Knowledge-Based Systems 246 (2022) 108699

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Community preserving mapping for network hyperbolic embedding

Dongsheng Ye [a], Hao Jiang [a,*], Ying Jiang [b], Qiang Wang [a], Yulin Hu [a]

[a] Electronic Information School, Wuhan University, Wuhan, 430072, China
[b] School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, 510000, China

ARTICLE INFO

Article history:
Received 24 September 2021
Received in revised form 26 March 2022
Accepted 27 March 2022
Available online 1 April 2022

Keywords:
Hyperbolic geometry
Network embedding
Community embedding

ABSTRACT

Hyperbolic embedding aims to reveal the hidden space by capturing most of the properties observed in real networks. Most existing hyperbolic embedding models map to learn the representation vectors while preserving the microscopic adjacency, which cannot accurately represent the mesoscopic community structures. To this end, this study proposes a Community Preserving Hyperbolic Embedding model (CPHE). Specifically, we regularize the likelihood function of hyperbolic embedding by adding to the community co-occurrence relation (CCR). We then construct a closed loop for node embedding and community detection. Thus, the representation vectors and CCR alternately update. Finally, to avoid the distortion of the representing community, an equivalent majorization based on the sum of linear ratios programming is achieved for the numerical solution. Application experiments are implemented on both synthetic and real-world networks to evaluate the embedding performance. The accuracy and normalized mutual information (NMI) of community detection improved by approximately 3% and 2.4%, respectively, and the area under the receiver operating characteristic curve (AUROC) of link prediction improved by approximately 1.3%, demonstrating the advantages of the proposed model.

1. Introduction

Networks, as universal representations of complex relationships, have been widely used to depict biological, social, and information systems [1]. Recently, the analysis of networks in a hidden space has attracted increasing attention recently owing to its intuitive and interpretable geometric construction [2–4]. Following these lines of research, networks generated using random hyperbolic graph (RHG) theory have reproduced sparsity, self-similarity, scale-free degree distribution, strong clustering, and small-world properties, which are common to most complex networks [5–7]. Therefore, the inverse process of embedding networks based on RHG, called hyperbolic embedding, has been widely applied in the fields of community detection [8–10], navigability [11–13], and network renormalization [2]. Compared with the embeddings in Euclidean space, studies demonstrate that hyperbolic embedding leads to lower distortions and complexities in downstream tasks [14].

Among these hyperbolic embedding methods, a common assumption is that the geodesic distances between the spatial vectors reflect the affinities between the corresponding nodes [15]. In this direction, most hyperbolic embedding methods, such as HyperMap [16,17], Laplacian-based Network Embedding (LaBNE) [18], and Coalescent [19], encode the adjacencies. On this basis, the Efficient Embedding (EE) [20] and Mercator [21] methods were extended to capture additional information from common neighbors for a more precise vector representation. However, all these methods primarily focus on the microscopic structure of the network and ignore the mesoscopic description. The community, intuitively defined as a group of densely connected nodes, is a prominent mesoscopic feature that reveals the organizational structures and functional components of a network [22]. For instance, the characteristics of social network communities, including the size of the largest component, size of the reachable cluster, and average clustering coefficient, have a significant influence on the relative success of infecting and avoiding diseases such as COVID-19 [23]. By the homophily principle [24], nodes within a community are more affinitive to each other than are those belonging to different communities. Therefore, even if two nodes from a community have a weak connection due to network sparsity, their latent affinities will be strong. Because downstream applications, such as community detection, similarity search, and node classification, strongly rely on these affinities. Embedding without preserving the community leads to incomplete geometric representations and incurs indeterminate errors [24]. Consequently, community preservation is critical for hyperbolic embedding.

To preserve the community during hyperbolic embedding, incorporating latent affinities into the adjacencies to guide the encoding of link likelihoods seems to be a promising solution. Following this, state-of-the-art hyperbolic embeddings, i.e., Community and Hyperbolic Mapping (CHM) [25] and Link Prediction

* Corresponding author.
E-mail addresses: yesdong@whu.edu.cn (D. Ye), jh@whu.edu.cn (H. Jiang).

https://doi.org/10.1016/j.knosys.2022.108699
0950-7051/

with Community Structure (LPCS) [26], first determine the range of a sector of each community in the hyperbolic plane. Then, they insert nodes into the sectors by maximizing local likelihoods. Such pipeline approaches depict communities by assigning regions in the embedding space; however, their rules for integrating communities are largely influenced by the community detection results of upstream algorithms. Moreover, the solutions to their models are suboptimal and unexplained, as there is no unified objective function to formulate both the community and adjacency. Then, models, such as [27,28], directly couple the likelihoods of community detection with the likelihoods of first- and second-order proximities. Consequently, latent affinities and adjacencies can be unified to encode the representation vectors using a Riemannian stochastic gradient descent [29]. However, the representation vectors obtained from these approaches are limited in describing the local features of the networks. The lack of encoding global features, especially the degree distribution, complicates their application in greedy routing [30] and network renormalization [2].

Thus, we aim to construct an RGH-based mapping regularized by the community, in which both the latent affinities and adjacencies are preserved under a global framework. Three main issues require attention for this mapping: community regularization, unsupervised embedding, and optimization. First, the relations depicted by the community are mesoscopic and multivariate, which are quite different from the microscopic and bivariate adjacency [31]. Thus, to construct the regularization, transforming the multivariate relations into bivariate relations is necessary. Second, because the priori community structure, such as the membership matrix and community label, cannot be directly given for most networks, coupling community detection and node embedding is crucial. Finally, the optimization for the CPHE can be performed analytically with respect to the radius, but not with respect to the angles [16]. Many approximate methods have been presented to implement a numerical solution to non-concave angular coordinate majorization with low complexity [17,20,32,33]. However, their strategies, essentially uniform sampling, would homogenize the cluster structure of the representation vectors.

To overcome these issues, this paper proposes a community-preserving hyperbolic embedding model called CPHE. First, analogized with the co-occurrence in computational linguistics [34], computer vision [35], and community ecology [36,37], the multivariate community relations were decomposed into bivariate co-occurrence relations, and the latent affinities were quantified using the community co-occurrence of node pairs. Subsequently, considering that the community shows a significant overlap with the spatial cluster of representation vectors [32,38,39], CPHE fits the mixture distribution of angular coordinates to detect the community and combines community-preserving embedding with community detection for an aligned representation. Finally, we propose a novel majorization equivalent to the objective function of CPHE, whose numerical solution is implemented based on the sum of linear ratio programming (SLR) [40]. Our contributions can be summarized as follows:

- We transform the community structure into CCRs to regularize the hyperbolic distance. To the best of our knowledge, we are the first to analytically model the latent affinities hidden in the community for hyperbolic embedding.
- We propose CPHE, which extends hyperbolic embedding with a CCR regularization and achieves community preserving while embedding.
- We use the sum of linear ratio programming to solve the likelihood maximization in CPHE, which effectively avoids the distortion of representing community caused by the uniform sampling.
- We evaluate the embeddings using three types of downstream applications on six synthetic datasets and twelve real-world networks. The proposed CPHE improved approximately 3% in accuracy, 2.4% in NMI for community detection, 0.7% accuracy in node classification, and 1.3% AUROC in link prediction, on average. On this basis, we analyze the effects of each of the three model parameters on the community preservation performance.

The remainder of this paper is organized as follows. Section 2 reviews related studies on hyperbolic network embeddings. Section 3 introduces the fundamentals of typical hyperbolic generation and embedding models. Then, Section 4 presents a novel community-preserving embedding model, deferred to as random hyperbolic graph theory. The corresponding algorithm using the sum of the linear ratio problem is presented. A series of comparative experiments are performed, and the parameters of the models are analyzed in Section 5. Section 6 presents the conclusions.

2. Related works

As mentioned previously, hyperbolic embedding aims to represent the nodes of a network into a hyperbolic space while retaining the network structure. In this section, we review related hyperbolic embedding methods and summarize their characteristics. Subsequently, we introduce existing approaches to preserve the community structure during embedding, including both Euclidean space-based and hyperbolic embedding. Table 1 illustrates these related studies in terms of category, reference, embedding space, preserved structure, fundamental theory, and application.

2.1. Hyperbolic embedding

Hyperbolic geometry emerges from relaxing Euclid's parallel postulate of geometry, and the space with a uniform negative curvature is called hyperbolic. According to Kleinberg's well-known result in [47], every connected finite graph has a greedy embedding in the hyperbolic plane (2D hyperbolic space), and many sophisticated real networks, such as the Internet, communication networks, and social networks, do not admit such an embedding in the Euclidean plane.

To represent networks with spatial vectors in a hyperbolic plane, embedding approaches are broadly divided into two groups. First, literature like [28,42,48] design the embedding loss using the implicit function between the proximity relation and hyperbolic distance. Thus, the optimization can be naturally solved using Riemannian stochastic gradient descent (R-SGD) [29] or Riemannian multidimensional scaling (R-MDS) [49]. However, the embedding vectors obtained by these methods are limited in interpretability. Second, based on various generation models called random hyperbolic graphs (RHG) [16,32,50,51], many hyperbolic embeddings inference representation vectors by maximizing the explicit likelihood of the connections [18,20,52]. As the obtained representation vectors are sufficient for reconstructing the diameter [53], component structure [54], clique size [55], and separation properties [56], these embeddings are able to mathematically analyze physical properties of the original networks in the hyperbolic plane. Correspondingly, the explicit formulations of RHG also result in a nonconvex embedding objective function, which complicates the optimization of the angular coordinates.

Manifold learning and candidate sampling solve the RHG-based embedding optimization from different viewpoints. The former one, inspired by the conformal property [52], estimates the angular coordinates by embedding a network into a general manifold rather than a hyperbolic space. For instance, the

Table 1
Compare embeddings in related works.

Category	Reference	Embedding space	Preserved structure	Fundamental theory	Applications
Hyperbolic embedding without generation model	[41]	Minkowski space	Adjacencies	R-MDS	Link prediction
	[42]	Pseudo-Riemannian manifold	Adjacencies	R-SGD	Hierarchy extraction
PSO model based embedding	[18]	Poincaré plate	Adjacencies, Ddegree distribution	Laplacian eigenmaps	Link prediction
	[19]	Poincaré disc, 3D hyperbolic space	Adjacencies, Ddegree distribution, Edge betweenness centrality	Isomap, Minimum curvilinearity, Laplacian eigenmaps	Link prediction, Greedy routing, Community detection
	[16]	Poincaré disc	Adjacencies, Ddegree distribution	Maximum likelihood	Network navigation, Link prediction
	[20]	Poincaré disc	Adjacencies, Ddegree distribution, Common neighbors	Maximum likelihood	Link prediction
	[33]	Poincaré disc	Adjacencies, Ddegree distribution	Laplacian eigenmaps, Maximum likelihood	Community detection, Link prediction
	[25]	Poincaré disc	Adjacencies, Ddegree distribution, Community	Maximum likelihood	Community detection, Link prediction
$\mathbb{S}^1$ model based embedding	[21]	$\mathbb{S}^1$ space	Adjacencies, Ddegree distribution	Laplacian eigenmaps	Link prediction, Community detection
	[43]	$\mathbb{S}^1$ space	Adjacencies, Ddegree distribution, Edge weights	Maximum likelihood	Network Navigation, Community detection, Node Classification
European space based embedding	[31]	European space	Adjacencies, Community	Matrix factorization	Community detection
	[44]	European space	Adjacencies, Community	Random walk	Link prediction, Community detection
	[45]	European space	Adjacencies, Community, Node attributes	Random walk	Link prediction, Node Classification
	[24]	European space	Adjacencies, Community,	Gradient descent	Community detection, Node Classification
	[46]	European space	Adjacencies, Node attributes	Neural network	Community detection, Node Classification

Laplacian operator constructs a 2D manifold representation of networks in LaBNE [18], and the tangent of the phase is calculated using the ratio of the first two nonzero eigenvectors. However, approximating distances on an unweighted adjacency matrix leads to a limited representational capacity in this embedding. Thus, there was a clear margin to improve the embedding performance by pre-weighting the network links using a convenient strategy. Coalescent Embedding [19] incorporates the degree of information and assigns weights to edges using structure rules, such as common neighbors and shortest path. In order to embed the aforementioned weighted graphs, they provides the interfaces for diverse manifold learning methods. Analogously, Mercator [21] infers the angular coordinates using Laplace eigenmaps in $\mathbb{S}^1$. In this model, weights deduced from the expectation of angular separation quantify the relations between the hidden degree and embedding distance and make the nodes with a large degree approach each other. The manifold learning technique avoids nonconvex optimization in RHG models but lacks the explicit quantization of the hidden space. Embedding networks into a general manifold produces a representation error if the manifold has obvious architectural differences from the hyperbolic space. Naturally, the complexity of this kind of approaches is susceptible to Eigendecomposition algorithm.

Another approaches sample candidates from the feasible region and search for approximate solutions among these candidates. HyperMap [16,17] infers the angular positions corresponding to the maximum likelihood within the candidates uniformly sampled from the continuous interval. The embedding is suitable for the World Trade Atlas 1870–2013 geometry analysis [32], which mapped the international trade system into a Poincaré disc and interpreted the trade connection as the geodesic distance. Efficient Embedding (EE) [20] generates the drawing of core nodes with a spring embedder and arranges the other nodes ordered by common neighbors. To reproduce the edge weights of networks, [43] embedded a network by incorporating the node strength and adjacency. The representation vectors were obtained by maximizing the likelihoods of the links and weights. Furthermore, the hybrid approach, such as LaBNE+HM [33], intends to aggregate the advantages of LaBNE and HyperMap for an efficient and accurate network embedding. They first use LaBNE to draft a geometric configuration of a given network, the draft is then passed on to HyperMap for refining the final mapping to hyperbolic space. Because these embeddings are constructed based on the RHG theory, their representation vectors can be analytically obtained and used to express the formation mechanism of edges.

2.2. Community embedding

Nevertheless, all approaches mentioned above examine the embedding distance only with microscopic affinity, such as adjacency and common neighbor. This leads to a weak quantification of the mesoscopic community and incurs non-negligible biases for network analysis. Hyperbolic embeddings such as in [25,26] preserve the community structure using a two-step approach: community detection and network embedding. However, these two steps are implemented independently and are based on different fundamental theories, which makes the representation vectors suboptimal and unexplained. Among the RHG-based embedding models, mapping, analytically derived under the maximum entropy principle, is nonlinear, and the complex coupling between representation vectors complicate directly incorporating community regularization.

Evidently, most of the existing community-preserving embedding models lay in a Euclidean space, and the mappings are constructed without the need to satisfy a network generation model. Modularized nonnegative matrix factorization [31] adopts the modularity matrix to model the community membership indicator. The representation vectors are obtained with the linear embedding and modularity approximation. In [44], the overlapping community structure was modeled using conditional probability during a random walk. Gao et al. [45] introduced topic models to vertex sequences to capture the community information of the network, and they designed a community-oriented random walk strategy to adaptively control the scope of random walks. In addition, Sandro et al. [24] regards each community as a multivariate Gaussian distribution and closed a loop to learn the representation vectors. During the learning, community and node embeddings were simultaneously implemented in Euclidean space. Others, such as [46], rely on graph convolutional network architectures to maximize the mutual information between patch representations and corresponding high-level summaries of graphs to retain the network information as much as possible. However, without meeting a network generation model, the embedding space of these models lacks physical significance, and the spatial structure of representation vectors cannot be interpreted with the natural features of networks, such as degree distribution and small world. This complicates choosing an embedding space with proper geometrical characteristics, especially dimensionality.

Unlike the above studies, we propose an extended PSO model-based embedding to map networks while preserving the community. With the proposed model, both the latent affinities hidden in the community and adjacencies can be maintained to obtain a less distorted representation. More importantly, because embedding is proposed in terms of a geometric generative model, the representation vectors are interpretable and can guide the construction of downstream network analysis models.

3. Definitions and preliminaries

This section first provides several fundamental definitions that were defined similarly to Palash et al. [57]. Then, the basic hyperbolic dynamic generation model [58] and corresponding hyperbolic growth embedding model [16] are briefly introduced.

3.1. Fundamental definitions

Definition 3.1 (*Network [59]*)**.** A network $G(V, E)$ is a collection of a vertex set $V = \{V_1, V_2, \ldots, V_n\}$ and an edge set $E \subseteq V \times V$.

The edges can be indicated as the adjacency matrix $\{a_{ij}\}_{n\times n}$, where $a_{ij} = 1$ if an edge exists between V_i and V_j, and $a_{ij} = 0$ otherwise. Evidently, $\{a_{ij}\}_{n\times n}$ is symmetric in an undirected network.

Definition 3.2 (*Hyperbolic Embedding*)**.** Given a network $G(V, E)$, the hyperbolic embedding of the network is a mapping from the vertex set to the hyperbolic spatial vectors, and the mapping preserves the proximity measure defined in network G.

Similar to the graph embedding in [57], hyperbolic embedding converts the network into low-dimensional vectors and quantifies the adjacency using the geodesic distance of the corresponding points. Nodes are mapped near for adjacent pairs and far for nonadjacent pairs in a hyperbolic space.

Definition 3.3 (*Community [22]*)**.** A community is a subset of the vertex set satisfying connectedness. The vertices in the community must have more edges than those linking the vertices of the community with the rest of the network.

The communities $C = \{C_1, C_2, \ldots, C_L\}$ of a network can be quantized by the membership matrix $\{w_{il}\}_{n\times L}$, where $w_{il} \in [0, 1]$ measures the degree of membership between node V_i and community C_l. The large value of w_{il} corresponds to a strong membership, and the constraint shown in (1) is defined for normalization.

$$\sum_{l=1}^{L} w_{il} = 1. \tag{1}$$

From this community definition, each row of the membership matrix $\{w_{il}\}_{n\times L}$ encodes the memberships between a member and all communities. Typically, the community label corresponds to the column label with the maximum membership.

3.2. Hyperbolic dynamic generation model

Network generation aims to reproduce network properties using geometrical measures. Popularity-similarity-optimization (PSO) [58] is a basic hyperbolic dynamic generation model that depicts the two dimensions of the embedding vectors, as popularity and similarity, under the RHG framework. In this model, each time a new node appears on the Poincaré disc, the edges are randomly created using the Boltzmann distribution [60] based on the Poincaré metric [61,62].

Let $\mathscr{G}\{G; P(G)\}$ be a canonical ensemble [63] of networks with a probability measure $P(G)$ for any graph $G \in \mathscr{G}$. $\{J_\xi\}_{\xi\in\Xi}$ is a set of observation functions with expectation values $\langle J_\xi\rangle$ for the ensemble. Thus, the probability distribution of $P(G)$ can be derived by maximizing the ensemble Gibbs entropy under the observation constraints as optimization (2).

$$\begin{aligned} &\max_{P(G)} \sum_{G\in\mathscr{G}} -P(G)\ln P(G) \\ &s.t. \sum_{G\in\mathscr{G}} P(G)J_\xi(G) = \langle J_\xi\rangle, \ where\ \xi \in \Xi \\ &\quad \sum_{G\in\mathscr{G}} P(G) = 1. \end{aligned} \tag{2}$$

Using the Lagrange multiplier method, $P(G)$ can be expressed as the following Boltzmann distribution:

$$P(G) = \frac{e^{-H(G)}}{Z}, \tag{3}$$

where $H(G) = \sum_{\xi\in\Xi} \alpha_\xi J_\xi(G)$, which is formed by coupling the Lagrange multipliers $\{\alpha_\xi\}$ to the observation constraints, is the graph Hamiltonian. In addition, $Z = \sum_{G\in\mathscr{G}} e^{-H(G)}$ is the partition function that maintains $\sum_{G\in\mathscr{G}} P(G) = 1$.

For the canonical ensemble of the geometric random graph above, two basic observation constraints represent the number of fermions and energy of the system, yielding [7]

$$\begin{cases} J_1(G) = \sum_{i<j} a_{ij} = \langle J_1\rangle \\ J_2(G) = \sum_{i<j} x_{ij}a_{ij} = \langle J_2\rangle \end{cases} \tag{4}$$

where a_{ij} is the element of the network adjacency matrix in the ith row and jth column, and x_{ij} is the geodesic distance in the Poincar'e plate disc, defined as (5)

$$x_{ij} = \frac{1}{\zeta}\operatorname{arccosh}(\cosh\zeta r_i \cosh\zeta r_j - \sinh\zeta r_i \sinh\zeta r_j \cos\theta_{ij}), \tag{5}$$

where $\zeta = \sqrt{-K}$ is a space parameter constructed using the curvature K of the hyperbolic space, and $r_j = \frac{2}{\zeta}\ln j$ and $\theta_{ij} = \pi - |\pi - |\theta_i - \theta_j||$ are the radial coordinates and phase difference at time j, respectively. Furthermore, x_{ij} can be approximated as (6) under the condition $\theta_{ij} > 2\sqrt{e^{-2\zeta r_i} + e^{-2\zeta r_j}}$,

$$x_{ij} \approx r_i + r_j + \frac{2}{\zeta}\ln\theta_{ij}/2. \tag{6}$$

Substituting Eqs. (4) into the observation constraint (3), we can denote the Lagrange multipliers as $\alpha_1 = -\frac{\zeta R_j}{2T}$ and $\alpha_2 = \frac{\zeta}{2T}$. Then,

$$P(G) = \frac{\prod_{i<j} e^{(R_j - x_{ij})a_{ij}\zeta/2T}}{Z} = \prod_{i<j} P(a_{ij}) \tag{7}$$

Thus,

$$P(a_{ij}) = \frac{e^{(R_j - x_{ij})a_{ij}\zeta/2T}}{e^{(R_j - x_{ij})a_{ij}\zeta/2T} + 1}. \tag{8}$$

The form of the connection probability can be simplified to formula (9) as $p_{ij} = P(a_{ij} = 1)$.

$$p_{ij} = \frac{1}{1 + e^{\frac{\zeta}{2T}(x_{ij} - R_j)}}. \tag{9}$$

Once in this form, the parameter T is interpreted as the temperature that controls the network clustering, which decreases almost linearly with $T \in [0, 1)$ and becomes asymptotically zero when $T > 1$. In addition, the threshold R_j can be derived from the model assumptions of the average node degree $2m$ and power-law degree distribution $P(k) \propto k^{-(1+1/\beta)}$, where $\beta \in (0, 1]$.

$$R_j = r_j - \frac{2}{\zeta} \ln \left[\frac{2T}{\sin(T\pi)} \frac{1 - e^{-\frac{\zeta}{2}(1-\beta)r_j}}{m(1-\beta)} \right]. \tag{10}$$

3.3. Hyperbolic growth embedding model

Under the PSO model framework, HyperMap [16] replays the maximum likelihood growth of the edges to embed the given network into the hyperbolic plane. Using Bayes' rule, the probability of the coordinates $\{r_j, \theta_j\}_{j\times 2}$ taken from the model parameter is formulated using the conditional probability as follows:

$$P\left(\{r_i(j), \theta_j\} \middle| \{a_{ij}\}, \zeta, \beta, m, T\right) = \frac{P\left(\{r_i(j), \theta_j\} \middle| \zeta, \beta, m, T\right) P\left(\{a_{ij}\} \middle| \{r_i(j), \theta_j\}, \zeta, \beta, m, T\right)}{P(\{a_{ij}\} | \zeta, \beta, m, T)}. \tag{11}$$

As the density function of $r_i(j)$ can be expressed as $P(r_i(j)) = \frac{\zeta}{2\beta} e^{\frac{\zeta}{2\beta}(r_i(j) - r_j)}$ based on the model assumption, the first term of the numerator in (11) is derived using the independence principle as follows:

$$P\left(\{r_i(j), \theta_j\} \middle| \zeta, \beta, m, T\right) = \left(\frac{\zeta}{4\pi\beta}\right)^j \prod_{i=1}^{j} e^{\frac{\zeta}{2\beta}(r_i(j) - r_j)}. \tag{12}$$

The concrete form of the second fraction is constructed with the probability $P(G)$ in formula (7), which quantifies the correlation between the adjacent matrix and the graph distribution of the ensemble.

$$P\left(\{a_{ij}\} \middle| \{r_i(j), \theta_j\}, \zeta, \beta, m, T\right) = \sum_{i<j} a_{ij} \ln p_{ij} + \sum_{i<j} (1 - a_{ij}) \ln(1 - p_{ij}). \tag{13}$$

Substituting both (12) and (13) into the original conditional probability (11), the likelihood function can be simplified as:

$$\mathcal{L} = C + \frac{\zeta}{2\beta} \sum_{i=1}^{j} r_i(j) + \sum_{i<j} a_{ij} \ln p_{ij} + \sum_{i<j} (1 - a_{ij}) \ln(1 - p_{ij}), \tag{14}$$

where $C = j\left(\ln \frac{\zeta}{4\pi\beta} - \frac{\zeta}{2\beta}\right) - \ln P\left(\{a_{ij}\} \middle| \zeta, \beta, m, T\right)$ is a constant that is independent from $\{r_j, \theta_j\}_{j\times 2}$.

Thus, the coordinates are inferable using the maximization of (14). The radial coordinates are calculated analytically according to the stationary point of the likelihood function as follows:

$$\frac{\partial \mathcal{L}}{\partial r_i(j)} = \frac{\zeta}{2\beta} - \frac{\zeta}{2T} \left(\sum_{t=1, t\neq i}^{j} a_{it} - \sum_{t=1, t\neq i}^{j} p_{it} \right). \tag{15}$$

Setting (15) equal to zero, we have $r_j = \frac{2}{\zeta} \ln j$. With respect to the angular coordinate, the inference is numerically obtained by sampling the likelihood $\mathcal{L}$ at different values of θ_j in $[0, 2\pi]$ separated by the interval $\frac{1}{j}$, owing to the indifferentiable and non-convex properties.

4. Methodology

This section presents the details of the proposed CPHE. First, the latent affinities hidden in the community are modeled to regularize the objective function of the hyperbolic embedding using the co-occurrence relation. Next, an unsupervised embedding framework with community fitting is proposed to simultaneously detect the community and embed nodes. Finally, a majorization equivalent to CPHE is proposed, and the corresponding solution is implemented to search the embedding coordinates.

4.1. Regularization based on community co-occurrence relations

As mentioned previously, the community expresses the latent affinities between multiple objects, whereas the hyperbolic metric used to construct the embedding objective function is bivariate. To regularize hyperbolic embedding using the latent affinities, we decomposed the community relations into bivariate community co-occurrence relations (CCRs) and quantified the latent affinity using the CCRs.

Regarding membership grade w_{il} as the class-specific distribution, a CCR can be generated using the Separable Mixture Model [64] as in (16)

$$\epsilon_{ij} = \chi_\upsilon \left(-\sum_{l=1}^{L} w_{il} w_{jl} \right), \tag{16}$$

where $\epsilon_{ij} \in \{0, 1\}$, $\epsilon_{ij} = 1$ denotes a CCR between node V_i and V_j, $\epsilon_{ij} = 0$ for no CCR. In addition, $\chi_\upsilon(x)$ is the indicator function controlled by the threshold υ, $\chi_\upsilon(x) = 1$ for $x < \upsilon$, and $\chi_\upsilon(x) = 0$ for $x \geq \upsilon$. Evidently, the CCR matrix $\{\epsilon_{ij}\}_{n\times n}$ analytically encodes the latent affinity hidden in the community.

Our goal is to closely map the node pairs with the CCR in a hyperbolic space. Correspondingly, the connection probability of node pairs with CCR should not be less than that of the same pairs without the CCR. As shown in (9), the distance x_{ij} is inversely proportional to the connection probability p_{ij}; thus, we extend $\tilde{x}_{ij}$ with the CCR regularization as in (17)

$$\tilde{x}_{ij} = \epsilon_{ij} x_{ij}^{IN} + (1 - \epsilon_{ij}) x_{ij}^{OUT} \quad s.t. \inf_{(i,j)\in\{(i,j)|\epsilon_{ij}=1\}} \left(x_{ij}^{IN} - x_{ij}^{OUT}\right) \geq 0. \tag{17}$$

where x_{ij}^{IN} denotes the hyperbolic distance for the pair (V_i, V_j) with the CCR, and x_{ij}^{OUT} for the pair without CCR. Substituting the approximation of the hyperbolic distance formula in (6) into the extended metric $\tilde{x}_{ij}$, we obtain

$$\tilde{x}_{ij} = r_i + r_j + \frac{2}{\zeta} \ln \frac{\tilde{\theta}_{ij}}{2} \quad s.t. \inf_{(i,j)\in\{(i,j)|\epsilon_{ij}=1\}} \left(\theta_{ij}^{IN} - \theta_{ij}^{OUT}\right) \geq 0 \tag{18}$$

where $\tilde{\theta}_{ij} = (\theta_{ij}^{IN})^{\epsilon_{ij}} (\theta_{ij}^{OUT})^{(1-\epsilon_{ij})}$ is the CCR regularized angle metric.

With respect to the angle metric, the embedding experiments in [25,32,38,39] showed that nodes belonging to a common community usually gather within a certain range of angles in a hyperbolic space. Thus, the CCR can also be expressed with the phase interval of the representation vectors using the following piecewise function (19):

$$\tilde{\theta}_{ij} = \begin{cases} \theta_{ij}^{IN}, & \text{if } \theta_{ij} < \tau \\ \theta_{ij}^{OUT}, & \text{if } \theta_{ij} \geq \tau \end{cases} \quad (19)$$
$$s.t. \inf_{(i,j)\in\{(i,j)|\epsilon_{ij}=1\}} \left(\theta_{ij}^{IN} - \theta_{ij}^{OUT}\right) \geq 0.$$

Similarly, $\theta_{ij} = \pi - |\pi - |\theta_i - \theta_j||$, where $\theta_i \in [0, 2\pi]$. And $\tau \in (0, \pi)$ is the geometric thresholds.

Before constructing $\tilde{\theta}_{ij}$, we limit θ_i to $(0, \pi)$, based on the maximum variance unfolding for network embedding [20]. As this limitation does not change the approximate uniform distribution on $[0, \pi]$ of θ_{ij} in the PSO model [58], the threshold R_j obtained by integrating θ_{ij} can still be derived as (10).

The Huber penalty function [65] is a popular loss function in deep learning because it is smooth and less sensitive to outliers than the L_2 loss function [66]. Based on this approximation method, we constructed the CCR-regularized angle metric $\tilde{\theta}_{ij}$ as in (20). The significant properties of $\tilde{\theta}_{ij}$ are introduced in Theorems 4.1–4.2, and their details are shown in Appendix A.

$$\tilde{\theta}_{ij} = \begin{cases} \frac{1}{2\tau}(\theta_i - \theta_j)^2 + \frac{\tau}{2}, & \text{if } |\theta_i - \theta_j| < \tau \\ |\theta_i - \theta_j|, & \text{if } |\theta_i - \theta_j| \geq \tau \end{cases} \quad (20)$$

Theorem 4.1. *The piecewise function $\tilde{\theta}_{ij}$ of θ_i and θ_j, defined as (20), is differentiable.*

Theorem 4.2. *Let $\theta_{ij}^{IN} = \frac{1}{2\tau}(\theta_i - \theta_j)^2 + \frac{\tau}{2}$ and $\theta_{ij}^{OUT} = |\theta_i - \theta_j|$. Then, $\theta_{ij}^{IN} \geq \theta_{ij}^{OUT}$ is established for $|\theta_i - \theta_j| < \tau$.*

Fig. 1(a) exhibits the original angle metric θ_{ij} and proposed angle metric $\tilde{\theta}_{ij}$ between θ_i and θ_j. Evidently, the proposed $\tilde{\theta}_{ij}$ is not less than θ_{ij}, when the phase separation is less than the geometric threshold τ. Further, $\tilde{\theta}_{ij}$ would degrade into θ_{ij} for $\tau = 0$, which corresponds to the inactive community regularization. Fixing the hyperbolic distance between node V_i and V_j, the phase separation $|\theta_j - \theta_i|$ obtained by the CCR-regularized metric would be less than that calculated using θ_{ij}. Fig. 1(b) illustrates the difference of phase separation $\Delta|\theta_j - \theta_i|$ obtained by these two metrics subject to $\tilde{x}_{ij} = x_{ij}$. Obviously, the difference decreases with the increase of phase separation and equals zero with respect to $|\theta_j - \theta_i| > \tau$. That is, the effect of the community regularization gradually wanes as the CCR becomes weaker during the representing using the given $\tilde{x}_{ij}$.

Moreover, the topological threshold υ acts on the community information encoding, whereas the geometrical threshold τ determines the sphere of community influence. Both control the community regularization of embedding in this model. A smaller υ and larger τ would produce a stronger community impact on the network representations.

4.2. Unsupervised embedding with community fitting

A priori community knowledge, such as community labels and membership matrix, is necessary to construct the CCR-regularized angle metric; however, it is not directly available in most real-world network analyses. Embedding after community detection is suboptimal owing to the independence of these two processes [67]. As the community shows a significant overlap with the spatial cluster of representation vectors [32,38,39], we fit the distribution of the angle coordinates to detect the community. Then, the hyperbolic distance regularized by the obtained community is used to embed the network to construct a closed loop. Community fitting and node embedding are alternately implemented to align the community with the angle cluster.

The likelihood of node embedding with the CCR-regularized angle metric is similar to the standard embedding likelihood in (14) and can be formulated as (21):

$$\tilde{\mathcal{L}} = C + \frac{\zeta}{2\beta}\sum_{i=1}^{j} r_i(j) + \sum_{i<j} a_{ij} \ln \tilde{p}_{ij} + \sum_{i<j}(1 - a_{ij}) \ln(1 - \tilde{p}_{ij}) \quad (21)$$

where $\tilde{p}_{ij} = \frac{1}{1+e^{\frac{\zeta}{2T}(\tilde{x}_{ij} - R_j)}}$. We set $\zeta = 1$ for the native representation of hyperbolic space because the curvature has minimal effect on the topological properties of networks [58].

The Gaussian mixture model (GMM) is a classical method used to represent the community structure. Node membership values can be calculated using the probability of Gaussian distribution. Therefore, we fit the angle coordinates using the family of Gaussian distributions, and the fitting of the likelihood is expressed as follows:

$$\mathcal{L}_{comm} = \sum_{i=1}^{n} \ln \sum_{l=1}^{L} W_l \mathcal{N}\left(\theta_i | \mu_l, \sigma_l\right). \quad (22)$$

In (22), the phase of node θ_i is defined as a random variable obeying the weighted superposition of the Gaussian distributions $\mathcal{N}(\theta_i|\mu_l, \sigma_l)$. μ_l and σ_l denote the expectation and standard deviation of the lth Gaussian distribution. W_l is the global membership parameter that measures the probability that cluster C_l contains an arbitrary node in graph G.

Finally, we incorporate community fitting into node embedding and add the likelihood of GMM to the likelihood of hyperbolic connection. The ultimate objective function is displayed as (23):

$$\mathcal{L}^* = \tilde{\mathcal{L}} + \mathcal{L}_{comm}. \quad (23)$$

And the representation vectors with community structure can be obtained by maximizing $\mathcal{L}^*$.

4.3. Solution to hyperbolic community embedding

The solution for maximizing $\mathcal{L}^*$ is implemented by alternately fitting the community and embedding nodes. Specifically, we maximize the community-fitting likelihood $\mathcal{L}_{comm}$ to achieve community detection. Based on the obtained community structure, we then maximize node embedding likelihood $\tilde{\mathcal{L}}$. Finally, we re-use the obtained embedding results to update the community divisions by fitting the distribution of representation vectors.

To maximize $\mathcal{L}_{comm}$, the Expectation Maximization algorithm is implemented to optimize the community fitting $\{w_{il}\}_{n\times L}$ according to Jensen's inequality [68]. The iteration formulations are as follows:

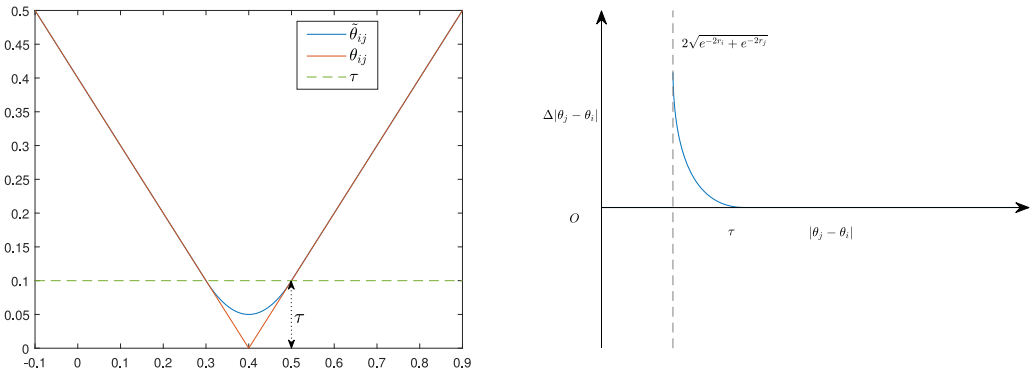
$$\mu_l = \frac{\sum_{i=1}^{n} w_{il}\theta_j}{\sum_{i=1}^{N} w_{il}}, \quad (24)$$

$$\sigma_l^2 = \frac{\sum_{i=1}^{n} w_{il}(\theta_i - \mu_l)^2}{\sum_{j=i}^{N} w_{il}}, \quad (25)$$

$$W_l = \frac{\sum_{i=1}^{n} w_{il}}{n}, \quad (26)$$

where

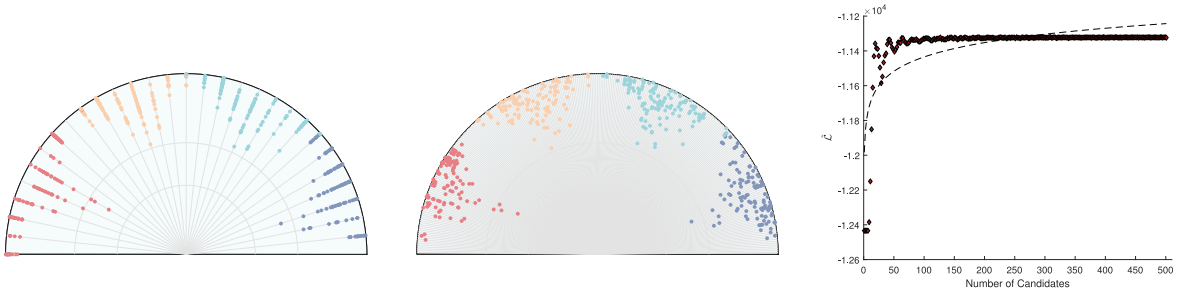
$$w_{il} = \frac{W_l \mathcal{N}\left(\theta_i|\mu_l, \sigma_l\right)}{\sum_{l=1}^{L} W_l \mathcal{N}\left(\theta_i|\mu_l, \sigma_l\right)}.$$



(a) The relations between θ_{ij} and $\tilde{\theta}_{ij}$

(b) The effect of $\tilde{\theta}_{ij}$ on $|\theta_j - \theta_i|$

Fig. 1. The features of the proposed CCR regularized angle metric $\tilde{\theta}_{ij}$.



(a) Embedding results when 30 candidates are sampled

(b) Embedding results when 300 candidates are sampled

(c) Relations between likelihoods $\tilde{\mathcal{L}}$ and number of candidates in the sampling

Fig. 2. Uniformly sampling in $[0, \pi]$ to solve the maximization of $\tilde{\mathcal{L}}$.

To maximize $\tilde{\mathcal{L}}$, the optimization can be similarly implemented using formula (15) for the radial coordinate, as CCR only regularizes the similarity corresponding to the angular coordinate. For the angular coordinate, the classical processing [16,17,20,32] sample the candidates on $[0, \pi]$ at equal angle intervals and chose solutions from these candidates according to the principle of maximizing likelihoods. However, as shown in Fig. 2(c), the maximum likelihood score of these solutions decreases with decreasing candidates numbers. The primary reason is that uniform sampling filters the given community structure and leads to an indistinguishable representation, as shown in Figs. 2(a) and 2(b). Consequently, sampling in this numerical method narrows the feasible region of the embedding optimization and causes a suboptimal inference of the community-preserving coordinates.

Hence, we transport the original majorization into the sum of the linear ratios (SLR) programming [40] to search for the angular coordinates in a continuous interval without uniform sampling. The iteration formulations (24)–(26) of community fitting are combined for the numerical solutions.

Based on the differentiability of the novel metric $\tilde{\theta}_{ij}$ demonstrated in Theorem 4.1, the partial derivative of $\tilde{\mathcal{L}}$ with respect to θ_j can be derived as (27)

$$\frac{\partial \tilde{\mathcal{L}}}{\partial \theta_j} = \frac{1}{T}\sum_{i=1}^{j}(\tilde{p}_{ij} - a_{ij})\left[\frac{\epsilon_{ij}}{\frac{1}{2\tau}(\theta_j-\theta_i)^2+\frac{\tau}{2}} * \frac{\theta_j-\theta_i}{\tau} + \frac{1-\epsilon_{ij}}{\theta_j-\theta_i}\right]. \tag{27}$$

By leaving out the first and second terms in (21) that are independent of θ_j, the maximization of the likelihood term can be decomposed into n subproblems. Concerning each subproblem, we directly provide the objective function as (28) to search for the angular coordinate θ_j. Theorem 4.3 elaborates the equivalence between the two majorizations.

$$\begin{aligned} &\min_{\theta_j}\sum_{i=1}^{j}\left|\tilde{p}_{ij}-a_{ij}\right|\left|\frac{\epsilon_{ij}}{\frac{1}{2\tau}(\theta_j-\theta_i)^2+\frac{\tau}{2}} * \frac{\theta_j-\theta_i}{\tau} + \frac{1-\epsilon_{ij}}{\theta_j-\theta_i}\right| \\ &s.t.\ \ \left|\theta_j-\theta_i\right| > 0 \\ &\quad 0 < \theta_j < \pi \end{aligned} \tag{28}$$

For $\epsilon_{ij} = 1$, we have $|\theta_i - \theta_j| < \tau$. Then, an $\eta \geq 1$ must exist satisfying $\tau \geq \eta|\theta_i - \theta_j|$ for all V_i in the community of V_j, and the majorizations (29) can be approximated as follows:

$$\begin{aligned} &\min_{\theta_j}\sum_{i=1}^{j}\frac{\left|\tilde{p}_{ij}-a_{ij}\right|}{|\theta_j-\theta_i|}\left[\frac{2\epsilon_{ij}}{(1+\eta^2)}+(1-\epsilon_{ij})\right] \\ &s.t.\ \ \left|\theta_j-\theta_i\right| > 0 \\ &\quad 0 < \theta_j < \pi \end{aligned} \tag{29}$$

Evidently, the objective function in (29) includes two terms: the left-hand for node embedding, and the right-hand for community regularization. When $\eta \to 1$, the community regularization tends to disappear. As the parameter η increases, the effect of community regularization becomes strong. Specifically, for a node pair (V_i, V_j) without CCR, the right-hand term in the objective function (29) equals 1, and the community regularization is not activated, whereas the right-hand term is less than 1 for the pair with CCR, which can induce a solution with smaller phase separation.

Theorem 4.3. *Let O_1 denotes the jth subproblem of the maximum likelihood*

$$\begin{aligned} &\max_{\theta_j} a_{ij}\ln\tilde{p}_{ij}+(1-a_{ij})\ln(1-\tilde{p}_{ij}) \\ &s.t.\ \ \left|\theta_j-\theta_i\right|>0 \\ &\quad 0<\theta_j<\pi \end{aligned}$$

where

$$\tilde{p}_{ij}=\frac{1}{1+2^{-\frac{1}{T}}\left[\frac{1}{2\tau}(\theta_j-\theta_i)^2+\frac{\tau}{2}\right]^{\frac{\epsilon_{ij}}{T}}\left|\theta_j-\theta_i\right|^{\frac{1-\epsilon_{ij}}{T}}e^{\frac{r_i+r_j-R_j}{2T}}},$$

$\epsilon_{ij}=\chi_\tau\left(\left|\theta_j-\theta_i\right|\right)$ *for* $0<\tau<\pi/2$ *and* $T>0$. O_2 *denotes*

$$\begin{aligned} &\min_{\theta_j}\left|\frac{\epsilon_{ij}}{\frac{1}{2\tau}(\theta_j-\theta_i)^2+\frac{\tau}{2}}*\frac{\theta_j-\theta_i}{\tau}+\frac{1-\epsilon_{ij}}{\theta_j-\theta_i}\right|\left|\tilde{p}_{ij}-a_{ij}\right| \\ &s.t.\ \ \left|\theta_j-\theta_i\right|>0 \\ &\quad 0<\theta_j<\pi \end{aligned}$$

Then, O_1 is equivalent to O_2.

However, the optimization routine for (29) is intractable, because $\tilde{p}_{ij}$ is a nonconvex function of θ_i. Inspired by the mean-field method [58], we approximate $\tilde{p}_{ij}$ using the average denoted by $\langle\tilde{p}_{ij}\rangle$. Formula (30) shows the analytical and approximate forms of $\langle\tilde{p}_{ij}\rangle$, whose details can be found in Appendix B.

$$\begin{aligned} \langle\tilde{p}_{ij}\rangle &= \frac{1}{\pi}\int_0^\pi \frac{1}{1+(e^{\frac{r_i+r_j-R_j}{2}}\frac{\tilde{\theta}_{ij}}{2})^{\frac{1}{T}}}d\tilde{\theta}_{ij} \\ &= {}_2F_1(1,T,1+T,-(\frac{\pi}{2}e^{\frac{r_i+r_j-R_j}{2}})^{\frac{1}{T}}) \\ &\approx \frac{2T}{\sin T\pi}e^{\frac{R_j-r_i-r_j}{2}}. \end{aligned} \tag{30}$$

Then we can obtain the final majorization as (31)

$$\begin{aligned} &\min_{\theta_j}\sum_{i=1}^{j}\frac{B_{ij}}{\left|\theta_j-\theta_i\right|} \\ &s.t.\ \ \left|\theta_j-\theta_i\right|>0 \\ &\quad 0<\theta_j<\pi \end{aligned} \tag{31}$$

where $B_{ij}=\left|\langle\tilde{p}_{ij}\rangle-a_{ij}\right|\left[\frac{2\epsilon_{ij}}{(1+\eta^2)}+(1-\epsilon_{ij})\right]$. Recalling the analysis of the PSO model [58], the popularity depicted by the node degree decreases as the radial coordinate increases. This means that the nodes farther from the center of the Poincaré disc require larger similarities to build connections. Correspondingly, a larger B_{ij} derived with the larger radial coordinates in (31) would lead to a larger penalty for the phase difference when $a_{ij}=1$.

Then, the above SLR problem (31) is substituted by the equivalent majorization (32). We applied the nonlinear least-square optimization to search for the optimal solution and employed the 'lsqnonlin' solver in MATLAB for the numerical implementation.

$$\begin{aligned} &\min_{\theta_j}\sum_{i=1}^{j}\left(\frac{B_{ij}}{\theta_j-\theta_i}\right)^2 \\ &s.t.\ \ \left|\theta_j-\theta_i\right|>0 \\ &\quad 0<\theta_j<\pi. \end{aligned} \tag{32}$$

To prevent the optimization from falling into numerous local optima, the initial angle coordinates obtained using the spectral embedding [19] were implemented. The angle coordinates were estimated as $\theta=\arctan\frac{y_1}{y_2}$ according to the conformal property [52], where the eigenvectors (y_1 and y_2) correspond to the first two smallest non-zero eigenvalues of the Laplacian matrix.

Algorithm 1 SLR programming for solving CPHE.

Input: The adjacent matrix $\{a_{ij}\}_{n\times n}$, the number of communities L

Output: The polar coordinates of nodes $(\boldsymbol{r},\boldsymbol{\theta})$, the membership matrix $\{w_{ij}\}_{n\times L}$

```
Set the model temperature T, learning parameter δ, topological threshold υ, and geometric threshold τ.
Estimate m using mean degree.
Estimate β by fitting the degree distribution.
Estimate the radial threshold {R_j}_{1×n} using formula (10). Estimate radial coordinates r.
Initialize θ^(0), t = 0.
While not converged do
    Fit the θ^(t) according to GMM using (24)–(26)
    Update the membership matrix {w_ij}^(t)_{n×L}
    for j = 1 to n do
        Estimate θ_j^(t+1) using the nonlinear least square optimization in (32)
    Check the convergence conditions |L*(t+1) − L*(t)|
    t = t + 1
```

The SLR programming for the proposed CPHE is summarized in Algorithm 1.

Following Algorithm 1, the coordinates are inferred by minimizing the square of the gradients of $\tilde{\mathcal{L}}$ in various directions. For arbitrary node V_j, the nonlinear least-squares optimization leads to a $O(Ij)$ computational complexity, where I denotes the iterations and j corresponds to the term numbers of the derivative of the target function (32). Therefore, the total complexity of the algorithm is $O(In^2)$.

5. Experiments and analysis

In this section, the performance of the proposed CPHE was tested by conducting a series of simulated experiments on synthetic and real-world datasets. The parameter settings of the proposed models were analyzed.

In the experiments, eighteen networks, including six groups of synthetic networks (generated using the nPSO [69]), six labeled real-world networks (BSN [43], CiteSeer [70], TerroristRel [71], BlogCatalog [72], polblogs [73] and UserBehavior [74]), and six unlabeled real-world networks (Facebook networks of six American institutions [75]), are implemented to test the performance of the proposed CPHE on community embedding. Detailed information on these networks are introduced before each experiment.

A comparative analysis of the proposed models was conducted using state-of-the-art embedding methods, including hyperbolic space-based methods, such as EE [20], LaBNE [18], Mercator [21], CHM [25], Coalescent [19], and Euclidean space-based methods, such as ComE [24], DGI [46], and M-NMF [31]. A GMM with identical parameters was applied on the inferred angular coordinates to test the performance of the community embedding. Six evaluation metrics, including the accuracy, normal mutual information, phase separation ratio, Rand index, modularity, and area under the receiver operating characteristic curve, were considered to quantify the outcomes.

5.1. Comparative experiments on synthetic networks

Six groups of synthetic networks (SN) were generated by nPSO [69] with the parameters listed in Table 2. To facilitate the analysis and reduce random errors, we only generated networks using the mixed Gaussian distribution of the angular coordinates, and each dataset comprised 20 networks generated with the

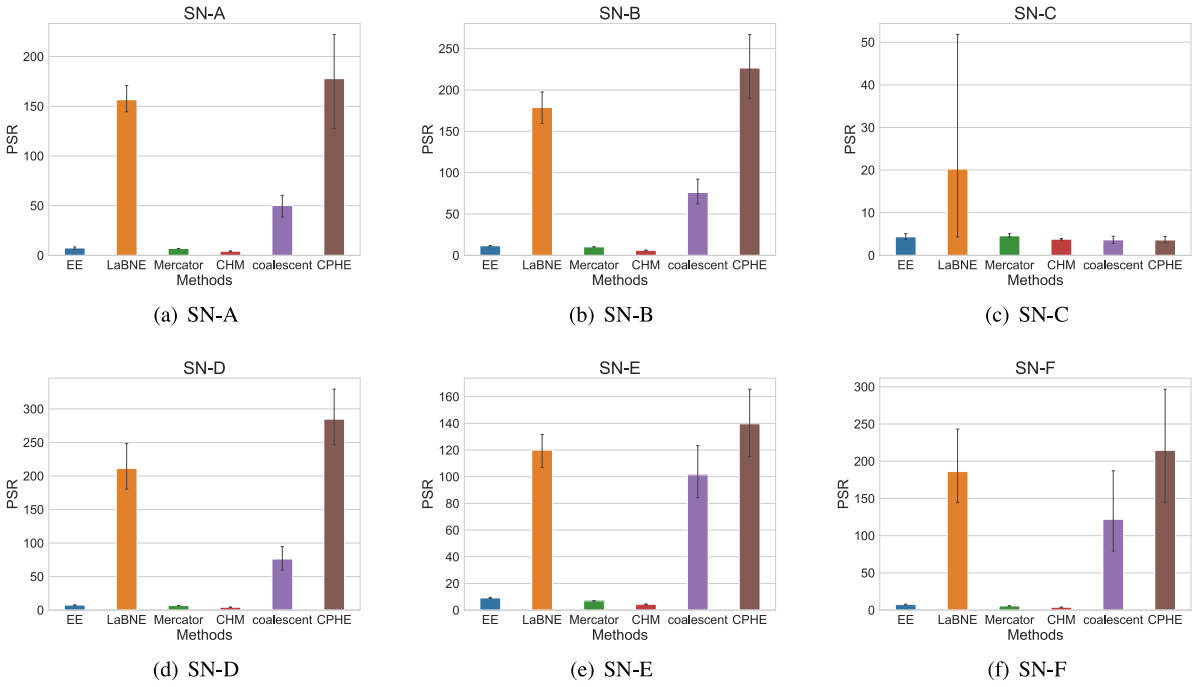


Fig. 3. The PSR of between-community and within-community on synthetic networks.

Table 2
Parameters setting for network generation in the nPSO model.

Datasets	n[a]	L[b]	$\langle\sigma\rangle$[c]	m[d]	T[e]	Community types[f]
SN-A	2000	5	$\pi/30$	5	0.1	Balanced
SN-B	2000	8	$\pi/30$	5	0.1	Balanced
SN-C	2000	5	$\pi/6$	5	0.1	Balanced
SN-D	2000	5	$\pi/30$	8	0.1	Balanced
SN-E	2000	5	$\pi/30$	5	0.3	Balanced
SN-F	2000	5	$\pi/30$	5	0.1	Unbalanced

[a]Number of nodes in the generated network.
[b]Number of communities in the generated network.
[c]Mean value of standard deviation of the angular coordinates.
[d]Half of the average node degree.
[e]Temperature of the generative model.
[f]Types of communities controlled by the global membership parameter W_l. 'Balanced' refers to an equal-scaled community division, whereas 'Unbalanced' refers to unequal case.

Table 3
Average accuracy of the community detection on synthetic networks.

Methods	SN-A	SN-B	SN-C	SN-D	SN-E	SN-F
ComE	0.7111	0.5467	0.5231	0.7338	0.7450	0.8390
DGI	0.4115	0.3524	0.3931	0.5561	0.4892	0.5731
M-NMF	0.3819	0.3763	0.4144	0.4438	0.4762	0.5103
EE	0.8706	0.6647	0.5491	**0.8544**	0.7935	0.7890
LaBNE	0.8689	0.6626	0.6926	0.8436	0.7550	0.6230
Mercator	0.8051	0.7156	0.6006	0.7407	0.7143	0.9195
CHM	0.6009	0.5707	0.5239	0.5365	0.7318	0.5561
Coalescent	0.8103	0.6688	0.6566	0.7002	0.7794	0.8992
CPHE	**0.9440**	**0.7620**	**0.6927**	0.8461	**0.8280**	**0.9425**

same parameters. Among all datasets, SN-A was the baseline, and the others were control groups constructed by changing one of the five parameters.

Networks in each group were embedded using the eight aforementioned methods and the proposed CPHE model. To reduce sampling bias, each result was obtained after 10 runs using the same model parameters. The GMM was applied to each representation vector to fit the membership matrices, and the cluster identification corresponding to the maximum membership degree is taken as the node's community label. To geometrically demonstrate the community-preserving performance of CPHE, we summed the phase separation of node pairs from same and different communities. Fig. 3 presents the phase separation ratio (PSR) of the between-community and within-community node pairs.

A large PSR indicates a compact spatial cluster of the representation vectors. In Fig. 3, the CPHE highlights the angle gaps of different communities compared with classical likelihood approaches. This indicates that the novel metric $\tilde{\theta}_{ij}$ brings the connected pairs from the common community closer during embedding. However, the CPHE performs weaker preservation on networks with an indistinct community (such as SN-C) than on a significant community.

Furthermore, Tables 3–4 list the average accuracies and NMIs obtained from the labels according to the original partitions. Evidently, CPHE obtains the highest accuracies and NMIs on the five groups of datasets. The proposed embedding method preserves most of the community structure during the embedding. Overall, most features of the generated networks, such as the expected degree, temperature, and community type, have a minimal effect on the community-preserving embedding, but the community number and standard deviation of the distribution produce significant fluctuations in its performance. Because the range of angular coordinates in the generated model is fixed and the larger community number and standard deviation of the distribution lead to a smaller angular distance between communities, the community structure of these vectors would be more indistinct. Thus, all embedding methods received a weaker performance in community preservation on the corresponding datasets (SN-B and SN-C). In addition, the poor performance of Euclidean space-based methods on hyperbolic generative networks indicates the importance of selecting the embedding space.

5.2. Node classification

According to [24], nodes from a common community tend to have the same characteristic labels. The community structure can impact the classification of nodes. Thus, we tested the

Table 4
Average NMI of the community detection on synthetic networks.

Methods	SN-A	SN-B	SN-C	SN-D	SN-E	SN-F
ComE	0.8158	0.7233	0.5942	0.8106	0.8490	0.8613
DGI	0.3325	0.3154	0.3447	0.3419	0.3765	0.4677
M-NMF	0.2774	0.3043	0.2981	0.2716	0.3151	0.3403
EE	0.8810	0.7368	0.6458	0.9138	0.8105	0.8041
LaBNE	0.9019	0.6878	**0.6549**	0.9105	0.8206	0.6426
Mercator	0.8269	0.7320	0.6061	0.7813	0.7621	0.8755
CHM	0.5666	0.6026	0.5129	0.5745	0.6911	0.5211
Coalescent	0.8694	0.7044	0.6421	0.8774	0.7950	0.8968
CPHE	**0.9057**	**0.7473**	0.6426	**0.9284**	**0.8834**	**0.8989**

Table 5
Parameters of the labeled real-world networks.

Network	n[a]	GCS[b]	E[c]	L[d]	CC[e]
BSN	8808	8808	1411947	9	administrative region
CiteSeer	3312	2110	7338	6	theme
TerroristRel	881	687	13768	2	whether contact
BlogCatalog	5196	5196	343486	7	interest
polblogs	1490	1222	33428	2	political view
UserBehavior	7716	7713	166495	5	merchandise category

[a]Number of nodes in the network.
[b]Number of nodes in the giant component of the network.
[c]Number of edges in the giant component of the network.
[d]Number of communities in the network.
[e]Categorical characteristic.

performance of the community-preserving embedding by implementing the node classification experiments on six real-world networks. These networks are introduced below, and Table 5 lists their parameters.

The base station network (BSN) [43] includes the Internet access behavior of subscribers mined from the UDR data of cellular communication in a Chinese city (Jinhua) for three weeks. Two base stations (nodes) are connected if personnel exchanges during the period, and each base station is labeled with one of the nine administrative regions in the city. The CiteSeer network [70] was constructed with the six themes of scientific publication and the reference relationships among them. The TerroristRel [71] contains relationships between terrorists and their contact information. BlogCatalog [72] is a network depicting the social relationships of the bloggers listed on the BlogCatalog website. The labels of the vertices represent bloggers' interests inferred using the metadata provided by the bloggers. Polblogs [73] is a directed network of hyperlinks between weblogs about US politics. Nodes in polblogs are divided into two classes according to their political views, and we construct a symmetric adjacent matrix to make the network undirected. UserBehavior [74] comprises 10568 users' 'cart' behavior on Taobao. Each node denotes merchandise, and an edge is produced between merchandises when a user adds both to their cart.

The eight aforementioned embedding methods were applied as baselines to quantify the homogeneity of the networks. The labels of the nodes are assigned to the embedding vectors for supervised learning. Specifically, a multiclass support vector machine is implemented to train the classifier. The accuracy of the 10-fold cross-validation in Table 6 reveals the separability of representation vectors.

From Table 6, all methods preserved the community structure well on the BSN, TerroristRel, CiteSeer, and polblogs, but not on UserBehavior and BlogCatalog, and the weak homogeneity of these networks seems to be the primary reason behind the results. On this basis, the geometric representation mapped by CPHE shows stronger separability than the baseline representations in most of the aforementioned real-world networks.

5.3. Community detection

Six unlabeled Facebook networks were implemented for community detection using CPHE and the other hyperbolic embedding methods. The nodes in each network stand for users from American institutions, and the links express existing friendships between their pages in September 2005. For each node, seven characteristics, i.e., status, gender, major, minor, residence, year, and high school, were recorded for social community comparisons under Boyd's assumption [76]. We selected the characteristic (shown in Table 7) corresponding to the highest assortativity as the baseline, according to the z-scores in [75].

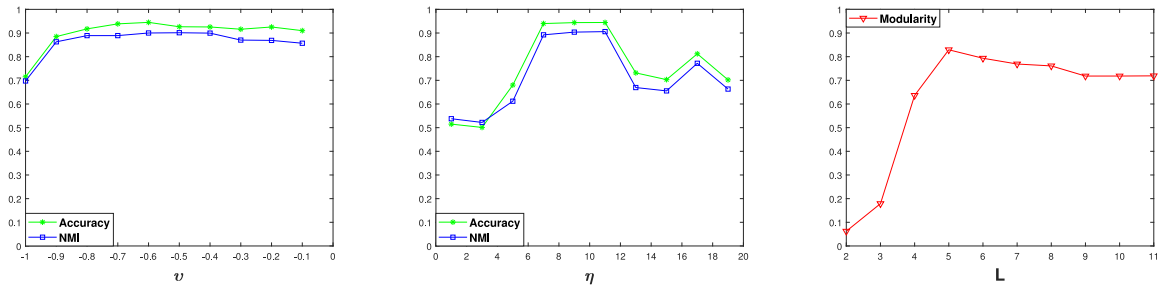
Similarly, we embedded Facebook networks with the aforementioned baseline methods and detected the community by learning the mixed Gaussian distribution of the embedding vectors. For each comparison, we calculated the Rand index in terms of $[M3 + (M - M1 - M2 + M3)]/M$ [75], where M is the total number of pairs in the network, $M1$ is the number of pairs within a community, $M2$ is the number of pairs with the same categorical value, and $M3$ denotes the number of pairs with the same community and categorical value. Table 8 presents the quantitative results of the embedding performance using the Rand index.

From Table 8, the proposed model brings out the highest RIs: 0.7660 for Caltech, 0.7312 for UCSC-male, 0.9133 for Amherst, 0.8992 for Bowdoin, and 0.7457 for Reed. The partitions obtained by the CPHE showed strong assortativity with the categorical characteristics. In addition, the modularity was calculated to quantify the quality of community partitions for these unlabeled networks, as shown in Table 9. The CPHE method yields the highest modularity, and the values are greater than 0.3 in most Facebook networks. This means that the coordinates embedded by the CPHE have the strongest ability to represent the community structure of these Facebook networks. Reed, whose community number is large, but the number of nodes is small, has weak modularity. Thus, each method achieves a modularity less than 0.3. Then, by combining Tables 8 and 9, the partitions fitted by CHM achieve good modularity, although the value of RI is close to that of other baseline methods. This is because the CHM must find the highly modular community partitions to guide the hyperbolic embedding. However, the proposed CPHE still achieved the highest modularity in the five datasets. To some extent, these results demonstrated that the alignment between community and spatial clusters contributes to community preservation.

5.4. Link prediction

To test the ability of preserving latent affinities during network embedding, we attempt to estimate the likelihood of the existence of a link using the obtained embedding vectors, i.e., link prediction. Specifically, we mapped the representation vectors obtained by these nine embedding methods into Euclidean space and then predicted the link score by calculating the cosine similarities of the transformed vectors. The area under the receiver operating characteristic (AUROC) curve was used to estimate the link prediction performance of the embedding methods, and Tables 10 and 11 present the results.

According to Tables 10 and 11, the proposed CPHE obtained the largest AUROCs on most datasets. In addition, other community preservation-based methods such as ComE and M-NMF also achieve good AUROCs, which suffices to show the great influence of hidden affinities on link prediction.



(a) Relationships between η and performance of community detection using CPHE

(b) Relationships between υ and performance of community detection using CPHE

(c) Relationships between L and modularity of community detection using CPHE

Fig. 4. Parameter analysis in CPHE on the baseline synthetic networks.

Table 6
Accuracies of the node classification on real-world networks.

Methods	BSN	CiteSeer	TerroristRel	BlogCatalog	polblogs	UserBehavior
ComE	0.8063	0.5100	0.8457	0.3699	0.6318	0.3239
DGI	0.2040	0.2630	0.7569	0.2005	0.7995	0.2447
M-NMF	0.9220	0.3962	0.8006	**0.6790**	0.9190	0.3441
EE	0.9123	0.5322	0.8326	0.3099	0.9362	0.3301
LaBNE	0.8471	0.5299	**0.8937**	0.2602	0.9002	0.2839
Mercator	0.9185	0.5147	0.8879	0.3543	0.9525	0.3145
CHM	0.8975	0.6109	0.8574	0.4546	0.9435	0.3346
Coalescent	0.9106	0.5270	0.8836	0.3433	0.9517	0.3068
CPHE	**0.9248**	**0.6182**	0.8821	0.4314	**0.9566**	**0.3511**

Table 7
Details of the Facebook networks.

Network	n^a	GCS[b]	E^c	L^d	HAC[e]
Caltech	769	762	33302	8	Residence
Rice	4087	4083	369652	9	Residence
UCSC-male	3708	3562	77436	10	Residence
Amherst	2235	2235	181908	15	Year
Bowdoin	2252	2250	168772	18	Year
Reed	962	962	37624	17	Year

[a]Number of nodes in the network.
[b]Number of nodes in the giant component of the network.
[c]Number of edges in the giant component of the network.
[d]Number of communities in the network.
[e]Characteristic with highest assortativity.

Table 8
The RI of the community partition on Facebook networks.

Methods	Caltech	Rice	UCSC-male	Amherst	Bowdoin	Reed
ComE	0.6719	**0.8331**	0.7185	0.8211	0.8991	0.7344
DGI	0.7115	0.7660	0.6970	0.6277	0.8180	0.6731
M-NMF	0.7532	0.8034	0.7297	0.8515	0.8449	0.7454
EE	0.7419	0.7914	0.7232	0.8977	0.8423	0.7484
LaBNE	0.6220	0.6173	0.3906	0.4825	0.5501	0.5742
Mercator	0.7421	0.8174	0.7003	0.8004	0.8696	0.7170
CHM	0.7055	0.7651	0.7242	0.8928	0.8848	0.7258
Coalescent	0.7165	0.8269	0.7153	0.8615	0.8433	0.7379
CPHE	**0.7660**	0.7855	**0.7312**	**0.9133**	**0.8992**	**0.7457**

5.5. Parameter analysis

In the proposed model, two new parameters exist for community representation to be set in advance: the geometric threshold η and topological threshold υ. The geometric threshold controls the encoding of the geometric community, whereas the topological threshold affects the transformation of the membership matrix. To avoid the impact of model temperature (T) on the embedding performance, the baseline synthetic dataset (SN-A) in Section 5.1 was embedded by CPHE for the experimental analysis of parameter selections.

Table 9
Modularity of the community partition on Facebook networks.

Methods	Caltech	Rice	UCSC-male	Amherst	Bowdoin	Reed
ComE	0.2171	0.1861	0.2574	0.2934	0.3375	**0.2346**
DGI	0.0015	0.0027	0.1031	0.1165	0.0002	0.0001
M-NMF	0.1767	0.1558	0.2048	0.2113	0.1841	0.1247
EE	0.2241	0.2830	0.2831	0.3058	0.2836	0.2063
LaBNE	0.1084	0.1114	0.0561	0.1185	0.1714	0.0412
Mercator	0.1586	0.1591	0.2535	0.2548	0.2318	0.1182
CHM	0.3843	0.4221	0.4416	0.4075	0.4204	0.2084
Coalescent	0.1682	0.2333	0.2165	0.1828	0.2171	0.1346
CPHE	**0.3886**	**0.4034**	**0.4668**	**0.4190**	**0.4431**	0.2116

Let $\eta \in [1, +\infty)$ and $\upsilon \in [-1, 0)$ from the CPHE increase by 2 and 0.1, respectively, and we change one parameter while fixing the others to assess the influence of each parameter on the performance of community preservation. Similar to the above experiments, CPHE was run 10 times on each network in the SN-A dataset, and the average clustering accuracies and average NMIs are recorded and displayed in Fig. 4.

Evidently, the clustering accuracies and NMIs in Fig. 4(a) first increase and then decrease with an increasing η. Controlled by the threshold η, the undersize and oversize of the geometric community produce fluctuations of the clustering accuracy and NMI of the embedding representations. Community regularization weakens for small values of η, whereas large values of η lead to the overregulation of unconnected nodes. For networks in SN-A, the critical point of the community overregulation ranges from 6 to 12.

From Fig. 4(b), the lowest accuracies and NMIs lie on $\upsilon = -1$, corresponding to the unactivated CCR regularization. Further, the change in the values of the geometric threshold (η) has a stronger influence on accuracies and NMIs of community detection than the topological threshold (υ) in CPHE. This is consistent with our intuition that when the community structure is evident, the sum

Table 10
AUROC of the link prediction on real-world networks.

Methods	BSN	CiteSeer	TerroristRel	BlogCatalog	polblogs	UserBehavior
ComE	0.8626	0.9111	0.9350	0.7455	0.7759	0.7115
DGI	0.6953	0.7207	0.7167	0.6354	0.6271	0.5971
M-NMF	0.8879	0.6288	0.8537	**0.8067**	0.7233	0.7171
EE	0.8789	0.9503	0.9309	0.7146	0.8318	0.6146
LaBNE	0.8181	0.9388	**0.9543**	0.6541	0.7745	0.6079
Mercator	0.8665	0.9566	0.9480	0.6947	0.7741	0.6682
CHM	0.8869	0.9547	0.9358	0.7288	0.8693	0.7005
Coalescent	0.8809	0.9408	0.9515	0.7026	0.7717	0.6534
CPHE	**0.8992**	**0.9608**	0.9249	0.7541	**0.8728**	**0.7311**

Table 11
AUROC of the link prediction on Facebook networks.

Methods	Caltech	Rice	UCSC-male	Amherst	Bowdoin	Reed
ComE	0.7231	0.7484	0.7944	0.8001	0.8007	0.7661
DGI	0.6045	0.5954	0.6018	0.5988	0.5979	0.6858
M-NMF	0.7693	0.7403	0.7970	**0.8466**	0.8098	0.7673
EE	0.7544	0.7467	0.7830	0.8101	0.8019	0.7170
LaBNE	0.6241	0.6865	0.6245	0.6928	0.7334	0.6832
Mercator	0.7317	0.7254	0.7807	0.7808	0.7955	0.7353
CHM	0.7547	0.7413	0.7939	0.7732	0.7836	0.7664
Coalescent	0.7308	0.7275	0.7927	0.7733	0.7694	0.7351
CPHE	**0.7761**	**0.7643**	**0.8012**	0.7968	**0.8166**	**0.8153**

of the membership product for the construction community co-occurrence relations in (16) approaches 1 or 0. In addition, the topological threshold υ has a minimal impact on the generation of community-level cut sets.

A high accuracy and NMI in community detection will be achieved by CPHE if η is set in the proximity of its critical points, which can be obtained by experiments and empiricism. Intuitively, η can be set smaller for networks with more communities and vice versa. A more precise setting depends on the inflection point of the community-preserving performance in the embedding experiments.

Another important parameter in implementing CPHE is the number of communities, denoted by L in this study. Although unknown in most network analyses, this parameter strongly influences the performance of CPHE, as shown in Fig. 4(c). While fixing the other parameters in CPHE, i.e., T, η, and υ, we increase L from 2 to 11 and implement CPHE for 10 runs on each network in the SN-A dataset. The modularity of the obtained partition was calculated to evaluate the performance of CPHE. The results show that the highest modularity is achieved at $L = 5$, which exactly matches the number of communities in the network generation model. Moreover, due to the sparsity of data agglomerating the communities cause more modularity loss than splitting a community. Thus, the modularity rapidly increases and then slowly decreases as L increases. This phenomenon helps us determine the practical L of CPHE; more specifically, the number of communities corresponding to the highest modularity can be set as L in the parameter adjustments of CPHE.

6. Conclusions

In this study, a novel hyperbolic embedding model is proposed to replay the hyperbolic growth of a network while preserving the community structure. In contrast to existing hyperbolic embedding models, we incorporated the latent affinities hidden in the community into the bivariate adjacency relations using the transformation of co-occurrence relations. The community and adjacencies were preserved in the embedding process. Next, we combined community fitting and node embedding to construct a hybrid likelihood objective function. The unsupervised embedding, based on this function, learns representation vectors under the constraint of the community structure. Finally, to avoid smoothing of the coordinate distribution caused by uniform sampling in the classical numerical solution, the sum of linear ratios programming is implemented. With the proposed model, the community distribution and node representation were updated to the optimum along the fitting gradient and embedding gradient, respectively. The community topology of the networks was well preserved in the form of a spatial cluster for downstream analysis. The efficiency of the proposed methods was verified through extensive experiments of node classification and community detection on six synthetic datasets and eleven real-world networks. The essential parameters for community quantification in the proposed models are analyzed, and a suggested strategy for selecting parameters is presented.

We conclude the paper by reiterating the extension potentials. Recall that we first investigated community preservation in hyperbolic embedding using CCR regularization. Regularization reveals the latent affinities in many network mining applications, particularly link prediction. In addition, because the proposed CCR regularized angle metric is differentiable, the solution to hyperbolic embedding can also be implemented using other approaches, such as gradient descent. Furthermore, the proposed CPHE handles networks with distinguishable and number-known communities well with the proper parameters according to the results of the experiments. Thus, this potentially enables us to evaluate the number of communities for an unsupervised network.

CRediT authorship contribution statement

Dongsheng Ye: Methodology, Software, Writing – original draft, Visualization. **Hao Jiang:** Conceptualization, Supervision. **Ying Jiang:** Formal analysis. **Qiang Wang:** Data curation, Validation, Investigation. **Yulin Hu:** Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (NSFC) under Grant U19B2004; in part by the Science and Technology Program of Guangzhou, China, under Grant 201804020053; in part by Zhongshan High-end Scientific Research Institutions Special Innovation Program, China under Grant 181129112748101; and in part by Guangdong "Major Project and Task List" Program, China under Grant 2019sdr002.

Appendix A. Proof of Theorem 4.1– Theorem 4.3

Proof of Theorem 4.1. Obviously, θ_i and θ_j is commutative in $\tilde{\theta}_{ij}$. The differentiability of the function with respect to θ_j can be evidenced by the same token for θ_i below.

$$\begin{cases} \left.\frac{\partial\tilde{\theta}_{ij}}{\partial\theta_i}\right|_{\theta_i-\theta_j\to\tau^-} = 1 = \left.\frac{\partial\tilde{\theta}_{ij}}{\partial\theta_i}\right|_{\theta_i-\theta_j\to\tau^+} \\ \left.\frac{\partial\tilde{\theta}_{ij}}{\partial\theta_i}\right|_{\theta_j-\theta_i\to\tau^-} = -1 = \left.\frac{\partial\tilde{\theta}_{ij}}{\partial\theta_i}\right|_{\theta_j-\theta_i\to\tau^+} \end{cases}. \quad \blacksquare \tag{A.1}$$

Proof of Theorem 4.2. Since $\tau > 0$, $\forall\ \theta_i, \theta_i$ we have

$$\theta_{ij}^{IN} = \frac{1}{2\tau}\left[(\theta_i-\theta_j)^2+\tau^2\right] \geq \frac{1}{2\tau} * 2|\theta_i-\theta_j|\tau = \theta_{ij}^{OUT}. \quad \blacksquare \tag{A.2}$$

Proof of Theorem 4.3. Let $\tilde{\mathcal{L}}_i = a_{ij}\ln\tilde{p}_{ij} + (1-a_{ij})\ln(1-\tilde{p}_{ij})$. Then

$$\frac{\partial\tilde{\mathcal{L}}_i}{\partial\theta_j} = \frac{1}{T}(\tilde{p}_{ij}-a_{ij})\left[\frac{\epsilon_{ij}}{\frac{1}{2\tau}(\theta_j-\theta_i)^2+\frac{\tau}{2}} * \frac{\theta_j-\theta_i}{\tau} + \frac{1-\epsilon_{ij}}{\theta_j-\theta_i}\right]. \tag{A.3}$$

We denote $\frac{\partial\tilde{\mathcal{L}}_i}{\partial\theta_j}$ by $Q_i(\theta_j)$. Obviously, when there is an edge between node V_i and node V_j which corresponds to $a_{ij} = 1$, we have

$$\begin{cases} Q_i(\theta_j) > 0, & \text{if } \theta_j < \theta_i \\ Q_i(\theta_j) < 0, & \text{if } \theta_j > \theta_i \end{cases}$$

Substitute the solution $\theta_j^* \to \theta_i$ of the majorization O_1 into $Q_i(\theta_j)$, we have $|Q_i(\theta_j^*)| = \min|Q_i(\theta_j)|$ as $\tau > 0$. Analogously, when $a_{ij} = 0$

$$\begin{cases} Q_i(\theta_j) < 0, & \text{if } \theta_j < \theta_i \\ Q_i(\theta_j) > 0, & \text{if } \theta_j > \theta_i \end{cases}$$

The optimal value θ_j^* of the majorization O_1 satisfies $|\theta_j^*-\theta_i| = \max|\theta_j-\theta_i|$. Thus $|Q_i(\theta_j^*)| = \min|Q_i(\theta_j)|$ for a particular value of $\tau < \pi/2 \leq \max\{\pi-\theta_i, \theta_i-0\}$. In summary, the O_1's solution is also the solution to (A.4) and vice versa.

$$\begin{aligned} &\min_{\theta_j}\left|Q_i(\theta_j)\right| \\ &s.t.\ \left|\theta_j-\theta_i\right| > 0 \\ &\quad 0 < \theta_j < \pi \end{aligned} \tag{A.4}$$

As $T > 0$, O_1 is equivalent to O_2. $\blacksquare$

Appendix B. Derivation of formula (30)

We now provide the full details of formula (30). Let $A = \frac{1}{2}e^{\frac{r_i+r_j-R_j}{2}}$ and it is obvious that the definite integral in (30) can be transformed into the hypergeometric series as follows:

$$\begin{aligned} \langle\tilde{p}_{ij}\rangle &= \frac{1}{\pi}\int_0^\pi \frac{1}{1+(A\tilde{\theta}_{ij})^{\frac{1}{T}}}d\tilde{\theta}_{ij} \\ &= \frac{1}{A\pi}\int_0^\pi \sum_{N=0}^{+\infty}\left[-(A\tilde{\theta}_{ij})^{\frac{1}{T}}\right]^N dA\tilde{\theta}_{ij} \\ &= \frac{1}{A\pi}\sum_{N=0}^{+\infty}(-1)^N\frac{T}{N+T}(A\pi)^{\frac{N}{T}+1} \\ &= \sum_{N=0}^{+\infty}\frac{\Gamma(T+N)}{\Gamma(T)}\frac{\Gamma(T+1)}{\Gamma(T+N+1)}\frac{N!}{N!}\left[-(A\pi)^{\frac{1}{T}}\right]^N \\ &= {}_2F_1(1,T,1+T,-(A\pi)^{\frac{1}{T}}). \end{aligned} \tag{B.5}$$

Formula (B.5) is equivalent to the value of hypergeometric function ${}_2F_1(1,T,1+T,-X^{\frac{1}{T}})$ when $X = A\pi$. We have

$$\begin{aligned} &{}_2F_1(1,T,1+T,-X^{\frac{1}{T}}) \\ &= -\frac{\mathrm{i}\Gamma(T+1)e^{-\mathrm{i}\pi T}}{2\Gamma(T)\Gamma(1)\sin T\pi}\int_1^{(0+)} S^{T-1}(1+SX^{\frac{1}{T}})^{-1}dS, \end{aligned} \tag{B.6}$$

where i is the imaginary number. Let $\mathcal{I}$ denotes the contour integral as (B.7) and we have

$$\begin{aligned} \mathcal{I} &= \int_1^{(0+)} S^{T-1}(1+SX^{\frac{1}{T}})^{-1}dS \\ &= \mathcal{I}_1+\mathcal{I}_2+\mathcal{I}_3, \end{aligned} \tag{B.7}$$

where

$$\mathcal{I}_1 = \lim_{\varepsilon\to 0}\int_1^{\varepsilon} S^{T-1}(1+SX^{\frac{1}{T}})^{-1}dS, \tag{B.8}$$

$$\mathcal{I}_2 = \lim_{\varepsilon\to 0}\int_0^{2\pi} -\frac{\mathrm{i}(\varepsilon e^{\mathrm{i}\varpi})^T}{1+\varepsilon e^{\varpi\mathrm{i}}X^{\frac{1}{T}}}d\varpi, \tag{B.9}$$

$$\mathcal{I}_3 = \lim_{\varepsilon\to 0}\int_\varepsilon^1 e^{2\pi T}S^{T-1}(1+SX^{\frac{1}{T}})^{-1}dS. \tag{B.10}$$

Obviously, when $\left|X^{\frac{1}{T}}\right| < +\infty$, $\mathcal{I}_2 = 0$. Then

$$\begin{aligned} \mathcal{I} &= (e^{2\pi T}-1)\int_0^1 S^{T-1}(1+SX^{\frac{1}{T}})^{-1}dS \\ &= 2ie^{\pi T\mathrm{i}}\sin T\pi\cdot X^{-1}\int_0^{X^{\frac{1}{T}}} S^{T-1}(1+SX^{\frac{1}{T}})^{-1}dS. \end{aligned} \tag{B.11}$$

According to the Euler's Reflection Formula, $\mathcal{I}$ is approximated as follows

$$\mathcal{I} \approx 2ie^{\pi T\mathrm{i}}\pi X^{-1}. \tag{B.12}$$

Substitute (B.12) into (B.6), we have

$${}_2F_1(1,T,1+T,-X^{\frac{1}{T}}) \approx \frac{T\pi}{X\sin T\pi}. \tag{B.13}$$

Thus

$$\begin{aligned} \langle\tilde{p}_{ij}\rangle &\approx \frac{T\pi}{A\pi\sin T\pi} \\ &= \frac{2T}{\sin T\pi}e^{\frac{R_j-r_i-r_j}{2}}. \end{aligned} \tag{B.14}$$

References

[1] Sepideh Bazzaz Abkenar, Ebrahim Mahdipour, Seyed Mahdi Jameii, Mostafa Haghi Kashani, A hybrid classification method for twitter spam detection based on differential evolution and random forest, Concurr. Comput.: Pract. Exper. 33 (21) (2021) e6381.

[2] Marián Boguñá, Ivan Bonamassa, Manlio De Domenico, Shlomo Havlin, Dmitri Krioukov, M. Ángeles Serrano, Network geometry, Nat. Rev. Phys. 3 (2021) 114–135.

[3] Kitsak Maksim, Voitalov Ivan, Krioukov Dmitri, Link prediction with hyperbolic geometry, Phys. Rev. Res. 2 (4) (2020).

[4] Adnan Zeb, Anwar Ul Haq, Junde Chen, Zhenfeng Lei, Defu Zhang, Learning hyperbolic attention-based embeddings for link prediction in knowledge graphs, Knowl.-Based Syst. (2021) 107369.

[5] Hoorn Pim Van Der, Lippner Gabor, Krioukov Dmitri, Sparse maximum-entropy random graphs with a given power-law degree distribution, J. Stat. Phys. 173 (3–4) (2018) 806–844.

[6] Dmitri Krioukov, Clustering implies geometry in networks, Phys. Rev. Lett. 116 (20) (2016) 208302.

[7] Marián Boguñá, Dmitri Krioukov, Pedro Almagro, M. Ángeles Serrano, Small worlds and clustering in spatial networks, Phys. Rev. Res. 2 (2020) 023040.

[8] Konstantin Zuev, Marián Boguñá, Ginestra Bianconi, Dmitri Krioukov, Emergence of soft communities from geometric preferential attachment, Sci. Rep. 5 (2015) 9421.

[9] M. Ángeles Serrano, Marián Boguñá, Francesc Sagués, Uncovering the hidden geometry behind metabolic networks, Mol. Biosyst. 8 (3) (2012) 843–850.
[10] Muscoloni Alessandro, Cannistraci Carlo Vittorio, Leveraging the nonuniform pso network model as a benchmark for performance evaluation in community detection and link prediction, New J. Phys. 20 (6) (2018) 063022.
[11] Muscoloni Alessandro, Cannistraci Carlo Vittorio, Navigability evaluation of complex networks by greedy routing efficiency, Proc. Natl. Acad. Sci. 116 (2019) 1468–1469.
[12] András Gulyás, József J. Bíró, Attila Kőrösi, Gábor Rétvári, Dmitri Krioukov, Navigable networks as nash equilibria of navigation games, Nat. Commun. 6 (2015) 7651.
[13] Allard Antoine, Serrano M. Ngeles, Navigable maps of structural brain networks across species, PLOS Comput. Biol. 16 (2020).
[14] Kevin Verbeek, Subhash Suri, Metric embedding, hyperbolic space, and social networks, in: Proceedings of the Thirtieth Annual Symposium on Computational Geometry, 2014, pp. 501–510.
[15] Wang Zhang, Xuan Guo, Wenjun Wang, Qiang Tian, Lin Pan, Pengfei Jiao, Role-based network embedding via structural features reconstruction with degree-regularized constraint, Knowl.-Based Syst. 218 (2021) 106872.
[16] Fragkiskos Papadopoulos, Constantinos Psomas, Dmitri Krioukov, Network mapping by replaying hyperbolic growth, IEEEACM Trans. Netw. 23 (1) (2015) 198–211.
[17] Fragkiskos Papadopoulos, Rodrigo Aldecoa, Dmitri Krioukov, Network geometry inference using common neighbors, Phys. Rev. E 92 (2) (2015) 022807.
[18] Gregorio Alanis-Lobato, Pablo Mier, Miguel A. Andrade-Navarro, Efficient embedding of complex networks to hyperbolic space via their laplacian, Sci. Rep. 6 (1) (2016) 30108.
[19] Alessandro Muscoloni, Josephine Maria Thomas, Sara Ciucci, Ginestra Bianconi, Carlo Vittorio Cannistraci, Machine learning meets complex networks via coalescent embedding in the hyperbolic space, Nat. Commun. 8 (1) (2017) 1615.
[20] Thomas Bläsius, Tobias Friedrich, Anton Krohmer, Sören Laue, Efficient embedding of scale-free graphs in the hyperbolic plane, IEEE/ACM Trans. Netw. 26 (2) (2018) 920–933.
[21] Guillermo García-Pérez, Antoine Allard, M. Ángeles Serrano, Marián Boguñá, Mercator: uncovering faithful hyperbolic embeddings of complex networks, New J. Phys. 21 (12) (2019).
[22] Santo Fortunato, Community detection in graphs, Phys. Rep. 486 (3–5) (2010) 75–174.
[23] Shreekanth M. Prabhu, Natarajan Subramanyam, Rhythm Girdhar, Containing covid-19 pandemic using community detection, J. Phys.: Conf. Ser. 1797 (2021) 012008.
[24] Sandro Cavallari, Vincent W. Zheng, Hongyun Cai, Kevin Chen-Chuan Chang, Erik Cambria, Learning community embedding with community detection and node embedding on graphs, in: The 2017 ACM on Conference on Information and Knowledge Management, CIKM, 2017, pp. 377–386.
[25] Zuxi Wang, Qingguang Li, Fengdong Jin, Wei Xiong, Yao Wu, Hyperbolic mapping of complex networks based on community information, Physica A 455 (2016) 104–119.
[26] Zuxi Wang, Yao Wu, Qingguang Li, Fengdong Jin, Wei Xiong, Link prediction based on hyperbolic mapping with community structure for complex networks, Physica A 450 (2016) 609–623.
[27] Maximilian Nickel, Douwe Kiela, Poincaré embeddings for learning hierarchical representations, 2017, arXiv preprint arXiv:1705.08039.
[28] Thomas Gerald, Hadi Zaatiti, Hatem Hajri, Nicolas Baskiotis, Olivier Schwander, From node embedding to community embedding: A hyperbolic approach, 2019, arXiv preprint arXiv:1907.01662.
[29] Silvere Bonnabel, Stochastic gradient descent on Riemannian manifolds, IEEE Trans. Automat. Control 58 (9) (2013) 2217–2229.
[30] Xiaoliang Xu, Mengzhao Wang, Yuxiang Wang, Dingcheng Ma, Two-stage routing with optimized guided search and greedy algorithm on proximity graph, Knowl.-Based Syst. 229 (2021) 107305.
[31] Xiao Wang, Peng Cui, Jing Wang, Jian Pei, Shiqiang Yang, Community preserving network embedding, in: The 31st AAAI Conference on Artificial Intelligence, 2017, pp. 203–209.
[32] Guillermo García-Pérez, Marián Boguá, Allard Antoine, Serrano M. Ngeles, The hidden hyperbolic geometry of international trade: World trade atlas 1870-2013, Sci. Rep. 6 (2016) 33441.
[33] Gregorio Alanis-Lobato, Pablo Mier, Miguel A. Andrade-Navarro, Manifold learning and maximum likelihood estimation for hyperbolic network embedding, Appl. Netw. Sci. 1 (1) (2016) 10.
[34] Frank Smadja, Macrocoding the Lexicon with Co-Occurrence Knowledge, Psychology Press, 2021.
[35] S. Pare, Ashish Kumar Bhandari, Anil Kumar, Girish Kumar Singh, An optimal color image multilevel thresholding technique using grey-level co-occurrence matrix, Expert Syst. Appl. 87 (2017) 335–362.
[36] Albert Barberán, Scott T. Bates, Emilio O. Casamayor, Noah Fierer, Using network analysis to explore co-occurrence patterns in soil microbial communities, ISME J. 6 (2) (2012) 343–351.
[37] F. Guillaume Blanchet, Kevin Cazelles, Dominique Gravel, Co-occurrence is not evidence of ecological interactions, Ecol. Lett. 23 (7) (2020) 1050–1063.
[38] Ali Faqeeh, Saeed Osat, Filippo Radicchi, Characterizing the analogy between hyperbolic embedding and community structure of complex networks, Phys. Rev. Lett. 121 (9) (2018) 0983011–098301.6.
[39] Alberto Cacciola, et al., Coalescent embedding in the hyperbolic space unsupervisedly discloses the hidden geometry of the brain, 2017, arXiv preprint arXiv:1705.04192.
[40] Hong-Wei Jiao, San-Yang Liu, A practicable branch and bound algorithm for sum of linear ratios problem, Eur. J. Oper. Res. 243 (3) (2015) 723–730.
[41] Frederic Sala, Chris De Sa, Albert Gu, Christopher Ré, Representation tradeoffs for hyperbolic embeddings, in: International Conference on Machine Learning, 2018, pp. 4460–4469.
[42] Marc Law, Jos Stam, Ultrahyperbolic representation learning, Adv. Neural Inf. Process. Syst. 33 (2020) 1668–1678.
[43] Shuwen Yi, Hao Jiang, Ying Jiang, Pan Zhou, Qiang Wang, A hyperbolic embedding method for weighted networks, IEEE Trans. Netw. Sci. Eng. 8 (1) (2020) 599–612.
[44] Meizi Li, Shuyi Lu, Lele Zhang, Yuping Zhang, Bo Zhang, A community detection method for social network based on community embedding, IEEE Trans. Comput. Soc. Syst. 8 (2) (2021) 308–318.
[45] Yuan Gao, Maoguo Gong, Yu Xie, Hua Zhong, Community-oriented attributed network embedding, Knowl.-Based Syst. 193 (2020) 105418.
[46] Petar Velickovic, William Fedus, William L. Hamilton, Pietro Liò, Yoshua Bengio, R. Devon Hjelm, Deep graph infomax, ICLR (Poster) 2 (3) (2019) 4.
[47] Robert Kleinberg, Geographic routing using hyperbolic space, in: IEEE INFOCOM 2007-26th IEEE International Conference on Computer Communications, IEEE, 2007, pp. 1902–1909.
[48] Benjamin Paul Chamberlain, James Clough, Marc Peter Deisenroth, Neural embeddings of graphs in hyperbolic space, 2017, arXiv preprint arXiv:1705.10359.
[49] Nina Miolane, Susan Holmes, Learning weighted submanifolds with variational autoencoders and riemannian variational autoencoders, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 14503–14511.
[50] Marián Boguñá, Fragkiskos Papadopoulos, Dmitri Krioukov, Sustaining the internet with hyperbolic mapping, Nature Commun. 1 (6) (2010) 62.
[51] Zongning Wu, Zengru Di, Ying Fan, An asymmetric popularity-similarity optimization method for embedding directed networks into hyperbolic space, Complexity 2020 (5) (2020) 1–16.
[52] Dmitri Krioukov, Fragkiskos Papadopoulos, Maksim Kitsak, Amin Vahdat, Marián Bogu ná, Hyperbolic geometry of complex networks, Phys. Rev. E 82 (2) (2010) 036106.
[53] Tobias Friedrich, Anton Krohmer, On the diameter of hyperbolic random graphs, SIAM J. Discrete Math. 32 (2) (2018) 1314–1334.
[54] Michel Bode, Nikolaos Fountoulakis, Tobias Müller, On the giant component of random hyperbolic graphs, in: The Seventh European Conference on Combinatorics, Graph Theory and Applications, 2013, pp. 425–429.
[55] Tobias Friedrich, Anton Krohmer, Cliques in hyperbolic random graphs, in: 2015 IEEE Conference on Computer Communications, INFOCOM, 2015, pp. 1544–1552.
[56] Thomas Bläsius, Tobias Friedrich, Anton Krohmer, Hyperbolic random graphs: Separators and treewidth, in: 24th Annual European Symposium on Algorithms, ESA 2016, 2016.
[57] Palash Goyal, Emilio Ferrara, Graph embedding techniques, applications, and performance: A survey, Knowl.-Based Syst. 151 (2018) 78–94.
[58] Fragkiskos Papadopoulos, Maksim Kitsak, M. Ángeles Serrano, Marián Boguá, Dmitri Krioukov, Popularity versus similarity in growing networks, Nature 489 (7417) (2012) 537–540.
[59] Sepideh Bazzaz Abkenar, Mostafa Haghi Kashani, Ebrahim Mahdipour, Seyed Mahdi Jameii, Big data analytics meets social media: A systematic review of techniques, open issues, and future directions, Telematics Informat. 57 (2021) 101517.
[60] A. Abramian, O. Devauchelle, G. Seizilles, E. Lajeunesse, Boltzmann distribution of sediment transport, Phys. Rev. Lett. 123 (1) (2019) 014501.
[61] Juyong Park, M.E.J. Newman, Statistical mechanics of networks, Phys. Rev. E 70 (6) (2004) 066117.
[62] Ginestra Bianconi, Statistical mechanics of multiplex networks: Entropy and overlap, Phys. Rev. E 87 (6) (2013) 062806.
[63] Giulio Cimini, Tiziano Squartini, Fabio Saracco, Diego Garlaschelli, Andrea Gabrielli, Guido Caldarelli, The statistical physics of real-world networks, Nat. Rev. Phys. 1 (1) (2019) 58–71.
[64] Thomas Hofmann, Jan Puzicha, Statistical Models for Co-Occurrence Data. A.I. Memo 1635, Massachusetts Institute of Technology, 1998.
[65] Art B. Owen, A robust hybrid of lasso and ridge regression, Contemp. Math. 443 (7) (2007) 59–72.

[66] Ross Girshick, Fast r-cnn, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1440–1448.
[67] Meizi Li, Shuyi Lu, Lele Zhang, Yuping Zhang, Bo Zhang, A community detection method for social network based on community embedding, IEEE Trans. Comput. Soc. Syst. (2021) 1–11.
[68] Xiaofei He, Deng Cai, Yuanlong Shao, Hujun Bao, Jiawei Han, Laplacian regularized gaussian mixture model for data clustering, IEEE Trans. Knowl. Data Eng. 23 (9) (2011) 1406–1418.
[69] Muscoloni Alessandro, Cannistraci Carlo Vittorio, A nonuniform popularity-similarity optimization (Npso) model to efficiently generate realistic complex networks with communities, New J. Phys. 20 (5) (2018) 052002.
[70] Haimin Zhang, Min Xu, Graph neural networks with multiple kernel ensemble attention, Knowl.-Based Syst. (2021) 107299.
[71] Bin Zhao, Prithviraj Sen, Lise Getoor, Entity and relationship labeling in affiliation networks, in: ICML Workshop on Statistical Network Analysis, Citeseer, 2006.
[72] Bentian Li, Dechang Pi, Learning deep neural networks for node classification, Expert Syst. Appl. 137 (2019) 324–334.
[73] Lada A. Adamic, Natalie Glance, The political blogosphere and the 2004 us election: divided they blog, in: Proceedings of the 3rd International Workshop on Link Discovery, 2005, pp. 36–43.
[74] Jingwei Zhuo, Ziru Xu, Wei Dai, Han Zhu, Han Li, Jian Xu, Kun Gai, Learning optimal tree models under beam search, in: International Conference on Machine Learning, PMLR, 2020, pp. 11650–11659.
[75] Amanda L. Traud, Peter J. Mucha, Mason A. Porter, Social structure of facebook networks, Physica A 391 (16) (2012) 4165–4180.
[76] Danah Boyd, Viewing american class divisions through facebook and myspace, Apophenia Blog Essay 24 (2007).