substantially more accurate analyses than those based on corresponding one-mode projections. Overall, B-Mercator represents a significant advancement in bipartite network analysis”) give the impression that the main contribution is the ability to embed bipartite graphs directly rather than their one-mode projections. However, as the manuscript itself already cites, bipartite hyperbolic embedding methods already exist in the literature, albeit restricted to two dimensions.

At the end of Section III/A, I do not understand the sentence mentioning the hyperboloid model. The hyperbolic distance approximation in Equation (5) is usually associated with the native ball representation of the hyperbolic space, and hyperboloids (instead of multi-dimensional balls) were not mentioned earlier in the manuscript. In addition, a reference would also be useful for the approximation formula in the final sentence of Section III/A.

Finally, Section III/B relies heavily on Ref. [30], making it difficult to reproduce the B-Mercator algorithm from this manuscript alone. Some choices would benefit from additional explanation. Why must the parameter beta_b lie in the interval (D, 2D)? Why was Laplacian Eigenmaps specifically chosen rather than another dimensionality reduction/matrix factorization method? Why is a uniform angular distribution imposed after Laplacian Eigenmaps at D=1? What happens if multiple nodes have the same degree—is B-Mercator’s output deterministic? Furthermore, the notations “{k_{A,i}, i, … , N_A}” and “{k_{B,i}, i, … , N_B}” are unclear, and Laplacian Eigenmaps above Equation (6) is missing a reference.

All things considered, I find the reasoning of the manuscript often incomplete and the main innovations not sufficiently emphasized. It would be important to clarify the role of the number of dimensions in the proposed embedding method and, if applicable, any limits to the tunability of D.

Reviewer #2

(Remarks to the Author)
Dear authors,

I found your manuscript very interesting and pleasant to read; therefore, I recommend it for publication in Communication Physics.

However, there is one main concern on my side that has to be addressed, alongside other minor comments that you can feel free to ignore. You can find all details in the attached document.

Wishing you all the best.

Version 1:

Reviewer comments:

Reviewer #1

(Remarks to the Author)
I thank the authors for addressing my questions, I am satisfied with the answers and the modifications of the manuscript. I have no further comments, I deem the manuscript suitable for publication in its present form.

Reviewer #2

(Remarks to the Author)
Dear authors,

I am satisfied with the answers to my previous comments contained in the rebuttal letter, as well as with the actions taken to improve the manuscript.

Best,
Riccardo

Resubmission of manuscript COMMSPHYS-25-1148-T
Mapping bipartite networks into multidimensional hyperbolic spaces

Robert Jankowski, Roya Aliakbarisani, M. Ángeles Serrano, Marián Boguñá

Reviewer #1

Comment 1.0.: The manuscript extends previous work on generating bipartite networks in hyperbolic space by relaxing the two-dimensional constraint of the bipartite S^1/H^2 model. Specifically, it introduces the S^D/H^{D+1} model, which operates in hyperbolic spaces of dimension D+1>=2. At the center of the contribution lies the B-Mercator embedding algorithm, which can be regarded as an adaptation of the earlier D-Mercator embedding method to bipartite networks. The algorithm combines Laplacian Eigenmaps with likelihood maximization under the corresponding hyperbolic network generation model. While earlier bipartite hyperbolic embedding techniques were designed to produce strictly two-dimensional layouts, the B-Mercator approach generalizes this setting by treating the dimensionality as a tunable parameter. The authors validate the performance of B-Mercator on both synthetic and real datasets. Although the adjustable embedding dimension appears to be the main innovation, the necessity of controlling the embedding dimension is not sufficiently motivated in the text, the range of dimensions explored remains rather limited (from 2D to 6D), and in most measurements the results are only presented for two- and three-dimensional embeddings.

Comment 1.1.: Regarding the abstract, since two-dimensional geometric models of bipartite networks were already described in Refs. [12], [27] (using the S^1 representation) and [15] (describing the connection between the S^1 and H^2 representations), I would suggest mentioning in the third sentence that the main innovation of the proposed S^D/H^{D+1} model lies in the tunable number of dimensions.

Reply: We thank the Referee for their constructive interaction, which has helped us improve the presentation of our work.

We should clarify that using arbitrary dimensions is, albeit important, but not the main motivation for our work. Please note that Refs. [12], [27], and [15] do not present embeddings of bipartite networks but first approximations to the problem. Specifically:

- In [12], we first embed the one-mode projection on one of the node types of the bipartite network using the $\mathbb{S}^1$ model. We then fix the coordinates of these nodes and, keeping them fixed, infer the coordinates of the nodes of the other type. However, even if the bipartite network follows the bipartite-$\mathbb{S}^1$ model, its one-mode projection does not follow the $\mathbb{S}^1$ model, because many correlated links cannot be explained by $\mathbb{S}^1$. Therefore, this method is only a coarse approximation that we used in the past, simply because a bipartite embedder was not yet available.
- In [15], we deal with structured datasets of nodes and features, with a unipartite network between nodes and a bipartite network between nodes and their features. We embed the **unipartite** network using the $\mathbb{S}^1$ model and assume that the coordinates of these nodes are the same in the bipartite network of nodes and features. We then keep the nodes' coordinates fixed and infer the coordinates of the features. However, in general, given a bipartite network, there need not exist a corresponding unipartite network that can be embedded with the $\mathbb{S}^1$ model; and even if such a network exists, there is no guarantee that the angular coordinates of nodes in the unipartite and bipartite embeddings coincide. Thus, while this approach may work in some cases, it is not a general embedding method for bipartite networks.

- Finally, in [27], the authors do not embed bipartite networks; they compute a coarse estimate of angular separation between nodes based on common neighbors. Again, this is not an embedding method for bipartite networks.

Thus, our work is not merely a generalization of existing methods to arbitrary dimensions; to our knowledge, B-Mercator is the first model-based embedding tool for bipartite networks.

Regarding the choice of dimension: because our model is effectively hyperbolic, networks can be embedded in very low dimensions with minimal distortion compared with Euclidean embeddings. This explains why Euclidean machine-learning methods typically require high-dimensional spaces. By contrast, prior work shows that real complex networks can be faithfully embedded in low-dimensional hyperbolic spaces, up to about $D = 10$. Note that in our geometric models the densities of triangles and higher-order cycles decrease as D increases; if the dimension is too large, the embedding will not produce enough transitivity relations to match the empirical ones [Nature Communications 13, 6096 (2022), Nature Communications 14, 7585 (2023), npj Complexity 1, 13 (2024)]. Some networks do benefit from a modest increase in dimension, however. We therefore select the embedding dimension by maximizing greedy-routing performance, a criterion that works well on synthetic networks, and apply the same procedure to the real networks analyzed here.

In any case, for a given real network with an observed bipartite clustering coefficient, there is an upper bound for the dimension that can be used. The limitation arises because the maximum bipartite clustering coefficient from the geometric model –achieved when $\beta_b \to \infty$– decreases with dimensionality (Figure R1). This is so because the distribution of distances between pairs of randomly chosen points in the space becomes highly picked as the dimension increases. Consequently, for networks with very large bipartite clustering, if the inferred dimension is set too high no value of β_b can reproduce the observed clustering.

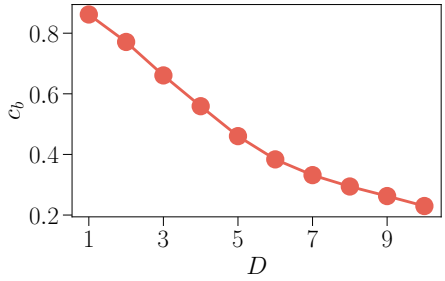

Figure R1: Relationship between the model's dimension and the bipartite clustering coefficient. Parameters: $N_A = 2000, N_B = 1000, \beta_b = 100D, \gamma_A = \gamma_B = 2.5, \langle k_A \rangle = 20$

Action taken: To clarify our contribution, we have added a discussion about these approaches in the introduction of the manuscript.

Comment 1.2.: In the Introduction, it would be helpful to emphasize why it is important that the proposed model and embedding method are multi-dimensional. What are the limitations of restricting network generation and embedding to two-dimensional spaces? It could also be made more explicit what the authors intended to achieve with the new model in comparison to previous bipartite geometric models. For example, such clarification might be added after the following sentence: "In recent years, network geometry [21] ... has also been extended to bipartite settings, proposing that nodes of both types can indeed lie in a shared latent space with connection probabilities governed by their mutual distances [12, 27]". In addition, the final paragraph of the Introduction might note that not only

one-mode projections have been embedded previously: Ref. [12], for instance, already provided actual bipartite embeddings, though only in two dimensions.

Reply: As discussed above, real networks can be embedded in low-dimensional hyperbolic spaces of dimension $D+1$. In the hyperbolic representation, the radial coordinate encodes node popularity, whereas the D-dimensional subspace encodes node similarity. For many networks, using $D=1$ suffices to represent the data. However, in some networks the similarity subspace is better represented in higher dimensions. See, for instance, Fig. 4 in [Nature Communications 14, 7585 (2023)], where two separate communities in the similarity space are mixed when represented with $D=1$, yet appear clearly separated with $D=2$.

As for the embedding mentioned by the Referee, in [12], we embedded the one-mode projection of one node type with the $\mathbb{S}^1$ model, fixed those coordinates, and then inferred the coordinates of the other type. But even if the bipartite network follows the bipartite-$\mathbb{S}^1$ model, its one-mode projection does not follow $\mathbb{S}^1$, as many correlated links are not accounted for. Thus, this approach is only a coarse approximation –one we used before a bipartite embedder was available.

Action taken: We have added a discussion in the introduction to emphasize the importance of selecting the appropriate dimension.

Comment 1.3.: I find Section I/A relatively difficult to follow, as the descriptions of embedding and network generation are somewhat intertwined. It might also be helpful to add a brief statement about how the network nodes are arranged angularly, rather than requiring the reader to search for this information in the Methods section.

Reply and action taken: We thank the Referee for this comment. We have rewritten this section so that now the explanation of the bipartite-$\mathbb{S}^1$ model and B-Mercator are separated.

Comment 1.4.: In Section I/B, the description of how the reproducibility of the topological properties was tested (also in the caption of Fig. S6 of the Supplementary Information) is rather hard to read.

Reply and action taken: We have rewritten this section to clarify it and make it more self-contained.

Comment 1.5.: In Fig. 2, I find it confusing that panels a–b (concerning the S^1/H^2 model) show only a single (angular) coordinate, while panels c–d (concerning the S^2/H^3 model) show three Cartesian coordinates. This creates the impression that panels a–b rely on the S^1 representation but panels c–d on the H^3 representation, even though the caption refers only to the S^1 and S^2 terms. I would recommend avoiding mixing different representations within a single figure and either showing only angular coordinates in both cases, or show Cartesian coordinates in both.

Reply: We thank the Referee for this comment. In the bipartite-$\mathbb{S}^2$ model the similarity space is represented as a sphere. In Fig. 2, we plotted it in the Cartesian coordinates, however, we agree that spherical coordinates might be more suitable.

Actions: We updated Fig. 2 by plotting the coordinates in the spherical representations. Additionally, we updated Supplementary Figures S2-S4 in a similar manner.

Furthermore, in the caption of Fig. 2, the authors state that they applied a transformation to minimize the average angular distance between the original and inferred coordinates "since the inferred coordinates for D=2 might be rotated". Was this transformation not used for D=1? If not, why?

Reply and action taken: Yes, we also applied this transformation in the case D=1, although in this case the procedure is simpler as it only involves a single rotation. We have removed the "$D = 2$" from the caption of the figure, so that now the sentence applies to both cases.

Also, I believe that instead of citing Section III ("Algorithm's time complexity") of the Supplementary Information of Ref. [30], Section IV ("Transformation of unit D-sphere") would be more appropriate.

Reply and action taken: The Referee is absolutely right. We have corrected this mistake in the new version of the manuscript.

In addition, Fig. S2 of the Supplementary Information appears to duplicate panels a–b of Fig. 2 in the main text. This could be noted in the caption of Fig. S2, or alternatively, Fig. S2 could present results for a different parameter setting (similar to how Fig. S3 differs from Fig. 2c–d).

Reply and action taken: We thank the Referee for bringing this to our attention. We have updated Supplementary Figures S2-S3 to show the quality of embeddings for networks with different properties.

In general, the rationale behind the parameter choices of the experiments comparing true and inferred node coordinates is not made clear. For example, in Fig. 2 not only the number of dimensions is increased between panels a–b and c–d, but in the meantime, the temperature is decreased, strengthening the coupling between the generated network topology and the underlying metric space. It therefore remains unclear whether the higher-dimensional node arrangements of the network generation process are easier to estimate by the embedding, or whether the observed effect is due to changes in other parameters. Similarly, in Figs. S2–S4 the dimensionality varies, but other parameters also change simultaneously, making it difficult to isolate the effect of D itself.

Reply: Actually, the relevant parameter to quantify the coupling with the metric space is the ratio between β_b and the dimension D. This is because the model undergoes a topological phase transition at $\beta_b = D$ between a clustered phase, for $\beta_b > D$, and an unclustered one below this critical value. This is the reason why when we increase the dimension we also increase the value of β_b.

Action taken: To isolate the impact of dimensionality on the estimation of the nodes' coordinates, in new Fig. 2, we now fix the models' parameters, varying only the dimension. Overall, we observe a similar pattern, with most points lying on the diagonal and some outliers. We further investigated these outliers, as suggested in Referee 2, Comment 2.2, and found that they are low-degree nodes (see new Supplementary Figs. S5–S6). Removing the lowest-degree nodes improves the estimation accuracy. Accuracy also increases with the geometric coupling parameter β_b (see new Supplementary Fig. S6).

Comment 1.6.: In Fig. S5 of the Supplementary Information, the use of D_{in} and D_{out} in the caption but only D (without subscript) in the figure is confusing.

Reply and action taken: We agree that the previous labels were confusing and have updated the figure accordingly. Each panel corresponds to a fixed input dimension D_{in}, and colors indicate the output (embedded) dimension D_{out}.

Comment 1.7.: With respect to greedy routing (Section I/C, the end of Section I/D, and Supplementary Sections 5–6), my main concern is that its importance is not validated on real networks. The authors "introduce bipartite greedy routing (BGR) as a practical tool to infer the network's effective dimension", stating that the performance of greedy routing on a bipartite graph can be considered as an indicator "effectively allowing us to determine the dimension that best captures its structure". On

synthetic networks generated by the S^D/H^{D+1} model, the results indeed show that "the performance of BGR is optimal when B-Mercator is used with the same dimension that was used to generate the network". In Table S2 of the Supplementary Information, three real bipartite networks are examined from the viewpoint of greedy routing; however, the corresponding choice of the number of embedding dimensions has not been underpinned by futher investigations. Section I/D in the main text focuses on the setting D=1 for the three real networks. Figures S15, S18 and S22 of the Supplementary Information examine the reproducibility of the topological properties of the three real networks based on their multi-dimensional embeddings (using D = 1, 2, 3 and 4), but while Table S2 identifies D=1 as the optimal setting for the metabolic and flavor networks, Figs. S18 and S22 do not show a clear superiority of this setting. Moreover, although the first paragraph of Section I/D mentions link prediction, such task is not examined for these three real networks, whereas for the real datasets studied e.g. from the viewpoint of link prediction in Section I/E, no results are reported regarding the BGR-based choice of D (and only D=1 and D=2 is tested). Overall, while stating that BGR can be used to determine the optimal embedding dimension, the manuscript does not seem to actually verify that the BGR-based choice of D stands out among the possible settings for real networks.

Reply: We thank the Referee for this comment. As the Referee noted, the Supplementary Figures showing the validation of the topological properties of the real networks exhibit good agreement not only for the optimal dimension but also for the other tested dimensions. However, these are only local properties that cannot detect the large scale organization of the network. This observation is consistent with previous results reported in Ref. [30]. For example, in Fig. 4 of [Nature Communications 14, 7585 (2023)] the best dimension was found to be $D = 2$. However, as shown in Supplementary Figure S27 of that work, the validation curves remain similar across all dimensions. Therefore, it is only at the mesoscopic and global scales, through the analysis of community structure and navigability, that the optimal dimension can be clearly identified.

Regarding the optimal dimension of the real bipartite networks, the Metabolic and Flavor datasets achieve the highest greedy routing success rate at $D = 1$. Therefore, we focus primarily on the visualization in one dimension, while also providing a comparison for $D = 2$. For the Unicodelang dataset, the success rate remains similar across all dimensions. Therefore, we selected $D = 1$ for visualization and interpretation purposes. This choice enabled us to introduce diversity measures (Fig. 5), which can be clearly observed from the embedding (Fig. 4).

The link prediction task in Section I/E is performed on unipartite networks. The three datasets used in Section I/D, however, are bipartite networks. In this case, predicting a link corresponds to predicting a relation between type-A and type-B nodes, for instance, in the Unicodelang dataset, between a given country and a given language. This is indeed an interesting direction, commonly explored in recommendation systems, and one that we plan to investigate in future work.

Regarding the node classification and link prediction tasks, as we mention in Comment 1.11, the performance would likely improve if each network were embedded with B-Mercator at its optimal dimensionality. Here we intentionally analyze the simplest cases, $D = 1$ and $D = 2$ to demonstrate that our method remains competitive with alternatives that use much higher dimensions. As the Referee notes, we also aim to visualize and interpret the embedding (Fig. 6c). Standard ML pipelines often require a second dimensionality-reduction step (e.g., t-SNE or PCA) for visualization. In contrast, our approach achieves strong predictive performance while remaining directly interpretable.

Comment 1.8.: In addition, it is not clear why the four variants of greedy routing (with source-target node pairs of type A–A, A–B, B–A, and B–B) are treated separately. Namely, Fig. 3b reports the proportion of successful greedy routes specifically with type-A source nodes and type-B target nodes and Supplementary Table S2 deals only with the A-A variant, even though both the success rate and

the mean stretch could have been evaluated considering all four types of greedy paths simultaneously. For synthetic networks, Figs. S11-12 show that the performance of the four different greedy routing variants are similar (according to the end of Section I/C: the "results are consistent across all variants of BGR"), but is this also true for the real networks examined in Supplementary Table S2? What was the rationale for restricting Supplementary Table S2 to the A–A variant?

Reply: We thank the Referee for this question. Indeed, in the synthetic networks, the choice of the BGR variant does not affect the final conclusions regarding the optimal dimension. Therefore, we initially used a single BGR variant to determine the optimal dimension in real networks. To improve clarity, we have now included the results for other BGR variants in new Supplementary Tables S3–S5. Overall, the optimal dimension remains consistent across variants, with only minor variations in success rates.

Action taken: We have added Supplementary Tables S3–S5 presenting the BGR results for other variants in real networks.

Comment 1.9.: In Section I/E, it would be helpful to clarify already in the first paragraph that node features and node labels are different concepts. In the second paragraph, selecting "embedding dimensions D=1 and D=2" should be justified. Since all the reference methods are evaluated up to 16 dimensions, why were the hyperbolic cases restricted to 2D and 3D? In the fourth paragraph (and in the caption of Fig. 6 and in the Supplementary Information), I believe the intended training–test ratio was 80/20 rather than 20/80. If this was not a typographical error, the authors should explain the reason for such an unusual choice.

Reply and action taken: We have modified the sentence in the first paragraph to clarify that nodes have assigned labels besides features. We have also added a sentence to motivate our choice of dimension when using our models.

We used a 20/80 train/test split to stress the classifier under label scarcity, i.e., to evaluate how well the embeddings support classification when only a small fraction of nodes are labeled. As suggested by the Referee, we also performed node classification task on the Wisconsin dataset with an 80/20 split (Figure R2 and new Supplementary Fig. S33). The performance trends and method ranking remained consistent with the low-label regime. This indicates that our conclusions are not an artifact of the specific label budget.

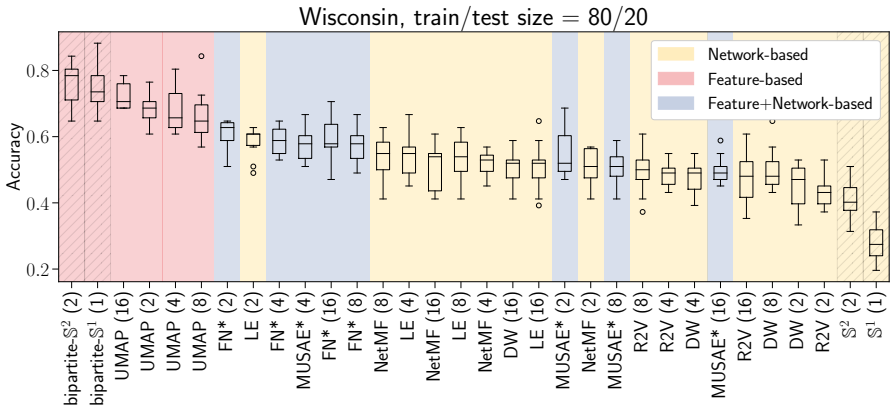

Figure R2: Accuracy of the node classification task for the Wisconsin dataset. The train/test split is 80/20.

Comment 1.9.:

While link prediction was performed using Euclidean distances for Euclidean methods and hyperbolic distances for hyperbolic methods, in the node classification task hyperbolic methods were evaluated using angular distances (which is reasonable as here the role of the angular coordinates is relatively clearly "separated" from that of the radial coordinates, but this remains unexplained here in the text), whereas Euclidean methods were evaluated using Euclidean distances. Was it tested whether Euclidean distance is indeed the best choice for node classification in Euclidean embeddings, or could one justify theoretically that angular distances reflect the node classes more poorly compared to Euclidean distances?

Reply: In the machine learning community Euclidean distance and cosine similarity are the standard choices. We follow common practice and use Euclidean distance in the classifier. Please note that k-means also defaults to Euclidean distance. Euclidean distance can degrade in very high dimensions (curse of dimensionality). Here the embedding dimension is at most 16, which should mitigate this issue.

To assess cosine similarity, we repeated the node-classification experiments by switching the classifier's metric from Euclidean to cosine for the Euclidean-based methods. Figure R3 reports the accuracies (also new Supplementary Figure S34). Compared with Figure 5 in the main text, the methods perform on par with Euclidean distance with a slight change in methods' ranking.

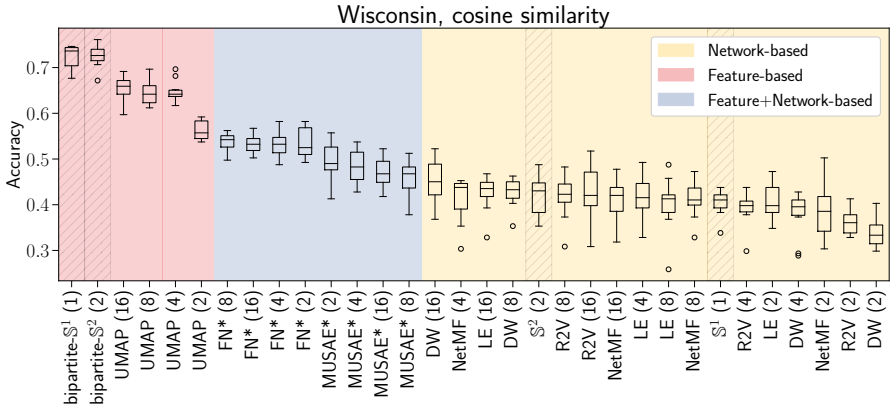

Figure R3: Accuracy of the node classification task for the Wisconsin dataset. For Euclidean-based methods we used cosine similarity as a metric in the classifier.

Comment 1.10.:

In Table S3 of the Supplementary Information, I believe that the value N_l for the Wisconsin dataset should be 5 rather than 7, since the main text states that there are five categories here. In the caption of Fig. S31, the phrasing "The remaining existing links form the training set" is confusing; "training network" (as in the main text) would be clearer.

Reply and action taken: We thank the Reviewer for noticing this typo. Indeed, there are 5 classes in the Wisconsin dataset. We have updated a new Supplementary Table 6, as well as the Supplementary Section 8, with a description of the datasets. We have also changed the caption of Supplementary Figures S35 and S43 replacing "training set" by "training network".

Comment 1.11.:

In the final paragraph of Section I/E, the authors state that "B-Mercator in low dimensions (D=1 and D=2) consistently ranks among the top approaches for the NC task and outperforms all feature-based methods for the LP task, for which D-Mercator is the best method in dimension D=1", and emphasize the "efficiency of our low-dimensional embeddings compared to high-dimensional alternatives". However, it is not clear from the results whether B-Mercator's performance might improve further by increasing the number of embedding dimensions. Given the size of the datasets, embeddings up to 16 dimensions cannot really be considered "high-dimensional"; even 16-dimensional node coordinate vectors represent a substantial dimension reduction compared to the adjacency matrixes. Though for graph visualization, 2D and 3D embeddings are obviously desirable, in machine learning tasks such as link prediction and node classification, why is it preferred to use e.g. a 3-dimensional embedding over 4D, 8D, or 16D ones? According to Fig. 6d, the 4-dimensional MUSAE embedding seems to perform the best, and its rankings are much better than for the 2-dimensional MUSAE embeddings. Why is it not expected that the performance of B-Mercator also could be increased with further increase in D, or why were higher values of D (that were tested for the reference methods) not tested for B-Mercator?

Reply: We thank the Referee for the comment. Performance would likely improve if each network were embedded with B-Mercator at its optimal dimensionality. Here we intentionally analyze the simplest cases, $D = 1$ and $D = 2$ to demonstrate that our method remains competitive with alternatives that use much higher dimensions. As the Referee notes, we also aim to visualize and interpret the embedding (Fig. 6c). Standard ML pipelines often require a second dimensionality-reduction step (e.g., t-SNE or PCA) for visualization. In contrast, our approach achieves strong predictive performance while remaining directly interpretable.

For the Euclidean methods, it has been observed by Gu et al. [1] that relatively low-dimensional embeddings typically suffice, with prediction performance and final loss comparable to those in much higher dimensions.

[1] Gu, W., Tandon, A., Ahn, Y. Y., & Radicchi, F. (2021). Principled approach to the selection of the embedding dimension of networks. Nature Communications, 12(1), 3772.

Action taken: We have added a paragraph descibing the choice of the dimensions in the Euclidean-based methods.

Comment 1.12.:

In Fig. 6b, the legend would avoid overlapping with the data if placed in the lower left corner. In addition, although the caption refers to diagonal hatching, I do not see it in Fig. 6a–b (unlike the analogous Supplementary figures). Besides, the meaning of the star notation for MUSAE and FN in panels a, b and d of Fig. 6 is not explained.

Reply and action taken: We moved the legend of Fig. 6b to the lower-left corner. We also added diagonal hatching to Figs. 6a and 6b. We marked MUSAE and FN with a star to indicate that they use both network topology and node features in the embedding. This information has been added to the caption of Fig. 6.

Comment 1.13.:

In Section II (Discussion), the authors state that "B-Mercator offers a powerful and interpretable way to capture community structure, hierarchical organization, and topological relationships", and mention "studying network hierarchies" as a potential application. Since the topic of hierarchy was not discussed earlier in the manuscript, its appearance in the Discussion seems unmotivated. Moreover,

the final sentences (B-Mercator provides "substantially more accurate analyses than those based on corresponding one-mode projections. Overall, B-Mercator represents a significant advancement in bipartite network analysis") give the impression that the main contribution is the ability to embed bipartite graphs directly rather than their one-mode projections. However, as the manuscript itself already cites, bipartite hyperbolic embedding methods already exist in the literature, albeit restricted to two dimensions.

Reply and action taken: We agree with the Referee that hierarchy was not discussed in the manuscript. However, hyperbolic geometry shares tree-like topological properties (e.g., exponential growth of volume with radius), so it naturally fits hierarchical data. We have added a sentence in the introduction to mention our previous work on hierarchies in complex unipartite networks (Ortiz et. al., 2020).

Ortiz, E., García-Pérez, G., & Serrano, M. Á. (2020). Geometric detection of hierarchical backbones in real networks. Physical Review Research, 2(3), 033519.

Comment 1.14.:

At the end of Section III/A, I do not understand the sentence mentioning the hyperboloid model. The hyperbolic distance approximation in Equation (5) is usually associated with the native ball representation of the hyperbolic space, and hyperboloids (instead of multi-dimensional balls) were not mentioned earlier in the manuscript. In addition, a reference would also be useful for the approximation formula in the final sentence of Section III/A.

Reply and action taken: The hyperboloid model (or similarly the native representation) of the hyperbolic space is one of the many possible representations of the hyperbolic space, the one for which the formula for the distance holds. We have changed the sentence from "hyperboloid model" to "native representation". We have also added a new reference [42] where the approximation of the hyperbolic distance is explained.

Comment 1.15.:

Finally, Section III/B relies heavily on Ref. [30], making it difficult to reproduce the B-Mercator algorithm from this manuscript alone. Some choices would benefit from additional explanation. Why must the parameter β_b lie in the interval (D, 2D)? Why was Laplacian Eigenmaps specifically chosen rather than another dimensionality reduction/matrix factorization method? Why is a uniform angular distribution imposed after Laplacian Eigenmaps at D=1? What happens if multiple nodes have the same degree—is B-Mercator's output deterministic? Furthermore, the notations "$k_{A,i}, i, \ldots, N_A$" and "$k_{B,i}, i, \ldots, N_B$" are unclear, and Laplacian Eigenmaps above Equation (6) is missing a reference.

Reply:

The β_b should be larger than D to be in the geometric region. B-Mercator estimates β_b, and we think that using an initial guess in the interval $(D, 2D)$ is reasonable. The exact choice of this initialization interval is not very relevant, since β_b is subsequently updated and inferred during optimization. Please notice that this step is clearly explained in the Sec III/B, where we write:

After the hidden degrees for both nodes A and B are computed, the synthetic graph from bipartite-$\mathbb{S}^D/\mathbb{H}^{D+1}$ is constructed and the bipartite clustering coefficient is calculated. If the computed bipartite clustering coefficient deviates from that of the original network, $\bar{c}_b$, the value of β_b is adjusted. Then, the process is repeated using the current estimation of hidden degrees until a

predetermined precision is reached.

Regarding the Laplacian Eigenmaps (LE), we decided to use it since it can be adopted for our bipartite-$\mathbb{S}^D$ model directly. We can compute the expected angular distance between two nodes according to the model and place it as entries to the adjacency matrix (see Eq. 7), so that, in fact, we use a model-adjusted LE, incorporating then geometric properties in the initial step of the algorithm. Of course, we could also use other dimensionality reduction algorithms, like UMAP or t-SNE, as initial step. We leave comparison of different initialization algorithms as future work.

The uniform adjustment prevents excessively large gaps between nodes. In the bipartite setting, nodes of the same type do not experience direct "repulsion": a type-A node i interacts with a type-A node j only through shared type-B neighbors. By contrast, in the unipartite embedding same-type nodes interact directly, so no such adjustment is required.

B-Mercator is inherently stochastic, and each run may produce different but statistically equivalent embeddings. These embeddings can appear rotated or mirrored relative to one another. In practice, however, varying the random seed does not significantly affect the final conclusions. To quantify uncertainty in inferred coordinates, recent work introduced BIGUE (Lizotte et al., 2025), a Markov chain Monte Carlo (MCMC) algorithm that samples the posterior distribution of a Bayesian hyperbolic random graph model. This approach, however, has so far been developed only for unipartite networks and is computationally limited to systems with fewer than 100 nodes. Extending BIGUE to bipartite networks thus represents a promising direction for future research.

Lizotte, S., Young, J. G., & Allard, A. (2025). Symmetry-driven embedding of networks in hyperbolic space. Communications Physics, 8(1), 199.

Action taken: We have added a citation to the original Laplacian Eigenmaps article. We revised the notation for the degrees of type-A and type-B nodes for greater clarity. We also added technical details of B-Mercator in Supplementary Section 12.

Comment 1.16.:

All things considered, I find the reasoning of the manuscript often incomplete and the main innovations not sufficiently emphasized. It would be important to clarify the role of the number of dimensions in the proposed embedding method and, if applicable, any limits to the tunability of D.

Reply: We would like to emphasize that our main contributions are: (i) extending the bipartite network model to higher dimensions, and (ii) developing B-Mercator, to our knowledge, the first hyperbolic bipartite network embedder, compatible with the bipartite-$\mathbb{S}^D/\mathbb{H}^{D+1}$ model. We hope these clarifications address the Referee's main concerns and thank them for very helpful comments.

Reviewer #2

Comment 2.0.: The manuscript proposes a novel and powerful method to deal with real networked systems with underlying bipartite structure, such as author-papers, user-product, country-language, metabolites-reactions, and plant-pollinator networks. The main goal in this paper is to build a framework to analyze bipartite structures directly, instead of applying a one-mode approach that flattens the network, implying a loss of data. To achieve such a goal, the manuscript includes the following

contributions

- Definition of a bipartite network model that is embedded with hyperbolic geometry.
- Extension of D-Mercator to the bipartite model, allowing the inference of node positions. This method is tested and validated on synthetic data, showing empirically its strength.
- Evaluation of B-Mercator performance on real dataset, namely, Unicodelang, Metabolic, and Flavor.
- Application of this methodology for node classification and link prediction tasks, showing significant progress compared to the current state of the art.

The authors provide strong empirical validation of the methodology proposed in this manuscript, and the results definitely fit the interests of several different communities working on Complex Networks. Therefore, I recommend this paper for publication in Communication Physics. However, I would appreciate it if the authors could address in their manuscript one main concern, detailed below. From my side, there are also a couple of minor concerns that do not necessarily require a change in the manuscript but could be taken into consideration by the authors for improvement.

Reply: We thank the Referee for their time and effort in assessing our paper and for their report, which has helped us to broaden our perspective and focus on improving the presentation of the results. We are delighted to learn that the Reviewer considers it suitable for publication in Communication Physics.

Comment 2.1.: The main question I had throughout the entire paper was: Why should the hyperbolic paradigm be a good choice for modeling real bipartite systems? In other words, it would be nice to have some extra explanation about the reason why bipartite hyperbolic random graphs are a principled model choice, from a more theoretical point of view. As written in the paper, "Empirical analyses have shown that these four-cycles can be abundant in real bipartite networks... This finding suggests that, indeed, bipartite networks may be embedded in a metric space where the likelihood of a connection between two nodes of different types decreases with the distance separating them". As far as I know, although hyperbolic models reproduce high levels of clustering, high clustering does not necessarily imply hyperbolic geometry [1], [2]. Thus, I assume this also applies to the bipartite case

From a theoretical standpoint, standard hyperbolic models are least-biased maximum entropy ensembles, hereby explaining why they are a principled model choice. Whether or not bipartite hyperbolic models are maximum entropy ensembles as well, it would be nice to at least elaborate a little more on other model-specific properties matching with real-world networks, other than matching topological features and high clustering.

[1] Roya Aliakbarisani, Marian Boguna, and M Serrano. Clustering does not always imply latent geometry. arXiv preprint arXiv:2503.05369, 2025.

[2] Riccardo Michielan, Nelly Litvak, and Clara Stegehuis. Detecting hyperbolic geometry in networks: Why triangles are not enough. Physical Review E, 106(5):054303, 2022.

Reply:

Unipartite $\mathbb{S}^D/\mathbb{H}^{D+1}$ model gives rise to a maximum-entropy ensemble of graphs. The same argument extends to bipartite settings. In our bipartite-$\mathbb{S}^D/\mathbb{H}^{D+1}$ model, the connection probability takes the

form

$$p_{uv} = \frac{1}{1+\chi_{uv}^{\beta_b}} \tag{1}$$

and, in $\mathbb{H}^{D+1}$, equivalently

$$p_{uv} = \frac{1}{1+\exp\left[\frac{\beta_b}{2}\left(x_{uv}-\widehat{R}\right)\right]}, \tag{2}$$

where x_{uv} is the effective hyperbolic distance, β_b is the inverse temperature. These expressions arise as the maximum-entropy solution once the expected degrees in both layers and the distance kernel are fixed.

We agree that high clustering does not imply latent geometry, and that triangles—and by analogy, four-cycles—are insufficient statistics [1,2]. Accordingly, our validation does not rely solely on four-cycles. Specifically, we verified that (i) empirical connection probabilities decrease with inferred distance and follow the model's expected functional form, (ii) bipartite greedy routing (BGR) achieves maximal success at the ground-truth dimension in synthetic networks, and (iii) the inferred bipartite embeddings exhibit high predictive power and generative adequacy. Moreover, beyond clustering, the bipartite hyperbolic model has been shown to reproduce degree heterogeneity, clustering spectra, and average nearest-neighbor degree relations. Additionally, in unipartite networks highly non-trivial multiscale symmetries of real networks are reproduced by our geometric models.

We do not claim that every bipartite network is geometric. The bipartite-$\mathbb{S}^D/\mathbb{H}^{D+1}$ model gives rise to a maximum-entropy ensemble of graphs that are therefore maximally random given their specific constraints [3]. In future work, we aim to develop a framework similar to that in [1] to determine whether a given bipartite network is geometric or not.

[1] Roya Aliakbarisani, Marian Boguna, and M Serrano. Clustering does not always imply latent geometry. Phys. Rev. Lett. (in press). arXiv preprint arXiv:2503.05369, 2025.

[2] Riccardo Michielan, Nelly Litvak, and Clara Stegehuis. Detecting hyperbolic geometry in networks: Why triangles are not enough. Physical Review E, 106(5):054303, 2022.

[3] Serrano, M. Á., Boguná, M., & Sagués, F. (2012). Uncovering the hidden geometry behind metabolic networks. Molecular biosystems, 8(3), 843-850.

Comment 2.2.: In Figure 2, it is possible to observe a few outliers in the validation of B-Mercator. Even though they are less than 1% of the total number of nodes, I believe it is interesting to investigate this mismatch in more detail. I might be wrong, but my intuition suggests these outliers could consist of isolated vertices or very low degree vertices for which a "geographical triangulation" is harder to obtain. Of course, this phenomenon should increase as the average degree in the network decreases, and in that case, it could significantly lower the Pearson correlation coefficient. If the authors also believe this is true, I think it would be nice to address this limitation of B-Mercator.

Reply:

We thank the Referee for this observation. B-Mercator requires a single giant connected component, so we hypothesized that the observed outliers correspond to the lowest-degree nodes. To test this, we filtered the network by retaining only nodes with degree above a threshold and compared ground-truth versus inferred coordinates. As shown in Figure R4, the outliers are indeed the lowest-degree nodes, as suggested by the Referee, and removing them improves the accuracy of the inferred coordinates.

Estimation quality also improves with the geometric coupling parameter β_b. As shown in Figure R5, with $\beta_b = 3$ there are far fewer outliers than with $\beta_b = 1.5$, and the remaining outliers are again the nodes with the smallest degrees.

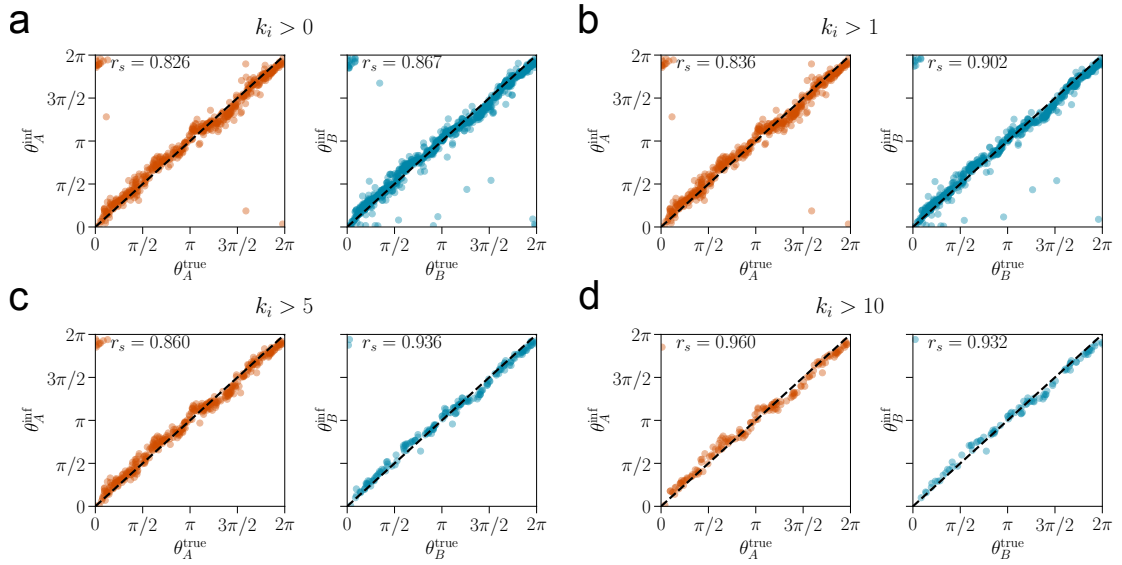

Figure R4: Relationship between the inferred and the true coordinates for $D = 1$ when we consider nodes with degree higher than (**a**) $k_i > 0$, (**b**) $k_i > 1$, (**c**) $k_i > 5$, (**d**) $k_i > 10$. Parameters: $(N_A, N_B, \gamma_A, \gamma_B, \langle k_A \rangle, \beta_b) = (500, 1000, 2.7, 2.1, 10, 1.5)$.

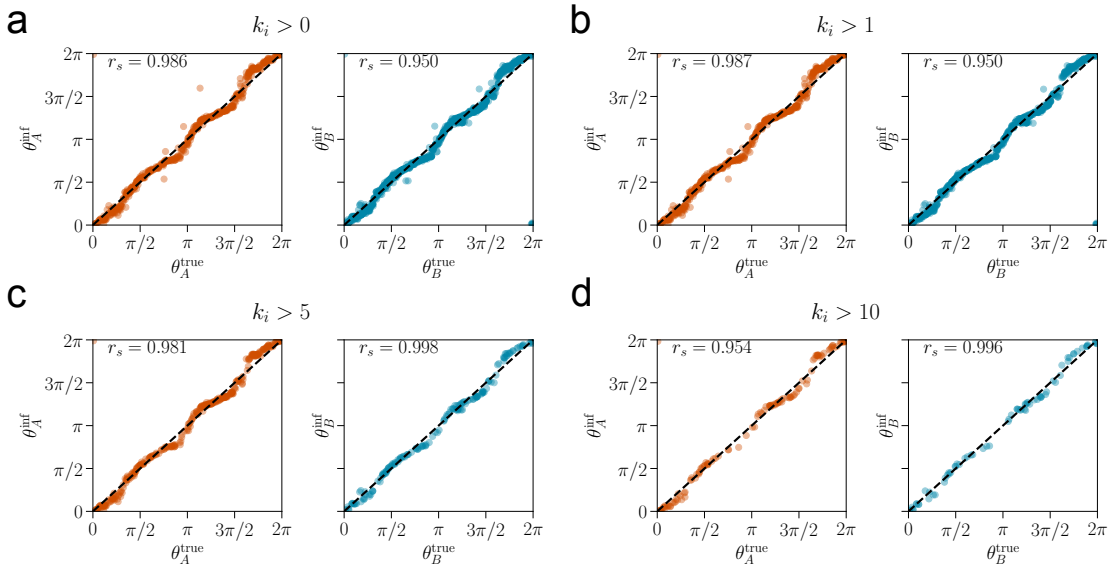

Figure R5: Relationship between the inferred and the true coordinates for $D = 1$ when we consider nodes with degree higher than (**a**) $k_i > 0$, (**b**) $k_i > 1$, (**c**) $k_i > 5$, (**d**) $k_i > 10$. Parameters: $(N_A, N_B, \gamma_A, \gamma_B, \langle k_A \rangle, \beta_b) = (500, 1000, 2.7, 2.1, 10, 3)$.

Action taken: We added two figures to the Supplementary Information.

Comment 2.3.: In Section IC, the authors state "A lower-dimensional latent space often requires fewer hops and exhibits more consistent greedy paths, thus reflecting more coherent topological structures. Conversely, if greedy routing frequently fails or requires excessive detours, it suggests a higher-dimensional or more complex geometry". I do not get exactly why this should be true, and there are no references associated. If they agree, the authors could elaborate on this or provide a reference in

support of this claim.

Reply and action taken: We agree with the Referee that this sentence was misleading so we have removed it from the manuscript.

Comment 2.4.: In Figure 5S (supplementary material), the notation in the picture for the dimensionD does not match the figure description. In the description, two different symbols are used, D_{in} and D_{out}. I suggest modifying the legend and figure axes so that it is explicit what the dimension is, either the one from the synthetic network or from the embedding through B-Mercator. Also, from this picture, it is not very clear how well the embedding retrieves the value β_b: if I understand correctly, the embedding in the correct dimension ($D_{in} = D_{out}$) should settle on the dotted black line, but that does not seem to be happening very well.

Reply: We agree that the previous labels were confusing and have updated the figure accordingly. Each panel corresponds to a fixed input dimension D_{in}, and colors indicate the output (embedded) dimension D_{out}.

We note that the inference of β_b is accurate when $D_{\text{in}} = D_{\text{out}}$, with points lying on the diagonal. The limitation arises because the maximum bipartite clustering coefficient decreases with dimensionality (Figure R6). Consequently, for networks with very large bipartite clustering, if the inferred dimension is set too high, β_b becomes non-identifiable: no value can reproduce the observed clustering, so β_b cannot be estimated.

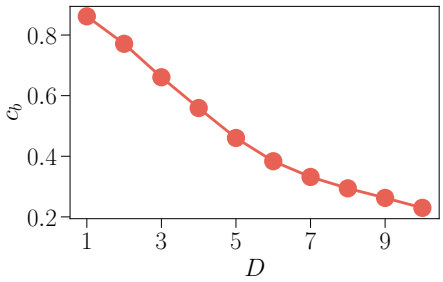

Figure R6: Relationship between the model's dimension and the bipartite clustering coefficient. Parameters: $N_A = 2000, N_B = 1000, \beta_b = 100D, \gamma_A = \gamma_B = 2.5, \langle k_A \rangle = 20$

Comment 2.5.: Throughout the text there is inconsistency with the term type-A and type-B. Sometimes the dash is there, sometimes it is not. I suggest adopting a uniform notation.

Reply and action taken: We have made the notation for type-A and type-B nodes consistent throughout the text.

Comment 2.6.: Multiple times the term network topology is used to express the connection strength (intended as the expected degree of nodes) in the network, concurrent phenomenon alongside geometric features. A reader who is not quite familiar with hyperbolic modeling or embedding may find this terminology confusing. Thus, I suggest explaining this terminology the first time it is used in the text, at the end of Section IA.

Reply and action taken: We have rewritten Section I/A to separate the explanations of the bipartite-$\mathbb{S}^D/\mathbb{H}^{D+1}$ model and B-Mercator, and to clarify the use of the term network topology.

Comment 2.7.: In Section IIIBa, why is the parameter β_b initially guessed in the range (D, 2D)? In

the model assumptions, the inverse temperature is assumed to be larger than D, not necessarily in the interval (D, 2D).

Reply: Yes, the β_b should be larger than D to be in the geometric region. B-Mercator estimates β_b, and we think that using an initial guess in the interval $(D, 2D)$ is reasonable. The exact choice of this initialization interval is not very relevant, since β_b is subsequently updated and inferred during optimization.