Electronic Journal of Statistics
Vol. 16 (2022) 671–699
ISSN: 1935-7524
https://doi.org/10.1214/21-EJS1969

Markov random geometric graph, MRGG: A growth model for temporal dynamic networks*

Quentin Duchemin and Yohann De Castro

LAMA, Univ Gustave Eiffel, Univ Paris Est Créteil, CNRS F-77447 Marne-la-Vallée, France
Institut Camille Jordan, École Centrale de Lyon, Lyon, France
e-mail: quentin.duchemin@univ-eiffel.com; yohann.de-castro@ec-lyon.fr

Abstract: We introduce Markov Random Geometric Graphs (MRGGs), a growth model for temporal dynamic networks. It is based on a Markovian latent space dynamic: consecutive latent points are sampled on the Euclidean Sphere using an unknown Markov kernel; and two nodes are connected with a probability depending on a unknown function of their latent geodesic distance.

More precisely, at each stamp-time k we add a latent point X_k sampled by jumping from the previous one X_{k-1} in a direction chosen uniformly Y_k and with a length r_k drawn from an unknown distribution called the *latitude function*. The connection probabilities between each pair of nodes are equal to the *envelope function* of the distance between these two latent points. We provide theoretical guarantees for the non-parametric estimation of the latitude and the envelope functions.

We propose an efficient algorithm that achieves those non-parametric estimation tasks based on an ad-hoc Hierarchical Agglomerative Clustering approach. As a by product, we show how MRGGs can be used to detect dependence structure in growing graphs and to solve link prediction problems.

MSC2020 subject classifications: Primary 05C80; secondary 62G05, 05C62, 60J05.
Keywords and phrases: Random geometric graph, non-parametric estimation, Markov chains, link prediction, spectral methods.

Received May 2021.

1. Introduction

In Random Geometric Graphs (RGG), nodes are sampled independently in latent space $\mathbb{R}^d$. Two nodes are connected if their distance is smaller than a threshold. A thorough probabilistic study of RGGs can be found in [26]. RGGs have been widely studied recently due to their ability to provide a powerful modeling tool for networks with spatial structure. We can mention applications in bioinformatics [16] or analysis of social media [17]. One main feature is to uncover hidden representation of nodes using latent space and to model interactions by relative positions between latent points.

*This work was supported by a grant from Région Ile-de-France.

Furthermore, nodes interactions may evolve with time. In some applications, this evolution is given by the arrival of new nodes as in online collection growth [22], online social network growth [3, 19], or outbreak modeling [31] for instance. The network is growing as more nodes are entering. Other time evolution modelings have been studied, we refer to [28] for a review.

A natural extension of RGG consists in accounting this time evolution. In [12], the expected length of connectivity and dis-connectivity periods of the Dynamic Random Geometric Graph is studied: each node choose at random an angle in $[0, 2\pi)$ and make a constant step size move in that direction. In [29], a random walk model for RGG on the hypercube is studied where at each time step a vertex is either appended or deleted from the graph. Their model falls into the class of Geometric Markovian Random Graphs that are generally defined in [8].

As far as we know, there is no extension of RGG to growth model for temporal dynamic networks. For the first time, in this paper, we consider a Markovian dynamic on the latent space where the new latent point is drawn with respect to the latest latent point and some Markov kernel to be estimated.

Estimation of graphon in RGGs: the Euclidean sphere case Random graphs with latent space can be defined using a *graphon*, cf. [23]. A graphon is a kernel function that defines edge distribution. In [30], Tang and al. prove that spectral method can recover the matrix formed by graphon evaluated at latent points up to an orthogonal transformation, assuming that graphon is a positive definite kernel (PSD). Going further, algorithms have been designed to estimate graphons, as in [20] which provide sharp rates for the Stochastic Block Model (SBM). Recently, the paper [7] provides a non-parametric algorithm to estimate RGGs on Euclidean spheres, without PSD assumption.

We present here RGG on Euclidean sphere. Given n points $X_1, X_2, \ldots, X_n$ on the Euclidean sphere $\mathbb{S}^{d-1}$, we set an edge between nodes i and j (where $i, j \in [n]$, $i \neq j$) with independent probability $\mathbf{p}(\langle X_i, X_j \rangle)$. The unknown function $\mathbf{p} : [-1, 1] \to [0, 1]$ is called the *envelope function*. This RGG is a graphon model with a symmetric kernel W given by $W(x, y) = \mathbf{p}(\langle x, y \rangle)$. Once the latent points are given, independently draw the random undirected adjacency matrix A by

$$A_{i,j} \sim \mathcal{B}(\mathbf{p}(\langle X_i, X_j \rangle)), \quad i < j$$

with Bernoulli r.v. drawn independently (set zero on the diagonal and complete by symmetry), and set

$$T_n := \frac{1}{n} \left(\mathbf{p}(\langle X_i, X_j \rangle)\right)_{i,j \in [n]} \quad \text{and} \quad \widehat{T}_n := \frac{1}{n} A, \tag{1}$$

We do not observe the latent point and we have to estimate the envelope $\mathbf{p}$ from A only. A standard strategy is to remark that $\widehat{T}_n$ is a random perturbation of T_n and to dig into T_n to uncover $\mathbf{p}$.

One important feature of this model is that the interactions between nodes is depicted by a simple object: the envelope function $\mathbf{p}$. The envelope summarises how individuals connect each others given their latent positions. Standard examples [6] are given by $\mathbf{p}_\tau(t) = \mathbb{1}_{\{t \geq \tau\}}$ where one connects two points as soon

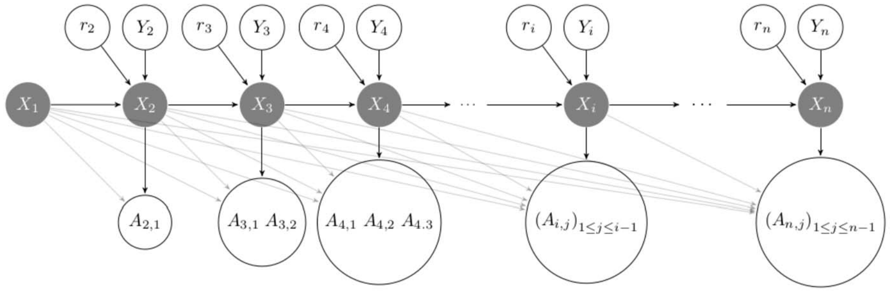


FIG 1. *Graphical model of the MRGG model: Markovian dynamics on Euclidean sphere where we jump from X_k onto X_{k+1}. The Y_k encodes direction of jump while r_k encodes its distance, see* (1).

as their geodesic distance is below some threshold. The non-parametric estimation of $\mathbf{p}$ is given by [7] where the authors assume that latent points X_i are independently and uniformly distributed on the sphere, which will not be the case in the present paper.

A new growth model: the latent Markovian dynamic Consider RGGs where latent points are sampled with Markovian jumps, the Graphical Model under consideration can be found in Figure 1. Namely, we sample n points $X_1, X_2, \ldots, X_n$ on the Euclidean sphere $\mathbb{S}^{d-1}$ using a Markovian dynamic. We start by sampling randomly X_1 on $\mathbb{S}^{d-1}$. Then, for any $i \in \{2, \ldots, n\}$, we sample

- a unit vector $Y_i \in \mathbb{S}^{d-1}$ uniformly, orthogonal to X_{i-1}.
- a real $r_i \in [-1, 1]$ encoding the distance between X_{i-1} and X_i, see (2). r_i is sampled from a distribution $f_{\mathcal{L}} : [-1, 1] \to [0, 1]$, called the *latitude function*.

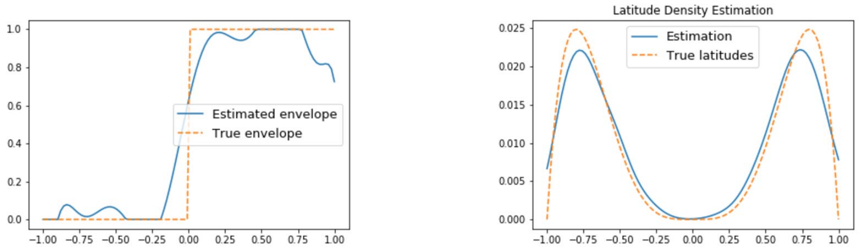
then X_i is defined by

$$X_i = r_i \times X_{i-1} + \sqrt{1 - r_i^2} \times Y_i \,.$$

This dynamic can be pictured as follows. Consider that X_{i-1} is the north pole, then chose uniformly a direction (i.e., a longitude) and, in a independent manner, randomly move along the latitudes (the longitude being fixed by the previous step). The geodesic distance γ_i drawn on the latitudes satisfies

$$\gamma_i = \arccos(r_i) \,, \tag{2}$$

where random variable $r_i = \langle X_i, X_{i-1} \rangle$ has density $f_{\mathcal{L}}(r_i)$. The resulting model will be referred to as the Markov Random Geometric Graph (MRGG) and is described with Figure 1.



(a) Envelope function

(b) Latitude function

FIG 2. *Non-parametric estimation of envelope and latitude functions using algorithms of Sections 2 and 3. We built a graph of* 1500 *nodes sampled on the sphere* $\mathbb{S}^2$ *and using envelope* $\mathbf{p}^{(1)}$ *and latitude* $f_{\mathcal{L}}^{(1)}$ *(dot orange curves) defined in Section 5 by Eq.*(11)*. The estimated envelope is thresholded to get a function in* $[0, 1]$ *and the estimated latitude function is normalized with integral* 1 *(plain blue lines).*

Temporal dynamic networks: MRGG estimation strategy Seldom growth models exist for temporal dynamic network modeling, see [28] for a review. In our model, we add one node at a time making a Markovian jump from the previous latent position. It results in

$$\text{the observation of } (A_{i,j})_{1\leq j\leq i-1} \text{ at time } T = i\,,$$

as pictured in Figure 1. Namely, we observe how a new node connects to the previous ones. For such dynamic, we aim at estimating the model, namely envelope $\mathbf{p}$ and respectively latitude $f_{\mathcal{L}}$. These functions capture in a simple function on $\Omega = [-1, 1]$ the range of interaction of nodes (represented by $\mathbf{p}$) and respectively the dynamic of the jumps in latent space (represented by $f_{\mathcal{L}}$), where, in abscissa Ω, values $r = \langle X_i, X_j\rangle$ near 1 corresponds to close point $X_i \simeq X_j$ while values close to -1 corresponds to antipodal points $X_i \simeq -X_j$. These functions may be non-parametric.

From snapshots of the graph at different time steps, can we recover envelope and latitude functions? This paper proves that it is possible under mild conditions on the Markovian dynamic of the latent points and our approach is summed up with Figure 3.

Define $\lambda(T_n) := (\lambda_1, \ldots, \lambda_n)$ and resp. $\lambda(\widehat{T}_n) := (\hat{\lambda}_1, \ldots, \hat{\lambda}_n)$ the spectrum of T_n and resp. $\widehat{T}_n$, see (1). Building clusters from $\lambda(\widehat{T}_n)$, Algorithm 1 (SCCHEi) estimates the spectrum of envelope $\mathbf{p}$ while Algorithm 3 [1] (HEiC, cf. Section F in Appendix) extracts d eigenvectors of $\widehat{T}_n$ to uncover the Gram matrix of the latent positions. Both can then be used to estimate the unknown functions of our model (cf. Figure 2).

Previous works The latent space approach to model dynamics of network has already been studied in a large span of recent works. Most of them focus on block models with dynamic generalizations covering discrete dynamic

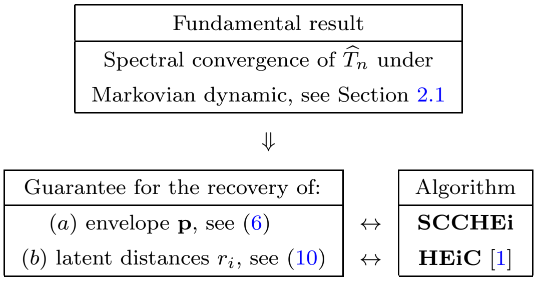

Fundamental result
Spectral convergence of $\widehat{T}_n$ under
Markovian dynamic, see Section 2.1
⇓
Guarantee for the recovery of:
(a) envelope $\mathbf{p}$, see (6)
(b) latent distances r_i, see (10)
↔
↔
Algorithm
SCCHEi
HEiC [1]

FIG 3. *Presentation of our method to recover the envelope and the latitude functions.*

evolution via hidden Markov models (cf. [24]) or continuous time analysis via extended Kalman filter (cf. [32]). [33] and [11] use a Gamma Markov process allowing to model evolving mixed membership in graphs using respectively the Bernoulli Poisson link function and the logistic function to generate edges from the latent space representation. While the above mentioned papers consider community based random graphs with fixed size where edges and communities change through time, we focus on growing RGGs on Euclidean sphere where new nodes are added along time.
Non-parametric estimation of RGGs on $\mathbb{S}^{d-1}$ has been investigated in [7] with i.i.d. latent points. Estimation of latent point relative distances with HEiC Algorithm has been introduced in [1] under i.i.d. latent points assumption. Phase transitions on the detection of geometry in RGGs (against Erdös Rényi alternatives) has been investigated in [6].

For the first time, we introduce latitude function and non-parametric estimations of envelope and latitude using new results on kernel matrices concentration with dependent variables (see Appendix).

Outline Sections 2 and 3 present the estimation method with new theoretical results under Markovian dynamic. These new results are random matrices operator norm control and resp. U-statistics control under Markovian dynamic, presented in the Appendix at Section H and resp. Section G. The envelope *adaptive* estimate is built from a size constrained clustering (Algorithm 1) tuned by slope heuristic Eq. (7), and the latitude function estimate (cf. Section 3.1) is derived from estimates of latent distances r_i. Our method can handle random graphs with logarithmic growth node degree (i.e., new comer at time $T = n$ connects to $\mathcal{O}(\log n)$ previous nodes), referred to as *relatively sparse* models, see Section 4. Sections 5 and 6 investigate synthetic data experiments. We propose heuristics to solve link prediction problems and to test for a Markovian dynamic. In a last section (Section 7), we dig deeper into the analysis of our methods by studying their behaviour under model mispecification or under slow mixing conditions. We conclude by presenting final remarks and future research directions.

Notations Consider a dimension $d \geq 3$. Denote by $\| \cdot \|_2$ (resp. $\langle \cdot, \cdot \rangle$) the Euclidean norm (resp. inner product) on $\mathbb{R}^d$. Consider the d-dimensional sphere $\mathbb{S}^{d-1} := \{x \in \mathbb{R}^d \ : \ \|x\|_2 = 1\}$ and denote by σ the uniform distribution on $\mathbb{S}^{d-1}$. For any matrix $M = (m_{i,j})_{i,j} \in \mathbb{R}^{D_1 \times D_2}$, we define $\|M\|_F^2 := \sum_{i=1}^{D_1} \sum_{j=1}^{D_2} |m_{i,j}|^2$ and the operator norm of M as $\|M\| := \sup_{x \in \mathbb{S}^{D_2-1}} \|Mx\|_2$. For two real valued sequences $(u_n)_{n\in\mathbb{N}}$ and $(v_n)_{n\in\mathbb{N}}$, denote $u_n \underset{n\to\infty}{=} \mathcal{O}(v_n)$ if there exist $k_1 > 0$ and $n_0 \in \mathbb{N}$ such that $\forall n > n_0$, $|u_n| \leq k_1 |v_n|$. For any $x, y \in \mathbb{R}$, $x \wedge y := \min(x, y)$ and $x \vee y := \max(x, y)$. Given two sequences x, y of reals–completing finite sequences by zeros–such that $\sum_i x_i^2 + y_i^2 < \infty$, we define the ℓ_2 rearrangement distance $\delta_2(x, y)$ as

$$\delta_2^2(x, y) := \inf_{\pi \in \mathfrak{S}} \sum_i (x_i - y_{\pi(i)})^2 \,,$$

where $\mathfrak{S}$ is the set of permutations with finite support. This distance is useful to compare two spectra.

2. Nonparametric estimation of the envelope function

One can associate with $W(x, y) = \mathbf{p}(\langle x, y \rangle)$ the integral operator

$$\mathbb{T}_W : L^2(\mathbb{S}^{d-1}) \to L^2(\mathbb{S}^{d-1}),$$

such that for any $g \in L^2(\mathbb{S}^{d-1})$,

$$\forall x \in \mathbb{S}^{d-1}, \quad (\mathbb{T}_W g)(x) = \int_{\mathbb{S}^{d-1}} g(y) \mathbf{p}(\langle x, y \rangle) \sigma(dy),$$

where σ is the Lebesgue measure on $\mathbb{S}^{d-1}$. The operator $\mathbb{T}_W$ is Hilbert-Schmidt and it has a countable number of bounded eigenvalues λ_k^* with zero as only accumulation point. The eigenfunctions of $\mathbb{T}_W$ have the remarkable property that they do not depend on $\mathbf{p}$ (cf. [9] Lemma 1.2.3): they are given by the real Spherical Harmonics. We denote $\mathcal{H}_l$ the space of real Spherical Harmonics of degree l with dimension d_l and with orthonormal basis $(Y_{l,j})_{j \in [d_l]}$ where

$$d_l := \dim(\mathcal{H}_l) = \begin{cases} 1 & \text{if } l = 0 \\ d & \text{if } l = 1 \\ \binom{l+d-1}{l} - \binom{l+d-3}{l-2} & \text{otherwise.} \end{cases}$$

We define also for all $R \in \mathbb{N}$, $\widetilde{R} := \sum_{l=0}^{R} d_l$. We end up with the following spectral decomposition

$$\mathbf{p}(\langle x, y \rangle) = \sum_{l \geq 0} p_l^* \sum_{1 \leq j \leq d_l} Y_{l,j}(x) Y_{l,j}(y) = \sum_{k \geq 0} p_k^* c_k G_k^{\beta}(\langle x, y \rangle) \,, \tag{3}$$

where $\lambda(\mathbb{T}_W) = \{p_0^*, p_1^*, \dots, p_1^*, \dots, p_l^*, \dots, p_l^*, \dots\}$ meaning that each eigenvalue p_l^* has multiplicity d_l; and G_k^{β} is the Gegenbauer polynomial of degree k

with parameter $\beta := \frac{d-2}{2}$ and $c_k := \frac{2k+d-2}{d-2}$ (cf. Appendix C). Since $\mathbf{p}$ is bounded, one has $\mathbf{p} \in L^2((-1,1), w_\beta)$ where the weight function w_β is defined by $w_\beta(t) := (1-t^2)^{\beta - \frac{1}{2}}$ and

$$L^2((-1,1), w_\beta) := \big\{ g : [-1,1] \to \mathbb{R} \;\big|\; \|g\|_2^2 := \int_{-1}^{1} |g(t)|^2 w_\beta(t) dt < +\infty \big\}.$$

Note that $\mathbf{p}$ can be decomposed as $\mathbf{p} \equiv \sum_{k \geq 0} p_k^* c_k G_k^\beta$ and the Gegenbauer polynomials G_k^β are an orthogonal basis of $L^2((-1,1), w_\beta)$.

We finally introduce for any resolution level $R \in \mathbb{N}$ the truncated graphon W_R which is obtained from W by keeping only the $\widetilde{R}$ first eigenvalues, that is

$$\forall x, y \in \mathbb{S}^{d-1}, \quad W_R(x,y) := \sum_{k=0}^{R} p_k^* \sum_{l=1}^{d_k} Y_{k,l}(x) Y_{k,l}(y).$$

Similarly, we denote for all $t \in [0,1]$, $\mathbf{p}_R(t) = \sum_{k=0}^{R} p_k^* c_k G_k^\beta(t)$.

Weighted Sobolev space The space $Z^s_{w_\beta}((-1,1))$ with regularity $s > 0$ is defined as the set of functions $g = \sum_{k \geq 0} g_k^* c_k G_k^\beta \in L^2((-1,1), w_\beta)$ such that

$$\|g\|^*_{Z^s_{w_\beta}((-1,1))} := \left[\sum_{l=0}^{\infty} d_l |g_l^*|^2 \left(1 + (l(l+2\beta))^s\right) \right]^{1/2} < \infty.$$

2.1. Integral operator spectrum estimation with dependent variables

One key result is a new control of U-statistics with latent Markov variables (cf. Section G) and it makes use of a Talagrand's concentration inequality for Markov chains. This article follows the hypotheses made on the Markov chain $(X_i)_{i \geq 1}$ by [10]. Namely, we work under the following assumption.

Assumption A The latitude function $f_{\mathcal{L}}$ is such that $\|f_{\mathcal{L}}\|_\infty < \infty$ and makes the chain $(X_i)_{i \geq 1}$ uniformly ergodic.

Theorem 1 is a theoretical guarantee for a random matrix approximation of the spectrum of integral operator with **dependent** latent variables. Theorem 5 in Appendix H gives explicitly the constants hidden in the big O below which depend on the spectral gap of the Markov chain $(X_i)_{i \geq 1}$.

Theorem 1. *We consider that Assumption A holds and we assume the envelope* $\mathbf{p}$ *has regularity* $s > 0$*. Then, it holds*

$$\mathbb{E}\left[\delta_2^2(\lambda(\mathbb{T}_W), \lambda(T_n)) \vee \delta_2^2(\lambda(\mathbb{T}_W), \lambda^{R_{opt}}(\widehat{T}_n)) \right] = \quad \mathcal{O}\left(\left[\frac{n}{\log^2(n)} \right]^{-\frac{2s}{2s+d-1}} \right),$$

with $\lambda^{R_{opt}}(\widehat{T}_n) = (\hat{\lambda}_1, \dots, \hat{\lambda}_{\widetilde{R}_{opt}}, 0, 0, \dots)$ *and* $R_{opt} = \lfloor \left(n / \log^2(n)\right)^{\frac{1}{2s+d-1}} \rfloor$*.* $\hat{\lambda}_1, \dots, \hat{\lambda}_n$ *are the eigenvalues of* $\widehat{T}_n$ *sorted in decreasing order of magnitude.*

Remark In Theorem 1 and Theorem 4, note that we recover, up to a log factor, the *minimax rate of non-parametric estimation* of s-regular functions on a space of (Riemannian) dimension $d-1$. Even with i.i.d. latent variables, it is still an open question to know if this rate is the minimax rate of non-parametric estimation of RGGs.

Eq. (3) shows that one could use an approximation of $(p_k^*)_{k\geq 1}$ to estimate the envelope $\mathbf{p}$ and Theorem 1 states we can recover $(p_k^*)_{k\geq 1}$ up to a permutation. In most cases, the problem of finding such a permutation is NP-hard and we introduce in the next section an efficient algorithm to fix this issue.

2.2. Size constrained clustering algorithm

Note the spectrum of $\mathbb{T}_W$ is given by $(p_l^*)_{l\geq 0}$ where p_l^* has multiplicity d_l. In order to recover envelope $\mathbf{p}$, we build clusters from eigenvalues of $\widehat{T}_n$ while respecting the dimension d_l of each eigen-space of $\mathbb{T}_W$. In [7], an algorithm is proposed testing all permutations of $\{0,\dots,R\}$ for a given maximal resolution R. To bypass the high computational cost of such approach, we propose an efficient method based on the tree built from *Hierarchical Agglomerative Clustering* (HAC). In the following, for any $\nu_1,\dots,\nu_n\in\mathbb{R}$, we denote by $\mathrm{HAC}(\{\nu_1,,\dots,\nu_n\},d_c)$ the tree built by a HAC on the real values $\nu_1,\dots,\nu_n$ using the complete linkage function d_c defined by $\forall A,B\subset\mathbb{R}$, $d_c(A,B)=\max_{a\in A}\max_{b\in B}\|a-b\|_2$. Algorithm 1 describes our approach.

Algorithm 1 Size Constrained Clustering for Harmonic Eigenvalues (SCCHEi).

```
Data: Resolution $R$, matrix $\widehat{T}_n = \frac{1}{n}A$, dimensions $(d_k)_{k=0}^{R}$.
 1: Let $\hat{\lambda}_1,\dots,\hat{\lambda}_n$ be the eigenvalues of $\widehat{T}_n$ sorted in decreasing order of magnitude.
 2: Set $\mathcal{P} := \{\hat{\lambda}_1,\dots,\hat{\lambda}_{\tilde{R}}\}$ and $dims = [d_0,d_1,\dots,d_R]$.
 3: while All eigenvalues in $\mathcal{P}$ are not clustered do
 4:     $tree \leftarrow$ HAC(nonclustered eigenvalues in $\mathcal{P}$, $d_c$)
 5:     for $d \in dims$ do
 6:         Search for a cluster of size $d$ in $tree$ as close as possible to the root.
 7:         if such a cluster $\mathcal{C}_d$ exists then Update$(dims, tree, \mathcal{C}_d, d)$.
 8:     end for
 9:     for $d \in dims$ do
10:         Search for the group $\mathcal{C}$ in $tree$ with a size larger than $d$ and as close as possible
            to $d$.
11:         if such a group exists then Update$(dims, tree, \mathcal{C}, d)$ else Go to line 3.
12:     end for
13: end while
Return: $\mathcal{C}_{d_0},\dots,\mathcal{C}_{d_R},\{\hat{\lambda}_{\tilde{R}+1},\dots,\hat{\lambda}_n\}$
```

Algorithm 2 Update$(dims, tree, \mathcal{C}, d)$.

```
1: Save the subset $\mathcal{C}_d$ consisting of the $d$ eigenvalues in $\mathcal{C}$ with the largest absolute values.
2: Delete from $tree$ all occurrences to eigenvalues in $\mathcal{C}_d$ and delete $d$ from $dims$.
```

Given some resolution level $R \in \mathbb{N}$, our estimator $\widehat{\mathbf{p}}_R$ of the envelope function $\mathbf{p}$ is obtained from the clustering of the eigenvalues obtained by the SCCHEi algorithm as follows

$$\widehat{\mathbf{p}}_R : t \mapsto \sum_{k=0}^{R} \widehat{p}_k c_k G_k^{\beta}(t) \quad \text{where} \quad \forall k \in \{0, \dots, R\}, \quad \widehat{p}_k := \frac{1}{d_k} \sum_{\lambda \in \mathcal{C}_{d_k}} \lambda. \tag{4}$$

2.3. Theoretical guarantees

Let us recall that for any resolution level $R \geq 0$,

$$\lambda(\mathbb{T}_{W_R}) = (\lambda_1^*, \dots, \lambda_{\widetilde{R}}^*, 0, 0, \dots) \text{ and } \lambda^R(\widehat{T}_n) = (\hat{\lambda}_1, \dots, \hat{\lambda}_{\widetilde{R}}, 0, 0, \dots)$$

where $\hat{\lambda}_1, \dots, \hat{\lambda}_n$ are the eigenvalues of $\widehat{T}_n$ sorted in decreasing order of magnitude. We order the eigenvalues $\hat{\lambda}_1, \dots, \hat{\lambda}_{\widetilde{R}}$ and in the following we consider that $\lambda^R(\widehat{T}_n)_1 \geq \cdots \geq \lambda^R(\widehat{T}_n)_{\widetilde{R}}$.

Theorem 2. *Let us consider some resolution level $R \in \mathbb{N}$. We keep the assumptions of Theorem 1. We recall that we consider $\lambda^R(\widehat{T}_n)_1 \geq \cdots \geq \lambda^R(\widehat{T}_n)_{\widetilde{R}}$.*

Then for n large enough, the clusters $\mathcal{C}_{d_0}, \dots, \mathcal{C}_{d_R}$ obtained from the SCCHEi algorithm satisfy

$$\delta_2^2(\lambda(\mathbb{T}_{W_R}), \lambda^R(\widehat{T}_n)) = \sum_{k=0}^{R} \sum_{\hat{\lambda} \in \mathcal{C}_{d_k}} (\hat{\lambda} - p_k^*)^2.$$

Proof of Theorem 2

Let us denote

$$\Delta^G = \min_{0 \leq k \neq l \leq R,\ p_k^* \neq p_l^*} |p_k^* - p_l^*| \wedge \min_{0 \leq k \leq R,\ p_k^* \neq 0} |p_k^*| > 0.$$

For any $g \in (0, \frac{\Delta^G}{4})$, the proof of Theorem 1 (cf. Appendix H) ensures that for n large enough it holds

$$\delta_2^2(\lambda(\mathbb{T}_{W_R}), \lambda^R(\widehat{T}_n)) \leq g^2. \tag{5}$$

Let us recall that

$$\delta_2^2(\lambda(\mathbb{T}_{W_R}), \lambda^R(\widehat{T}_n)) = \inf_{\pi \in \mathfrak{S}} \sum_{i \geq 1} \Big(\lambda(\mathbb{T}_{W_R})_{\pi(i)} - \lambda^R(\widehat{T}_n)_i \Big)^2.$$

The proof of Theorem 2 relies on the following two Lemmas. The proofs of these Lemmas are postponed to Section D.

Lemma 1. *We keep the assumptions of Theorem 2. Then, for n large enough for Eq. (5) to hold, one can choose a permutation π^* such that*

- $\pi^*(\{1,\dots,\widetilde{R}\}) = \{1,\dots,\widetilde{R}\}$.
- $\delta_2^2(\lambda(\mathbb{T}_{W_R}),\lambda^R(\widehat{T}_n)) = \sum_{i=1}^{\widetilde{R}}(\lambda(\mathbb{T}_{W_R})_{\pi^*(i)} - \lambda^R(\widehat{T}_n)_i)^2$.

Moreover, the function f^* given by

$$\begin{aligned} f^* : \{1,\dots,\widetilde{R}\} &\to \{p_k^*,\ 0 \le k \le R\} \\ i &\mapsto \lambda(\mathbb{T}_{W_R})_{\pi^*(i)}, \end{aligned}$$

is non-increasing.

Lemma 2. *We keep the assumptions and notations of Lemma 1. A clustering $\left(\widehat{\mathcal{C}}_{d_k}\right)_{0\le k\le R}$ at depth R in the tree of the HAC algorithm applied to $\mathcal{P} := \{\lambda^R(\widehat{T}_n)_1,\dots,\lambda^R(\widehat{T}_n)_{\widetilde{R}}\}$ is said to be of type $(\mathcal{S})$ if it satisfies:*

$$\begin{aligned} \widehat{\mathcal{C}}_{d_0} &\subset \{\lambda^R(\widehat{T}_n)_i \mid 1 \le i \le \widetilde{R},\ f^*(i) = p_0^*\}, \quad \#\widehat{\mathcal{C}}_{d_0} = d_0, \\ \widehat{\mathcal{C}}_{d_1} &\subset \{\lambda^R(\widehat{T}_n)_i \mid 1 \le i \le \widetilde{R},\ f^*(i) = p_1^*\}, \quad \#\widehat{\mathcal{C}}_{d_1} = d_1, \\ &\dots \\ \widehat{\mathcal{C}}_{d_R} &\subset \{\lambda^R(\widehat{T}_n)_i \mid 1 \le i \le \widetilde{R},\ f^*(i) = p_R^*\}, \quad \#\widehat{\mathcal{C}}_{d_R} = d_R. \end{aligned}$$

Then the HAC algorithm with complete linkage applied to $\mathcal{P}$ reaches (after $\widetilde{R} - R - 1$ iterations) a state $\left(\widehat{\mathcal{C}}_{d_k}\right)_{0\le k\le R}$ of type $(\mathcal{S})$. As a consequence, the SCCHEi algorithm returns the clusters $\mathcal{C}_{d_0} = \widehat{\mathcal{C}}_{d_0},\dots,\mathcal{C}_{d_R} = \widehat{\mathcal{C}}_{d_R}$.

Theorem 2 directly follows from the conclusion of Lemma 2 since we get that

$$\begin{aligned} \sum_{k=0}^{R}\sum_{\hat{\lambda}\in\mathcal{C}_{d_k}}(\hat{\lambda} - p_k^*)^2 &= \sum_{i=1}^{\widetilde{R}}(\lambda^R(\widehat{T}_n)_i - f^*(i))^2 = \sum_{i=1}^{\widetilde{R}}(\lambda^R(\widehat{T}_n)_i - \lambda(\mathbb{T}_{W_R})_{\pi^*(i)})^2 \\ &= \delta_2^2(\lambda(\mathbb{T}_{W_R}),\lambda^R(\widehat{T}_n)), \end{aligned}$$

where the first equality comes from the conclusion of Lemma 2, the second one comes from the definition of f^* from Lemma 1 and the last one comes from the choice of π^* from Lemma 1.

■

Theorem 2 ensures that under appropriate conditions, the SCCHEi leads to a clustering of the eigenvalues of the adjacency matrix that achieves the δ_2 distance between $\lambda(\mathbb{T}_{W_R})$ and $\lambda^R(\widehat{T}_n)$. Nevertheless, this is not a sufficient condition to ensure that the L^2 error between the true envelope function and our plug-in estimator (cf. Eq. (4)) goes to 0 has $n \to +\infty$. This is due to identifiability issues coming from the δ_2 metric. This was already mentioned in [7, Section

3.6], where the authors present the following example. Consider the case $d = 3$, which implies $\beta = 1/2$, $d_k = 2k + 1$, $c_k = 2k + 1$. For $\mu > 0$, let

$$\mathbf{p}_a = \frac{1}{2}c_0G_0^\beta + \mu c_1G_1^\beta + 0 \times c_2G_2^\beta + 0 \times c_3G_3^\beta + \mu c_4G_4^\beta$$
$$\mathbf{p}_b = \frac{1}{2}c_0G_0^\beta + 0 \times c_1G_1^\beta + \mu c_2G_2^\beta + \mu c_3G_3^\beta + 0 \times c_4G_4^\beta$$

Then the associated spectrum are

$$\lambda_a^* = (1/2, \underbrace{\mu, \mu, \mu}_{3}, \underbrace{0, 0, 0, 0, 0}_{5}, \underbrace{0, 0, 0, 0, 0, 0, 0}_{7}, \underbrace{\mu, \mu, \mu, \mu, \mu, \mu, \mu, \mu, \mu}_{9})$$
$$\lambda_b^* = (1/2, \underbrace{0, 0, 0}_{3}, \underbrace{\mu, \mu, \mu, \mu, \mu}_{5}, \underbrace{\mu, \mu, \mu, \mu, \mu, \mu, \mu}_{7}, \underbrace{0, 0, 0, 0, 0, 0, 0, 0, 0}_{9})$$

which are indistinguishable in δ_2 metric, although $\|\mathbf{p}_a - \mathbf{p}_b\|_2 = \mu\sqrt{24}$.

Nevertheless, we can obtain a theoretical guarantee on the L^2 error between the true envelope function and our plug-in estimate using Theorem 2 if we consider additional conditions on the eigenvalues $(p_k^*)_{k\geq 0}$.

Theorem 3. *Assume that the envelope function* $\mathbf{p}$ *is polynomial of degree* $D \in \mathbb{N}$*, i.e.,* $p_k^* = 0$ *for any* $k > D$ *and* $p_D^* \neq 0$*. Assume also that all nonzeros* p_k^* *for* $k \in \{0, \dots, D\}$ *are distinct and that* $R \geq D$*. Then for* n *large enough it holds with probability at least* $1 - n^{-8}$*,*

$$\|\widehat{\mathbf{p}}_R - \mathbf{p}\|_2^2 \leq c\frac{\widetilde{R}}{n}\ln(n),$$

where $c > 0$ *is a universal numerical constant.*

Remarks

- The question of whether the problem of estimating $\mathbf{p}$ is NP-hard was still completely open. Theorem 3 brings a first partial answer to this question by showing that $\mathbf{p}$ can be estimated in polynomial time in the case where $\mathbf{p}$ is a polynomial with all non-zero eigenvalues distinct.
- The proof of Theorem 3 is strictly analogous to the one of [7, Proposition 9]. In a nutshell, considering that the envelope function $\mathbf{p}$ is a polynomial with all non-zeros eigenvalues p_k^* distinct ensures that (since $R \geq D$)

$$\delta_2^2(\lambda(\mathbb{T}_{W_R}), \lambda^R(\widehat{T}_n)) = \delta_2^2(\lambda(\mathbb{T}_W), \lambda^R(\widehat{T}_n)),$$

which coincides with the L^2 norm of the difference between $\mathbf{p}$ and its estimate

$$\widehat{\mathbf{p}}_{opt,R} := \sum_{k=0}^{R} \widehat{p}_{opt,k}c_kG_k^\beta \quad \text{with} \quad \widehat{p}_{opt,k} := \frac{1}{d_k}\sum_{i\in(\pi^*)^{-1}([\widetilde{k}+1, \widetilde{k+1}])} \lambda^R(\widehat{T}_n)_i,$$

where π^* is a permutation as defined in Lemma 1. Since we proved that for n large enough, the clusters returned by the SCCHEi algorithm correspond to an allocation given by f^*, we deduce that the L^2 norm between $\mathbf{p}$ and our plug-in estimate $\widehat{\mathbf{p}}_R$ is equal to the δ_2 distance between spectra. The result then comes directly using Theorem 1.

2.4. Adaptation: slope heuristic as model selection of Resolution

A data-driven choice of model size R can be done by *slope heuristic*, see [2] for a nice review. One main idea of slope heuristic is to penalize the empirical risk by $\kappa\,\mathrm{pen}(\widetilde{R})$ and to calibrate $\kappa > 0$. If the sequence $(\mathrm{pen}(\widetilde{R}))_{\widetilde{R}}$ is equivalent to the sequence of variances of the population risk of empirical risk minimizer (ERM) as model size $\widetilde{R}$ grows, then, penalizing the empirical risk (as done in Eq. (7)), one may ultimately uncover an empirical version of the U-shaped curve of the population risk. Hence, minimizing it, one builds a model size $\hat{R}$ that balances between bias (*under-fitting* regime) and variance (*over-fitting* regime). First, note that empirical risk is given by the intra-class variance below.

Definition 1. *For any output $(\mathcal{C}_{d_0}, \dots, \mathcal{C}_{d_R}, \Lambda)$ of the Algorithm SCCHEi, the thresholded intra-class variance is defined by*

$$\mathcal{I}_R := \frac{1}{n}\left[\sum_{k=0}^{R}\sum_{\lambda\in\mathcal{C}_{d_k}}\left(\lambda - \frac{1}{d_k}\sum_{\lambda'\in\mathcal{C}_{d_k}}\lambda'\right)^2 + \sum_{\lambda\in\Lambda}\lambda^2\right],$$

and the estimations $(\hat{p}_k)_{k\ge 0}$ of the eigenvalues $(p^*_k)_{k\ge 0}$ is given by

$$\forall k \in \mathbb{N}, \quad \hat{p}_k = \begin{cases} \frac{1}{d_k}\sum_{\lambda\in\mathcal{C}_{d_k}}\lambda & \textit{if } k \in \{0,\dots,\hat{R}\} \\ 0 & \textit{otherwise.}\end{cases} \tag{6}$$

Second, as underlined in the proof of Theorem 1 (see Theorem 5 in the Appendix), the estimator's variance of our estimator scales linearly in $\widetilde{R}$.

Hence, we apply Algorithm SCCHEi for R varying from 0 to $R_{\max}$ (with $R_{\max} := \max\{R \ge 0 \,:\, \widetilde{R} \le n\}$) to compute the *thresholded intra-class variance* $\mathcal{I}_R$ (see Definition 1) and given some $\kappa > 0$, we select

$$R(\kappa) \in \operatorname*{arg\,min}_{R\in\{0,\dots,R_{\max}\}}\left\{\mathcal{I}_R + \kappa\frac{\widetilde{R}}{n}\right\}. \tag{7}$$

The hyper-parameter κ controlling the bias-variance trade-off is set to $2\kappa_0$ where κ_0 is the value of $\kappa > 0$ leading to the "*largest jump*" of the function $\kappa \mapsto R(\kappa)$. Once $\widehat{R} := R(2\kappa_0)$ has been computed, we approximate the envelope function $\mathbf{p}$ using Eq. (6) (see Eq. (20) in Appendix for the closed form). We denote this estimator $\widehat{\mathbf{p}}$ and with the notations of Eq. (4) it holds $\widehat{\mathbf{p}} = \widehat{\mathbf{p}}_{\widehat{R}}$. In Appendix E, we describe this slope heuristic on a concrete example and our results can be reproduced using the notebook *Experiments*[1] in the Supplementary Material.

[1] https://github.com/quentin-duchemin/Markovian-random-geometric-graph

3. Nonparametric estimation of the latitude function

3.1. Our approach to estimate the latitude function in a nutshell

In Theorem 4 (see below), we show that we are able to estimate consistently the pairwise distances encoded by the Gram matrix G^* where

$$G^* := \frac{1}{n}\left(\langle X_i, X_j\rangle\right)_{i,j\in[n]}.$$

Taking the diagonal just above the main diagonal (referred to as *superdiagonal*) of $\widehat{G}$ - an estimate of the matrix G to be specified - we get estimates of the i.i.d. random variables $(\langle X_i, X_{i-1}\rangle)_{2\le i\le n} = (r_i)_{2\le i\le n}$ sampled from $f_{\mathcal{L}}$. Using $(\hat{r}_i)_{2\le i\le n}$ the superdiagonal of $n\widehat{G}$, we can build a kernel density estimator of the latitude function $f_{\mathcal{L}}$. In the following, we describe the algorithm used to build our estimator $\hat{G}$ with theoretical guarantees.

3.2. Spectral gap condition and Gram matrix estimation

The Gegenbauer polynomial of degree one is defined by $G_1^\beta(t) = 2\beta t,\ \forall t \in [-1,1]$. As a consequence, using the *addition theorem* (cf. [9, Lem.1.2.3 and Thm.1.2.6]), the Gram matrix G^* is related to the Gegenbauer polynomial of degree one. More precisely, for any $i, j \in [n]$ it holds

$$G^*_{i,j} = \frac{1}{2\beta n} G_1^\beta(\langle X_i, X_j\rangle) = \frac{1}{nd}\sum_{k=1}^{d} Y_{1,k}(X_i)Y_{1,k}(X_j). \tag{8}$$

Denoting for all $k \in [d]$ $v_k^* := \frac{1}{\sqrt{n}}\left(Y_{1,k}(X_1), \dots, Y_{1,k}(X_n)\right) \in \mathbb{R}^n$, and $V^* = (v_1^*, \dots, v_d^*) \in \mathbb{R}^{n\times d}$, Eq. (8) becomes

$$G^* := \frac{1}{d} V^*(V^*)^\top.$$

We will prove that for n large enough there exists a matrix $\widehat{V} \in \mathbb{R}^{n\times d}$ where each column is an eigenvector of $\widehat{T}_n$, such that $\widehat{G} := \frac{1}{d}\widehat{V}\widehat{V}^\top$ approximates G^* well, in the sense that the Frobenius norm $\|G^* - \widehat{G}\|_F$ converges to 0. To choose the d eigenvectors of the matrix $\widehat{T}_n$ that we will use to build the matrix $\widehat{V}$, we need the following spectral gap condition

$$\Delta^* := \min_{k\in\mathbb{N},\ k\neq 1} |p_1^* - p_k^*| > 0. \tag{9}$$

This condition will allow us to apply Davis-Kahan type inequalities.

Now, thanks to Theorem 1, we know that the spectrum of the matrix $\widehat{T}_n$ converges towards the spectrum of the integral operator $\mathbb{T}_W$. Then, based on Eq. (8), one can naturally think that extracting the d eigenvectors of the matrix $\widehat{T}_n$ related with the eigenvalues that converge towards p_1^*, we can approximate the Gram matrix G^* of the latent positions. Theorem 4 proves that the latter intuition is true with high probability under the spectral gap condition (9). The algorithm HEiC [1] (cf. Section F for a presentation) aims at identifying the above mentioned d eigenvectors of the matrix $\widehat{T}_n$ to build our estimate of the Gram matrix G^*.

Theorem 4. *We consider that Assumption A holds, we assume $\Delta^* > 0$, and we assume that graphon W has regularity $s > 0$. We denote $\widehat{V} \in \mathbb{R}^{n\times d}$ the d eigenvectors of the matrix $\widehat{T}_n$ associated with the eigenvalues returned by the algorithm HEiC and we define $\widehat{G} := \frac{1}{d}\widehat{V}\widehat{V}^\top$. Then for n large enough and for some constant $D > 0$, it holds with probability at least $1 - 5/n^2$,*

$$\|G^* - \widehat{G}\|_F \leq D\left(\frac{n}{\log^2(n)}\right)^{\frac{-s}{2s+d-1}}. \tag{10}$$

Based on Theorem 4, we propose a kernel density approach to estimate the latitude function $f_{\mathcal{L}}$ based on the super-diagonal of the matrix $\widehat{G}$, namely $\left(\hat{r}_i := n\widehat{G}_{i-1,i}\right)_{i\in\{2,\dots,n\}}$. In the following, we denote $\hat{f}_{\mathcal{L}}$ this estimator.

4. Relatively sparse regime

Although this paper deals with the so-called *dense* regime (i.e. when the expected number of neighbors of each node scales linearly with n), our results may be generalized to the *relatively sparse* model connecting nodes i and j with probability $W(X_i, X_j) = \zeta_n \mathbf{p}(\langle X_i, X_j\rangle)$ where $\zeta_n \in (0,1]$ satisfies $\liminf_n \zeta_n n/\log n \geq Z$ for some universal constant $Z > 0$.

In the relatively sparse model, one can show following the proof of Theorem 1 that the resolution should be chosen as $\widehat{R} = \left(\frac{n\zeta_n}{1+\zeta_n \log^2 n}\right)^{\frac{1}{2s+d-1}}$. Specifying that $\lambda^* = (p_0^*, p_1^*, \dots, p_1^*, p_2^*, \dots)$ and $\widehat{T}_n = A/n$, Theorem 1 becomes for a graphon with regularity $s > 0$

$$\mathbb{E}\left[\delta_2^2\left(\lambda^*, \frac{\lambda(\widehat{T}_n)}{\zeta_n}\right)\right] = \mathcal{O}\left(\left(\frac{n\zeta_n}{1+\zeta_n\log^2 n}\right)^{\frac{-2s}{2s+d-1}}\right).$$

Figure 4 illustrates the estimation of the latitude and the envelope functions in some relatively sparse regimes.

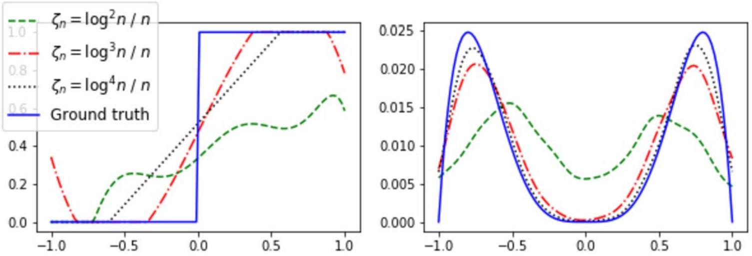


FIG 4. *Results of our algorithms for graph of size* 2000 *with functions* $\mathbf{p}^{(1)}$ *and* $f_{\mathcal{L}}^{(1)}$ *of Eq.*(11) *and sparsity parameter* $\zeta_n = \log^k n/n,\ k \in \{2,3,4\}$.

5. Experiments

In the following, we test our methods using different envelope and latitude functions. Note that a common choice of connection functions in RGGs are the *Rayleigh fading* activation functions which take the form

$$\mathcal{R}_{\zeta,\eta,r}(\rho) = \exp\left[-\zeta\rho^{\eta}\right], \quad \zeta > 0, \eta > 0.$$

Any Rayleigh function $\mathcal{R}_{\zeta,\eta}$ corresponds to the following envelope function

$$\mathbf{p}_{\zeta,\eta} : t \mapsto \mathcal{R}_{\zeta,\eta}(2(1-t)),$$

so that it holds

$$\forall x, y \in \mathbb{S}^{d-1}, \quad \mathbf{p}_{\zeta,\eta}(\langle x, y\rangle) = \mathcal{R}_{\zeta,\eta}(\|x-y\|_2).$$

Let us also denote for any $\alpha, \beta > 0$ $g(\cdot;\alpha,\beta)$ the density of the beta distribution $\mathcal{B}(\alpha,\beta)$ with parameters (α,β). In this paper, we will study the numerical results of our methods considering the following envelope and latitude functions

$$\mathbf{p}^{(1)} : x \mapsto \mathbb{1}_{x\geq 0}, \qquad\qquad \mathbf{p}^{(2)} \equiv \mathbf{p}_{0.5,1}$$

$$f_{\mathcal{L}}^{(1)} : r \mapsto \begin{cases} \frac{1}{2}g(1-r;2,2) & \text{if } r \geq 0 \\ \frac{1}{2}g(1+r;2,2) & \text{otherwise} \end{cases}, \qquad f_{\mathcal{L}}^{(2)} : r \mapsto \frac{1}{2}g\left(\frac{1-r}{2};1,3\right)$$

$$\text{and} \quad \mathbf{p}^{(3)} \equiv \mathbf{p}_{0.25,3}$$

$$f_{\mathcal{L}}^{(3)} : r \mapsto \frac{1}{2}g\left(\frac{1-r}{2};2,2\right). \tag{11}$$

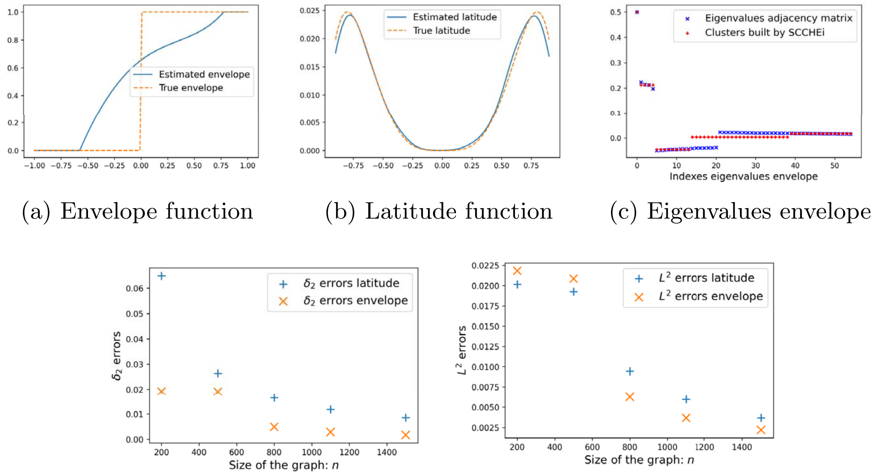
Note that considering the latitude function $f_{\mathcal{L}}^{(2)}$ (resp. $f_{\mathcal{L}}^{(3)}$) is equivalent to consider that one fourth of the Euclidean distance between consecutive latent positions is distributed as $Z \sim \mathcal{B}(1,3)$ (resp. $Z \sim \mathcal{B}(2,2)$). With Figures 5, 6 and 7, we present the results of our experiments for the three different settings described in Eq. (11). In each case, we work with a latent dimension $d = 4$ and we show:

1. the estimates of the envelope and latitude functions obtained with our adaptive procedure working the graph of 1500 nodes (see Figures (a) and (b)).

2. the corresponding clustering obtained by the SCCHEi algorithm for the resolution level R determined by the slope heuristic (see Figures (c)).

   Blue crosses represent the $\widetilde{R}$ eigenvalues of $\widehat{T}_n$ with the largest magnitude, which are used to form clusters corresponding to the $R+1$-first spherical harmonic spaces. The red plus are the estimated eigenvalues $(\hat{p}_k)_{0\le k\le R}$ (plotted with multiplicity) defined from the clustering given by our algorithm SCCHEi (see Eq. (6)). Those results show that SCCHEi achieves a relevant clustering of the eigenvalues of $\widehat{T}_n$ which allows us to recover the envelope function.

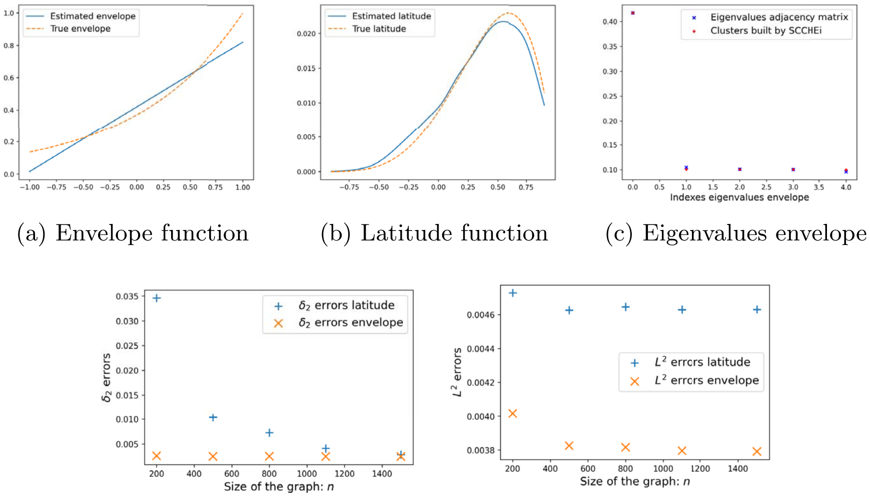
3. the errors between the estimated functions and the true ones in δ_2 metric and in L^2 norm for different size of graphs (see Figures (d) and (e)). We notice that a significant decrease of the δ_2 distance between spectra does not necessarily means that the L^2 norm between the estimated and the true envelope functions shrinks seriously. We refer in particular to Figures 5 and 7. The identifiability issue highlighted in Section 2.3 is one of the possible explanations of this phenomenon. Nevertheless, these experiments show that both the δ_2 and L^2 errors on our estimate of the envelope or the latitude functions are decreasing as the size of the graph is getting larger. Let us also recall that Theorem 3 ensures that the L^2 error on our estimate of the envelope function goes to zero as n grows when $\mathbf{p}$ has a finite number of non zeros eigenvalues that are all distinct.



(a) Envelope function (b) Latitude function (c) Eigenvalues envelope

(d) δ_2 errors (e) L^2 errors

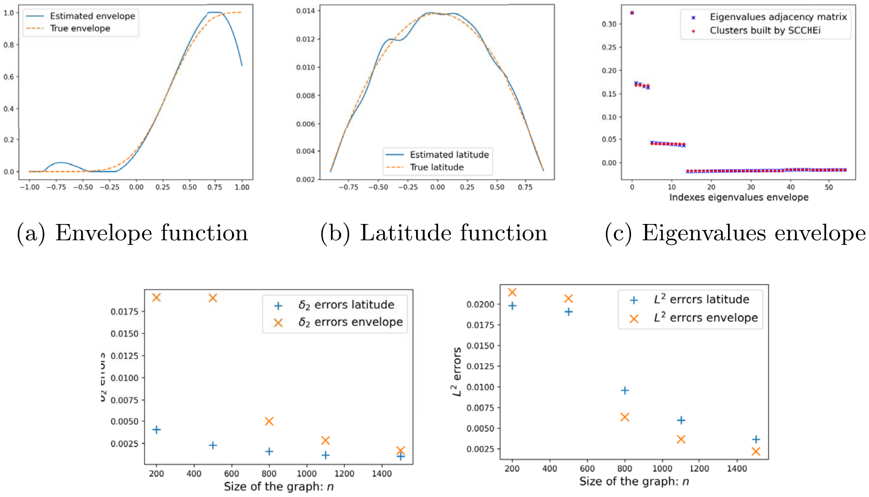
FIG 5. *Results for $d = 4$, the envelope $\mathbf{p}^{(1)}$ and the latitude $f_{\mathcal{L}}^{(1)}$ of Eq.(11).*



(a) Envelope function (b) Latitude function (c) Eigenvalues envelope

(d) δ_2 errors (e) L^2 errors

FIG 6. *Results for $d = 4$, the envelope $\mathbf{p}^{(2)}$ and the latitude $f_{\mathcal{L}}^{(2)}$ of Eq.*(11).



(a) Envelope function (b) Latitude function (c) Eigenvalues envelope

(d) δ_2 errors (e) L^2 errors

FIG 7. *Results for $d = 4$, the envelope $\mathbf{p}^{(3)}$ and the latitude $f_{\mathcal{L}}^{(3)}$ of Eq.*(11).

6. Applications

In this section, we apply the MRGG model to link prediction and hypothesis testing.

6.1. *Markovian dynamic testing*

As a first application of our model, we propose a hypothesis test to statistically distinguish between an independent sampling the latent positions and a Markovian dynamic. The null is then set to $\mathbb{H}_0$: *nodes are independent and uniformly distributed on the sphere* (i.e., *no Markovian dynamic*). Our test is based on estimate $\hat{f}_{\mathcal{L}}$ of latitude and thus the null can be rephrased as $\mathbb{H}_0 : \; f_{\mathcal{L}} = f^0_{\mathcal{L}}$ where $f^0_{\mathcal{L}}$ is the latitude of uniform law, dynamic is then i.i.d. dynamic.

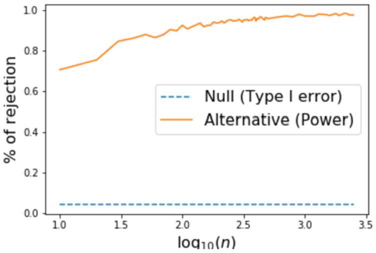


FIG 8. *Hypothesis testing.*

Figure 8 shows the power of a hypothesis test with level 5% (Type I error). One can use any *black-box goodness-of-fit test* comparing $\hat{f}_{\mathcal{L}}$ to $f^0_{\mathcal{L}}$, and we choose χ^2-test discretizing $(-1, 1)$ in 70 regular intervals. Rejection region is calibrated (i.e., threshold of the χ^2-test here) by *Monte Carlo simulations under the null.* It allows us to control Type I error as depicted by dotted blue line. We choose alternative given by Heaviside envelope $\mathbf{p}^{(1)}$ and latitude $f^{(1)}_{\mathcal{L}}$ of Eq. (11). We run our algorithm to estimate latitude from which we sample a batch to compute the χ^2-test statistic. We see that for graphs of size larger than $1,000$, the rejection rate is almost 1 under the alternative (Type II error is almost zero), the test is very powerful.

6.2. *Link prediction*

Suppose that we observe a graph with n nodes. Link prediction is the task that consists in estimating the probability of connection between a given node of the graph and the upcoming node.

6.2.1. *Bayes link prediction*

We propose to show the usefulness of our model solving a link prediction problem. Let us recall that we do not estimate the latent positions but only the *pairwise distances* (embedding task is not necessary for our purpose). Denoting by $\mathrm{proj}_{X_n^\perp}(\cdot)$ the orthogonal projection onto the orthogonal complement of $\mathrm{Span}(\mathrm{X_n})$, the decomposition of $\langle X_i, X_{n+1}\rangle$ defined by

$$\begin{aligned}&\langle X_i, X_n\rangle\langle X_n, X_{n+1}\rangle\\&+\sqrt{1-\langle X_n, X_{n+1}\rangle^2}\sqrt{1-\langle X_i, X_n\rangle^2}\langle\frac{\mathrm{proj}_{X_n^\perp}(X_i)}{\|\mathrm{proj}_{X_n^\perp}(X_i)\|_2}, Y_{n+1}\rangle,\end{aligned}\tag{12}$$

shows that latent distances are enough for link prediction. Indeed, it can be achieved using a *forward step* on our Markovian dynamic, giving the posterior probability (cf. Definition 2) $\eta_i(\mathbf{D}_{1:n})$ defined by

$$\int_{[-1,1]^2} \mathbf{p}\left(\langle X_i, X_n\rangle r+\sqrt{1-r^2}\sqrt{1-\langle X_i, X_n\rangle^2}u\right) f_{\mathcal{L}}(r) w_{\frac{d-3}{2}}(u)\frac{\Gamma(\frac{d-1}{2})}{\Gamma(\frac{d-2}{2})\sqrt{\pi}}drdu,\tag{13}$$

where $w_{\frac{d-3}{2}}(u) := (1-u^2)^{\frac{d-3}{2}-\frac{1}{2}}$ and where $\Gamma : a \in]0,+\infty[\mapsto \int_0^{+\infty} t^{a-1}e^{-t}dt.$

Definition 2 (Posterior probability function)**.** *The posterior probability function η is defined for any latent pairwise distances $\mathbf{D}_{1:n} = (\langle X_i, X_j\rangle)_{1\le i,j\le n} \in [-1,1]^{n\times n}$ by*

$$\forall i \in [n], \quad \eta_i(\mathbf{D}_{1:n}) = \mathbb{P}\left(A_{i,n+1} = 1 \mid \mathbf{D}_{1:n}\right),$$

where $A_{i,n+1} \sim \mathcal{B}\left(\mathbf{p}(\langle X_i, X_{n+1}\rangle)\right)$ is a random variable that equals 1 if there is an edge between nodes i and $n+1$, and is zero otherwise.

We consider a classifier g (cf. Definition 3) and an algorithm that, given some latent pairwise distances $\mathbf{D}_{1:n}$, estimates $A_{i,n+1}$ by putting an edge between nodes X_i and X_{n+1} if $g_i(\mathbf{D}_{1:n})$ is 1.

Definition 3. *A classifier is a function which associates to any pairwise distances $\mathbf{D}_{1:n} = (\langle X_i, X_j\rangle)_{1\le i,j\le n}$, a label $(g_i(\mathbf{D}_{1:n}))_{i\in[n]} \in \{0,1\}^n$.*

The risk of this algorithm is as in *binary classification*,

$$\begin{aligned}\mathcal{R}(g, \mathbf{D}_{1:n}) &:= \frac{1}{n}\sum_{i=1}^{n}\mathbb{P}\left(g_i(\mathbf{D}_{1:n}) \neq A_{i,n+1} \mid \mathbf{D}_{1:n}\right)\\&= \frac{1}{n}\sum_{i=1}^{n}\left\{(1-\eta_i(\mathbf{D}_{1:n}))\mathbb{1}_{g_i(\mathbf{D}_{1:n})=1} + \eta_i(\mathbf{D}_{1:n})\mathbb{1}_{g_i(\mathbf{D}_{1:n})=0}\right\},\end{aligned}\tag{14}$$

where we used the independence between $A_{i,n+1}$ and $g_i(\mathbf{D}_{1:n})$ conditionally on $\sigma(\mathbf{D}_{1:n})$. Pushing further this analogy, we can define the classification error of some classifier g by $L(g) = \mathbb{E}\left[\mathcal{R}(g, \mathbf{D}_{1:n})\right]$. Proposition 1 shows that the Bayes estimator - introduced in Definition 4 - is optimal for the risk defined in Eq. (14).

Definition 4 (Bayes estimator)**.** *We keep the notations of Definition 2. The Bayes estimator g^* of $(A_{i,n+1})_{1\le i\le n}$ is defined by*

$$\forall i \in [n], \quad g_i^*(\mathbf{D}_{1:n}) = \begin{cases} 1 & \textit{if } \eta_i(\mathbf{D}_{1:n}) \ge \frac{1}{2}\\ 0 & \textit{otherwise.}\end{cases}$$

Proposition 1 (Optimality of the Bayes classifier for the risk $\mathcal{R}$)**.** *We keep the notations of Definitions 2 and 4. For any classifier g, it holds for all $i \in [n]$,*

$$\begin{aligned}&\mathbb{P}\left(g_i(\mathbf{D}_{1:n}) \neq A_{i,n+1} \mid \mathbf{D}_{1:n}\right) - \mathbb{P}\left(g_i^*(\mathbf{D}_{1:n}) \neq A_{i,n+1} \mid \mathbf{D}_{1:n}\right)\\ &= 2\left|\eta_i(\mathbf{D}_{1:n}) - \frac{1}{2}\right| \times \mathbb{E}\left\{\mathbb{1}_{g_i(\mathbf{D}_{1:n}) \neq g_i^*(\mathbf{D}_{1:n})} \mid \mathbf{D}_{1:n}\right\},\end{aligned}$$

which immediately implies that

$$\mathcal{R}(g, \mathbf{D}_{1:n}) \geq \mathcal{R}(g^*, \mathbf{D}_{1:n}) \text{ and therefore } L(g) \geq L(g^*).$$

6.2.2. *Heuristic for link prediction*

One natural method to approximate the Bayes classifier from the previous section is to use the *plug-in approach.* This leads to the MRGG classifier introduced in Definition 5.

Definition 5 (The MRGG classifier)**.** *For any n and any $i \in [n]$, we define $\hat{\eta}_i(\mathbf{D}_{1:n})$ as*

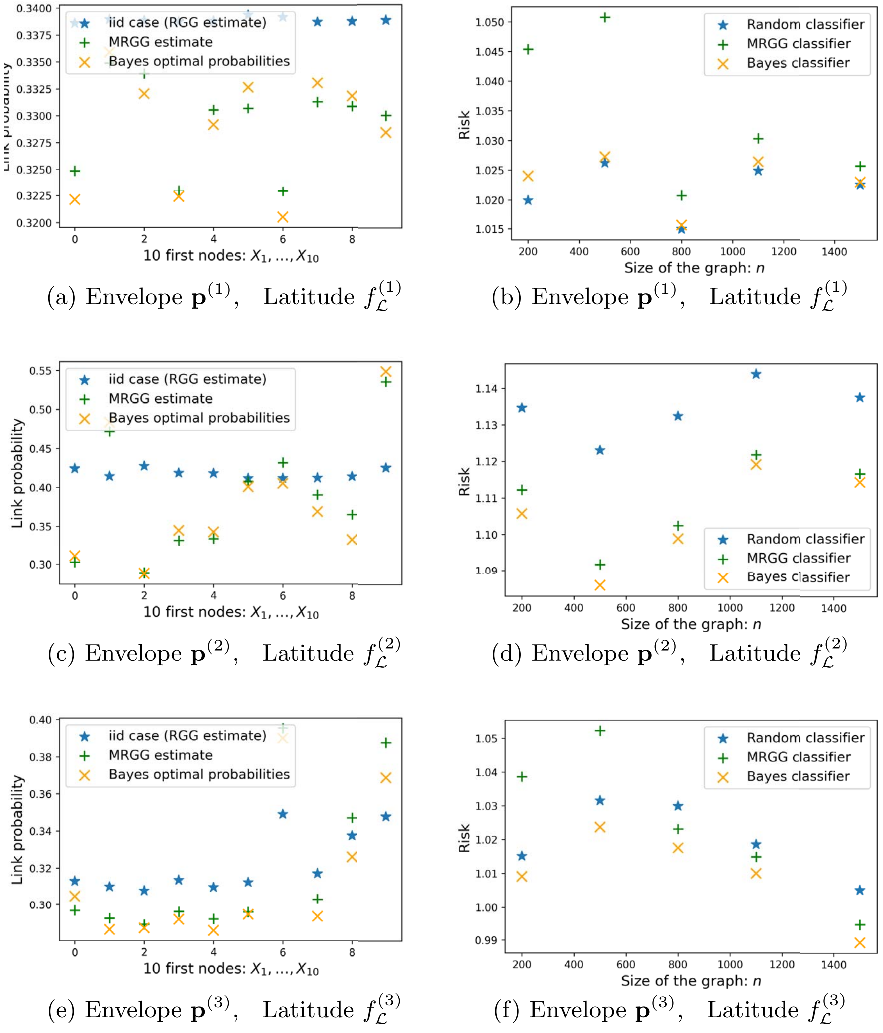
$$\int \widehat{\mathbf{p}}\left(\widehat{r}_{i,n} r + \sqrt{1-r^2}\sqrt{1-\widehat{r}_{i,n}^2}\, u\right) \hat{f}_{\mathcal{L}}(r) w_{\frac{d-3}{2}}(u) \frac{\Gamma(\frac{d-1}{2})}{\Gamma(\frac{d-2}{2})\sqrt{\pi}} dr du, \tag{15}$$

where $\widehat{\mathbf{p}}$ and $\hat{f}_{\mathcal{L}}$ denote respectively the estimate of the envelope function and the latitude function with our method and where $\widehat{r} := n\widehat{G}$. The MRGG classifier is defined by

$$\forall i \in [n], \quad g_i^{MRGG}(\mathbf{D}_{1:n}) = \begin{cases} 1 & \text{if } \hat{\eta}_i(\mathbf{D}_{1:n}) \geq \frac{1}{2} \\ 0 & \text{otherwise.}\end{cases}$$

To illustrate our approach we work with a graph of 1500 nodes with $d = 4$, and we consider the envelope and latitude functions defined in Eq. (11). The plots on the left column of Figure 9 show that we are able to recover the probabilities of connection of the nodes already present in the graph with the coming node X_{n+1}. Using the decomposition of $\langle X_i, X_{n+1}\rangle$ given by Eq. (12), orange crosses are computed using Eq. (13). Green plus are computed similarly replacing $\mathbf{p}$ and $f_{\mathcal{L}}$ by their estimations $\widehat{\mathbf{p}}$ and $\hat{f}_{\mathcal{L}}$ following Eq. (15). Blue stars are computed using Eq. (13) by replacing $f_{\mathcal{L}}$ by $\frac{w_\beta}{\|w_\beta\|_1}$ (with $\beta = \frac{d-2}{2}$) which implicitly supposes that the points are sampled uniformly on the sphere.

With the plots on the left column of Figure 9, we compare the risk of the *random* classifier - whose guess $g_i(\mathbf{D}_{1:n})$ is a Bernoulli random variable with parameter given by the ratio of edges compared to complete graph - with the risk of the MRGG classifier (cf. Definition 5). These figures show that for a small number of nodes, the risk estimate provided by the MRGG classifier can be significantly far from the one of the Bayes classifier. However, when the number of nodes is getting larger, the MRGG classifier gives similar results compared to the optimal Bayes classifier. This risk estimate can be significantly smaller than the one of the random classifier (see for example the plots corresponding to the envelope $\mathbf{p}^{(2)}$ and the latitude $f_{\mathcal{L}}^{(2)}$).



(a) Envelope $\mathbf{p}^{(1)}$, Latitude $f_{\mathcal{L}}^{(1)}$ (b) Envelope $\mathbf{p}^{(1)}$, Latitude $f_{\mathcal{L}}^{(1)}$

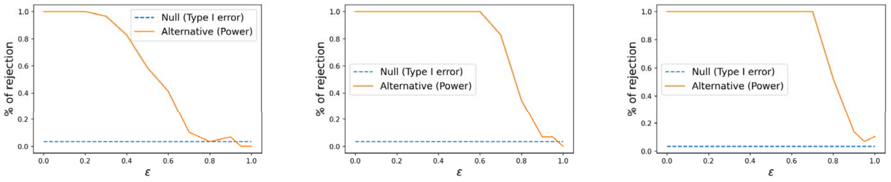
(c) Envelope $\mathbf{p}^{(2)}$, Latitude $f_{\mathcal{L}}^{(2)}$ (d) Envelope $\mathbf{p}^{(2)}$, Latitude $f_{\mathcal{L}}^{(2)}$

(e) Envelope $\mathbf{p}^{(3)}$, Latitude $f_{\mathcal{L}}^{(3)}$ (f) Envelope $\mathbf{p}^{(3)}$, Latitude $f_{\mathcal{L}}^{(3)}$

Fig 9. ← ***On the left:*** *Link predictions between the future node* X_{n+1} *and the* 10 *first nodes* $X_1, \ldots, X_{10}$. → ***On the right:*** *Comparison between the risk (defined in Eq.*(14)*) of the MRGG classifier, the random classifier and the risk of the optimal Bayes classifier.*

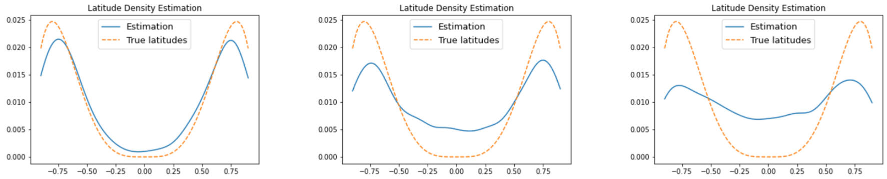
7. Discussion

In this section, we want to push the investigation of the performance of our estimation methods as far as possible. In Section 7.1 we study the robustness of our methods under model mispecification before inspecting the influence of the mixing time of the Markov chain $(X_i)_{i\geq 1}$ on the estimation error in Section 7.2. On a more theoretical side, we show that replacing the use of the complete linkage by the Ward distance in the SCCHEi algorithm, Theorem 2 might not be true anymore. We conclude with some remarks and by highlighting future research directions.



(a) $n = 200$ (b) $n = 500$ (c) $n = 1500$

Fig 10. *Studying the robustness of our method under model mispecification. We study the evolution of power for Markovian Dynamic Testing when the mixture parameter* ϵ *ranges* $(0,1)$*. We conduct this analysis for different values of* n*.*



(a) $\epsilon = 0.1$ (b) $\epsilon = 0.5$ (c) $\epsilon = 0.7$

Fig 11. *Studying the robustness of the estimation of the latitude function under model mispecification. We plot our kernel density estimator of the latitude function for* $n = 1500$, $d = 3$ *and for* $\epsilon \in \{0.1, 0.5, 0.7\}$*. We use the envelope* $\mathbf{p}^{(1)}$ *and latitude function* $f_{\mathcal{L}}^{(1)}$ *defined in Eq.*(11)*.*

7.1. Robustness to model mispecification

We consider a mixture model for the sampling scheme of the latent position. We fix some $\epsilon \in (0,1)$ and we draw X_1 randomly on the sphere. Then at time step $i \geq 2$, the point X_i is sampled as follows:

- with probability $1-\epsilon$, X_i is drawn following the Markovian dynamic described in Section 1 (based on X_{i-1}).
- with probability ϵ, X_i is drawn uniformly on the sphere.

Figure 10 and Figure 11 show the numerical results obtained under this mispecified model. We consider the hypothesis testing question presented in Section 6.1 with the same settings namely $d = 3$ and the envelope and latitude functions $\mathbf{p}^{(1)}$ and $f_{\mathcal{L}}^{(1)}$ of Eq. (11). We can see that when $\epsilon = 0$, the power of our test is 1 and we always reject the null hypothesis (uniform sampling of the latent positions) under the alternative. On the contrary, when $\epsilon = 1$, the points are sampled uniformly on the sphere and we obtain a power of the order of the level of our test (i.e. 5%) as expected. The larger the sample size n is, the greater ϵ can be chosen while keeping a large power. In the case where $n = 1500$, one can afford to sample 75% of latent positions uniformly (and the rest using our Markovian sampling scheme) while keeping a power equal to 1. Figure 11 shows that the larger ϵ is, the closer the estimated latitude function is to $\frac{w_\beta}{\|w_\beta\|_1} \equiv \frac{1}{2}$ (since $d = 3$) which corresponds to the density of a one-dimensional marginal of a uniform random point on $\mathbb{S}^{d-1}$.

7.2. Influence of mixing time on estimation error

In order to assert that the dependence of the latent variables has an influence on the estimation of the unknown functions of our model, we would require a minimax bound. The derivation of such minimax result is still an open problem, even in the independent setting (cf. [7]). Nevertheless, by making explicit the constants involved in concentration inequalities, we can show that the mixing time of the latent Markovian dynamic affects our bound on the δ_2 error between spectra. For any $r^* \in (-1, 1)$, let us consider the following latitude function

$$f_{\mathcal{L}}^{r^*}(r) := \frac{1}{I(r^*)}(1-r^2)^{\frac{d-3}{2}} \mathbb{1}_{r\in(r^*,1)}, \quad I(r^*) := \int_{r^*}^{1} (1-r^2)^{\frac{d-3}{2}} dr.$$

Note that the Markov transition kernel P of the chain $(X_i)_{i\geq 1}$ using this latitude function is the one that starting from a point $x \in \mathbb{S}^{d-1}$ samples uniformly a point in the set $\{z \in \mathbb{S}^{d-1} \mid \|x - z\|_2^2 \leq 2(1 - r^*)\}$. In particular, when $r^* = -1$, we recover the uniform distribution on the sphere. It is clear that the closer r^* to one, the larger the mixing time of the chain. One can show that for any $r^* \in (-1, 1)$, the chain is uniformly ergodic by proving that there exist an integer $m \geq 1$, a constant $\delta_m > 0$ and a probability measure ν such that

$$\forall x \in \mathbb{S}^{d-1},\ \forall A \in \Sigma, \quad P^m(x, A) \geq \delta_m \nu(A) \quad \text{(cf. Definition 9)}. \tag{16}$$

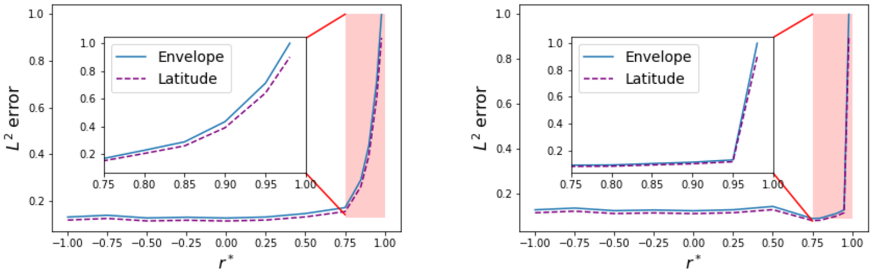
Eq. (16) holds by considering for example $\nu = \pi$ the uniform distribution on the sphere. It is straightforward to show that the smallest integer $m(r^*) \geq 1$ satisfying Eq. (16) is larger than $\frac{2}{1-r^*}$.[2] Taking a closer look at the constants involved in the concentration inequality from [10] (cf. [10, Section 3.1.1]), we get that

$$\mathbb{E}\left[\delta_2^2(\lambda(\mathbb{T}_W), \lambda(T_n)) \vee \delta_2^2(\lambda(\mathbb{T}_W), \lambda^{R_{opt}}(\widehat{T}_n))\right] < C \left[\frac{n}{\log^2(n)}\right]^{-\frac{2s}{2s+d-1}},$$

where $C > m(r^*)^2 \tau(r^*)^2 \|f_{\mathcal{L}}^{r^*}\|_\infty$ and $\tau(r^*) \geq 1$ is the Orlicz norm of some regeneration time. Since for any $0 < r^* < 1$,

$$I(r^*) = \int_{r^*}^{1} (1-r^2)^{\frac{d-3}{2}} dr = \int_0^{1-r^*} e^{\frac{d-3}{2}\ln(1-(r+r^*)^2)} dr$$

[2] Indeed, the latitude function $f_{\mathcal{L}}^{r^*}$ allows to make a jump at each time step of size at most $1 - r^*$. Since the length of the shortest arc on $\mathbb{S}^{d-1}$ joining the north pole to the south pole is 2, the result follows.



(a) $n = 200$

(b) $n = 1500$

FIG 12. *Studying the influence of the mixing time of the chain on the L^2 errors between (i) the envelope function and its estimate (using our adaptive procedure), and (ii) the latitude function and its estimate obtained with a kernel estimator.*

$$
\begin{aligned}
&= (1-(r^*)^2)^{\frac{d-3}{2}} \int_0^{1-r^*} e^{\frac{d-3}{2}\left\{\ln(1-(r+r^*)^2)-\ln(1-(r^*)^2)\right\}} dr \\
&\leq (1-(r^*)^2)^{\frac{d-3}{2}} \int_0^{1-r^*} e^{-\frac{d-3}{2}\left\{\frac{2rr^*+r^2}{1-(r^*)^2}\right\}} dr \\
&\leq (1-(r^*)^2)^{\frac{d-3}{2}} \int_0^{1-r^*} e^{-\frac{d-3}{2}\left\{2rr^*+r^2\right\}} dr \\
&\leq (1-(r^*)^2)^{\frac{d-3}{2}} \int_0^{1} e^{-\frac{d-3}{2}\left\{2rr^*\right\}} dr \\
&\leq (1-(r^*)^2)^{\frac{d-3}{2}} \left(1 \wedge \frac{1}{r^*(d-3)}\right),
\end{aligned}
$$

we get that $\|f_{\mathcal{L}}^{r^*}\|_\infty \geq \frac{1}{I(r^*)}(1-(r^*)^2)^{\frac{d-3}{2}} \geq r^*(d-3)$. Finally we obtain

$$
C > \frac{2r^*}{1-r^*}(d-3),
$$

where $r^* \mapsto \frac{2r^*}{1-r^*}(d-3)$ is increasing in r^* and diverges to $+\infty$ when $r^* \to 1^-$. Hence, the closer r^* is to one, the slower the chain is mixing, and the poorer is our bound.

Figure 12 presents the result of the simulations using the latitude function $f_{\mathcal{L}}^{r^*}$ and the envelope function $\mathbf{p} : t \mapsto \mathbb{1}_{t\geq 0}$. We compute the L^2 error between the true and the estimated envelope functions (respectively the true and the estimated latitude functions). When r^* is getting closer to 1, the chain is mixing slowly and we need to increase the sample size if we want to prevent the L^2 errors from blowing up. Graphs have been generated with a latent dimension $d = 3$ and by sampling the latent positions using our isotropic sampling procedure with latitude function $f_{\mathcal{L}}^{r^*}$.

7.3. Choice of the clustering algorithm for the SCCHEi

The SCCHEi algorithm relies on the clustering of the eigenvalues of the adjacency matrix provided by the HAC with complete linkage. In this section, we motivate the use of the HAC algorithm with complete linkage by showing that the theoretical results from Section 2.3 could be much more involved to establish by using another clustering procedure. Indeed, if one would consider for example the HAC with the Ward distance, the theoretical result obtained for the correctness of the SCCHEi algorithm (cf. Theorems 2 and 3) is likely to be no longer true (even if the sample size n is chosen arbitrarily large). Let us show this on a simple example.

We fix a resolution level $R = 2$ and we consider some $\Delta^G > 0$. We set $p_0^* = 4\Delta^G$, $p_1^* = 3\Delta^G$, $p_2^* = 2\Delta^G$, and $p_k^* = 0$ for all $k \geq 3$. Let us consider some $g \in (0, \Delta^G/4)$ that can be taken arbitrarily small. Let us denote $\lambda^R(\widehat{T}_n) = (\hat{\lambda}_1, \dots, \hat{\lambda}_{\widetilde{R}}, 0, 0 \dots)$ and assume that it holds $\hat{\lambda}_1 = p_0^*$, $\hat{\lambda}_2 = \cdots = \hat{\lambda}_{d+1} = p_1^*$ (we recall that $d_1 = d$), $\hat{\lambda}_{d+2} = \cdots = \hat{\lambda}_{d+1+\lfloor d_2/2 \rfloor} = p_2^* + g$ and $\hat{\lambda}_{d+2+\lfloor d_2/2 \rfloor} = \cdots = \hat{\lambda}_{1+d+d_2} = p_2^* - g$. To simplify the presentation, we will assume in the following that $d_2 = \frac{(d+1)d}{2} - 1$ is even (which holds for example if $d = 2k$ for any $k \geq 1$ odd). Figure 13 gives a visualization of this example.

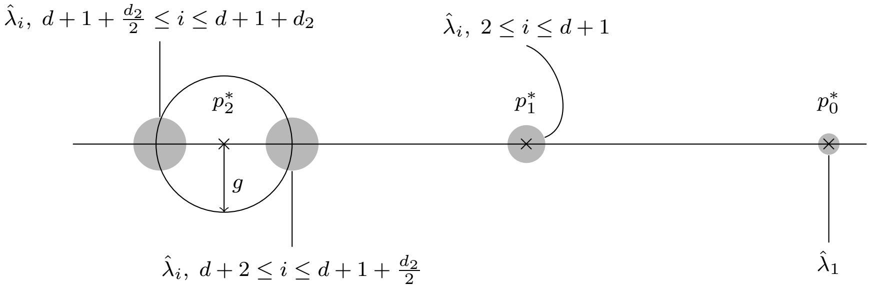

$\hat{\lambda}_i,\ d+1+\frac{d_2}{2} \leq i \leq d+1+d_2$
$\hat{\lambda}_i,\ 2 \leq i \leq d+1$
p_2^*
p_1^*
p_0^*
g
$\hat{\lambda}_i,\ d+2 \leq i \leq d+1+\frac{d_2}{2}$
$\hat{\lambda}_1$

FIG 13. *Visualization of the eigenvalues of the envelope function of our example.*

Applying the HAC algorithm (with the Ward distance) to the eigenvalues $(\hat{\lambda}_1, \dots, \hat{\lambda}_{\widetilde{R}})$, it is obvious that the state reached after $\widetilde{R} - 4 = 1 + d + d_2 - 4$ iterations in the HAC procedure will be

$$\begin{aligned} \widehat{\mathcal{G}}_0 &:= \{\hat{\lambda}_1\} \\ \widehat{\mathcal{G}}_1 &:= \{\hat{\lambda}_i \mid 2 \leq i \leq d\} \\ \widehat{\mathcal{G}}_2 &:= \{\hat{\lambda}_i \mid d+2 \leq i \leq d+1+d_2/2\} \\ \widehat{\mathcal{G}}_3 &:= \{\hat{\lambda}_i \mid d+2+d_2/2 \leq i \leq 1+d+d_2\} \end{aligned}$$

Hence, in order to understand which clusters will be merged at the next step of the HAC algorithm, we compute the Ward distance between the different clusters. Let us recall that for two finite and non-empty sets $S, S' \subset \mathbb{R}$ with respective cardinality $\#S$ and $\#S'$, the Ward distance between S and S' is given by

$$d_W(S, S') := \frac{\#S \times \#S'}{\#S + \#S'} \left(\frac{1}{\#S} \sum_{x_s \in S} x_s - \frac{1}{\#S'} \sum_{x'_s \in S'} x'_s \right)^2 .$$

Ward distances between clusters

	$\widehat{\mathcal{G}}_1$	$\widehat{\mathcal{G}}_2$	$\widehat{\mathcal{G}}_3$
$\widehat{\mathcal{G}}_0$	$\frac{d}{d+1}(\Delta^G)^2$	$\frac{d_2}{d_2+2}(2\Delta^G - g)^2$	$\frac{d_2}{d_2+2}(2\Delta^G + g)^2$
$\widehat{\mathcal{G}}_1$		$\frac{d \times d_2}{2d+d_2}(\Delta^G - g)^2$	$\frac{d \times d_2}{2d+d_2}(\Delta^G + g)^2$
$\widehat{\mathcal{G}}_2$			$d_2 \times g^2$

We deduce that all Ward distances between pair of clusters are scaling at least linearly with d except the Ward distances between $\widehat{\mathcal{G}}_0$ and the other three clusters $\widehat{\mathcal{G}}_1$, $\widehat{\mathcal{G}}_2$ and $\widehat{\mathcal{G}}_3$. Indeed, for any $i \in \{1, 2, 3\}$, $d_W(\widehat{\mathcal{G}}_0, \widehat{\mathcal{G}}_i)$ remains bounded independently of the latent dimension d. Hence, for any $g \in (0, \Delta^G/4)$ which can be chosen arbitrarily small, one can take d large enough to ensure that

$$\max \left\{ d_W(\widehat{\mathcal{G}}_0, \widehat{\mathcal{G}}_i) \ , \ i \in \{1, 2, 3\} \right\} < d_W(\widehat{\mathcal{G}}_2, \widehat{\mathcal{G}}_3). \tag{17}$$

We deduce that for any $g \in (0, \Delta^G/4)$, we can choose d large enough to ensure that Eq. (17) holds and thus the clusters merged between depths 4 and 3 from the root of the HAC's tree will not be $\widehat{\mathcal{G}}_2$ and $\widehat{\mathcal{G}}_3$. This means that the state obtained at depth 3 from the root is not of type $(\mathcal{S})$ (in the sense defined in Lemma 2).

If this is not a sufficient condition to state that the SCCHEi will fail to recover the correct clusters, this example shows that the use of Ward distance can lead to some unexpected clustering of the eigenvalues. Our example proves that using the HAC algorithm with the Ward distance, the result of Lemma 2 does not hold anymore. Namely, regardless of how large the sample size is chosen, there are situations (in particular for a large latent dimension) where the states of type $(\mathcal{S})$ (cf. Lemma 2) are never reached in the HAC tree with the Ward distance. Hence obtaining a theoretical guarantee for the clustering provided by the SCCHEi in this framework may be impossible or at least much more involved.

7.4. Concluding remarks

7.4.1. Estimation of the latent dimension

The proposed methods implicitly assume that the latent dimension d is known. The article [1] proved that the latent dimension d can be easily recovered in

practice for n large enough provided that the spectral gap condition (9) holds. In the following, we briefly describe their approach.
Given some matrix $\widehat{T}_n$ as input and some set of candidates $\mathcal{D}$ for the dimension d (typically $\mathcal{D} = \{2, 3, \dots, d_{\max}\}$), apply the Algorithm HEiC (cf. Algorithm 3 in Section F) for any $d_c \in \mathcal{D}$ and store the returned value $gap := gap(d_c)$. Let us recall that $gap(d_c)$ corresponds to the largest gap between a bulk of d_c eigenvalues of $\widehat{T}_n$ and the rest of the spectrum (see the definition of Gap_1 in Section F for details). Once we have computed the different gaps, we pick the candidate d_c that led to the largest one. Given the guarantees provided by Proposition 4, the previously described procedure will find the correct dimension, with high probability (on the event $\mathcal{E}$ with the notations of Proposition 4), if the true dimension of the latent space is in the candidate set $\mathcal{D}$.

7.4.2. Future research directions

Our work encourages the development of growth model in random graphs and in particular the derivation of similar results in MRGGs with other latent spaces. It would be also desirable to extend our methods to the case where we consider more complex Markovian sampling of the latent positions, typically one that is not isotropic. Our work leaves open the question of getting a theoretical guarantee for the estimation of the latitude function. If we proved (with Theorem 4) that we can consistently estimate the Gram matrix of the latent positions in Frobenius norm, this is not sufficient to ensure that our kernel density estimator is consistent since we cannot ensure that $\frac{1}{n-1}\sqrt{\sum_{i=2}^{n}(r_i - \hat{r}_i)^2}$ tends to 0 as n goes to $+\infty$. Deriving a theoretical result regarding the estimation of the latitude function seems challenging and we believe that it would require significantly different proof techniques.

Supplementary Material

Appendix for "Markov random geometric graph, MRGG: A growth model for temporal dynamic networks"
(doi: 10.1214/22-EJS1969SUPP; .pdf). Sections A to C: basic definitions and complements. Sections D to F: algorithms and experiments. Sections G to I: proofs of theoretical results.

References

[1] Araya, E. and De Castro, Y. (2019). Latent Distance Estimation for Random Geometric Graphs. In *Advances in Neural Information Processing Systems* 8721–8731.

[2] Arlot, S. (2019). Minimal penalties and the slope heuristics: a survey. *Journal de la Société Française de Statistique* **160** 1–106. MR4021408

[3] Backstrom, L., Huttenlocher, D., Kleinberg, J. and Lan, X. (2006). Group formation in large social networks: membership, growth, and evolution. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining* 44–54.

[4] Bandeira, A. S. and van Handel, R. (2016). Sharp nonasymptotic bounds on the norm of random matrices with independent entries. *Ann. Probab.* **44** 2479–2506. MR3531673

[5] Bhatia, R. (1996). *Matrix Analysis. Graduate Texts in Mathematics.* Springer New York. MR1477662

[6] Bubeck, S., Ding, J., Eldan, R. and Rácz, M. Z. (2016). Testing for high-dimensional geometry in random graphs. *Random Structures & Algorithms* **49** 503–532. MR3545825

[7] De Castro, Y., Lacour, C. and Ngoc, T. M. P. (2020). Adaptive Estimation of Nonparametric Geometric Graphs. *Mathematical Statistics and Learning.* MR4165265

[8] Clementi, A. E. F., Pasquale, F., Monti, A. and Silvestri, R. (2009). Information spreading in stationary Markovian evolving graphs. *2009 IEEE International Symposium on Parallel & Distributed Processing.*

[9] Dai, F. and Xu, Y. (2013). *Approximation theory and harmonic analysis on spheres and balls.* Springer. MR3060033

[10] Duchemin, Q., de Castro, Y. and Lacour, C. (2020). Concentration inequality for U-statistics of order two for uniformly ergodic Markov chains.

[11] Durante, D. and Dunson, D. (2014). Bayesian logistic gaussian process models for dynamic networks. In *Artificial Intelligence and Statistics* 194–201. PMLR.

[12] Díaz, J., Mitsche, D. and Pérez-Giménez, X. (2008). On the connectivity of dynamic random geometric graphs. MR2487629

[13] Fan, J., Jiang, B. and Sun, Q. (2018). Hoeffding's lemma for Markov Chains and its applications to statistical learning.

[14] Ferré, D., Hervé, L. and Ledoux, J. (2012). Limit theorems for stationary Markov processes with L2-spectral gap. *Annales de l'Institut Henri Poincaré, Probabilités et Statistiques* **48** 396–423. MR2954261

[15] Gilles, P. (1989). *The volume of convex bodies and Banach space geometry.* Cambridge University Press. MR1036275

[16] Higham, D. J., Rašajski, M. and Pržulj, N. (2008). Fitting a geometric graph to a protein–protein interaction network. *Bioinformatics* **24** 1093-1099.

[17] Hoff, P. D., Raftery, A. E. and Handcock, M. S. (2002). Latent Space Approaches to Social Network Analysis. *Journal of the American Statistical Association* **97** 1090-1098. MR1951262

[18] Jiang, B., Sun, Q. and Fan, J. (2018). Bernstein's inequality for general Markov Chains.

[19] Jin, E. M., Girvan, M. and Newman, M. E. (2001). Structure of growing social networks. *Physical review E* **64** 046132.

[20] Klopp, O., Tsybakov, A. B. and Verzelen, N. (2017). Oracle inequal-

ities for network models and sparse graphon estimation. *The Annals of Statistics* **45** 316–354. MR3611494

[21] Koltchinskii, V. and Giné, E. (2000). Random Matrix Approximation of Spectra of Integral Operators. *Bernoulli* **6**. MR1781185

[22] Lo, C., Cheng, J. and Leskovec, J. (2017). Understanding Online Collection Growth Over Time: A Case Study of Pinterest. In *Proceedings of the 26th International Conference on World Wide Web Companion. WWW '17 Companion* 545–554. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE.

[23] Lovász, L. (2012). *Large Networks and Graph Limits*. American Mathematical Society. MR3012035

[24] Matias, C. and Miele, V. (2017). Statistical clustering of temporal networks through a dynamic Stochastic Block Model. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **79** 1119–1141. MR3689311

[25] Meyn, S. and Tweedie, R. (1993). *Markov Chains and Stochastic Stability* **92**. MR1287609

[26] Penrose, M. (2003). Random Geometric Graphs. MR1986198

[27] Roberts, G. O. and Rosenthal, J. S. (2004). General state space Markov chains and MCMC algorithms. *Probability Surveys* **1** 20–71. MR2095565

[28] Rossetti, G. and Cazabet, R. (2018). Community discovery in dynamic networks: a survey. *ACM Computing Surveys (CSUR)* **51** 1–37.

[29] Staples, R. S. G. S. (2009). Dynamic Geometric Graph Processes: Adjacency Operator Approach. MR2737624

[30] Tang, M., Sussman, D. L. and Priebe, C. E. (2013). Universally consistent vertex classification for latent positions graphs. *The Annals of Statistics* **41** 1406–1430. MR3113816

[31] Ugander, J., Backstrom, L., Marlow, C. and Kleinberg, J. (2012). Structural diversity in social contagion. *Proceedings of the National Academy of Sciences* **109** 5962–5966.

[32] Xu, K. S. and Hero, A. O. (2014). Dynamic Stochastic Blockmodels for Time-Evolving Social Networks. *IEEE Journal of Selected Topics in Signal Processing* **8** 552-562.

[33] Yang, S. and Koeppl, H. (2018). Dependent Relational Gamma Process Models for Longitudinal Networks. In *Proceedings of the 35th International Conference on Machine Learning* (J. Dy and A. Krause, eds.). *Proceedings of Machine Learning Research* **80** 5551–5560. PMLR.

[34] Yu, Y., Wang, T. and Samworth, R. J. (2014). A useful variant of the Davis–Kahan theorem for statisticians. *Biometrika* **102** 315-323. MR3371006