# THÈSE

En vue de l'obtention du

## DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

**Délivré par :** *l'Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)*

**Présentée et soutenue le** *05.10.2017* **par :**

IOANA ALEXANDRA GAVRA

## Algorithmes stochastiques d'optimisation sous incertitude sur des structures complexes. Convergence et applications.

**JURY**

| | | |
|---|---|---|
| MARC ARNAUDON | Professeur d'Université | Membre du Jury |
| ARNAUD DOUCET | Professeur d'Université | Rapporteur |
| SÉBASTIEN GADAT | Professeur d'Université | Directeur de thèse |
| LAURENT MICLO | Directeur de recherche | Directeur de thèse |
| ERIC MOULINES | Professeur d'Université | Membre du Jury |
| ALAIN TROUVÉ | Professeur d'Université | Rapporteur |
| NATHALIE VILLA-VIALANEIX | Chargée de recherche | Membre du Jury |

**École doctorale et spécialité :**

*MITT : Domaine Mathématiques : Mathématiques appliquées*

**Unité de Recherche :**

*Institut de Mathématiques de Toulouse*

**Directeur(s) de Thèse :**

*Sébastien Gadat* et *Laurent Miclo*

**Rapporteurs :**

*Arnaud Doucet* et *Alain Trouvé*

# Remerciements

Je vais commencer mes remerciements de manière classique, c'est-à-dire par mes directeurs de thèse Sébastien Gadat et Laurent Miclo. Je vous remercie de m'avoir initiée à la recherche mathématique et d'avoir partagé une partie de votre connaissances avec moi. Je remercie Sébastien, pour tout ce qu'il m'a appris, pour toutes ses lectures et relectures (pas toujours faciles), pour son soutien, ses conseils, son optimisme et sa bonne humeur. Je remercie Laurent, dont la passion pour le vaste domaine de mathématiques et la rigueur sont toujours une source de motivation.

Je voudrais aussi remercier Arnaud Doucet et Alain Trouvé d'avoir accepté de rapporter ma thèse et pour le temps et l'attention accordés à ce manuscrit. Je tiens également à remercier Marc Arnaudon, Eric Moulines et Nathalie Villa-Vialaneix pour leur participation au jury. Merci à Nathalie pour les conseils qu'elle a pu me donner concernant les simulations pendant ma première année de thèse et à Marc, ses travaux avec Laurent étant une des principales sources d'inspiration pour ma thèse.

Merci Laurent Risser, c'est toujours un plaisir de travailler avec toi. J'ai aussi beaucoup apprécié la collaboration avec Clément et nos travaux ont été l'objet de nombreuses suggestions et remarques de la part de Sébastien Gerchinovitz, que je souhaite remercier pour ça.

Je remercie Marie-Line Domenjole, Françoise Michel, Martine Labruyère et Agnès Requis pour leur amabilité et efficacité.

Merci à tous mes enseignants de Master. En particulier, à Mihai Maris pour son soutien et sa générosité incroyable et à Jean-Marc Bouclet sans lequel je n'aurais jamais réussi à m'inscrire en première année.

Je remercie chaleureusement mes collègues de première année de Master, qui ont eu la patience de m'apprendre le français et sans lesquels la première année à Toulouse aurait été difficile. Merci à Nadège, pour son altruisme sans réserve et sa générosité, à Hugo C. pour sa bonne humeur et sans lequel j'aurais eu beaucoup plus de mal à trouver un appartement, à Maxime et Damien qui m'ont aussi aidé avec toutes sortes de démarches et à Hugo B. qui m'a confié son poisson rouge pendant quelque jours. Merci aussi à Alix et Edith pour leur accueil et à Elsa pour sa visite guidée de Bordeaux.

En plus de m'initier à la recherche, la thèse m'a offert beaucoup de liberté, la chance d'aborder des sujets qui m'ont beaucoup intéressée, d'avoir des directeurs de thèse qui ont su bien m'encadrer, mais aussi d'être entourée par les doctorants de l'IMT qui ont couronné ces trois dernières années de beaucoup de joie et de charme.

Merci à Claire et Kevin, qui pendant trois ans ont rempli notre bureau d'une ambiance très agréable et de bonne humeur. Merci pour vos conseils et aide, merci à toi Claire, pour ton sourire contagieux et à toi Kevin, pour les catapultes, l'escalade, et des autres choses que tu m'as fait découvrir. Merci aussi à Pierre et Olfa, avec lesquels j'ai aussi partagé le bureau (même si c'était moins longtemps), pour les bons moments passés ensemble.

Je veux aussi adresser des remerciements chaleureux à tous ceux qui ont participé à la pause habituelle entre midi et deux de la salle café : Anton (ainsi que pour ta spontanéité et originalité), Antoine P., Maylis & Valentin (merci aussi d'avoir eu l'initiative d'organiser les deux premiers championnats), Aurore, Kevin, Marc, Clément, Michael, Maël, Fabien et d'autres. Merci aussi à tous les autres doctorants qui ont participé à la bonne ambiance du laboratoire, Stéphane (pour tous nos petites batailles), Laure, Sylvain (merci de m'avoir fait découvrir ton endroit préféré de vacances), Fanny, Raphaël (pour les fois où on est rentrés à pied), Yuri, mon grand frère de thèse Sofiane, Benoît, Malika, Mélanie, Ibrahim, ma grande sœur de thèse Magali, Antoine B., Tristan (même si tu n'es pas doctorant), William, Silvère et ceux que j'ai pu oublier.

Guillaume, c'était merveilleux de partager ces trois dernières années avec toi. C'est difficile de te remercier en quelques mots pour tout ce que je voudrais, donc je te remercie simplement d'être là. Je veux aussi remercier tes parents pour leur gentillesse et les nombreux repas et apéros auxquels ils m'ont accueillie pendant ces trois ans.

Faptul că am ajuns să fac un doctorat în matematică se datorează şi anumitor persoane pe care le-am întâlnit de-a lungul vieţii. Nu pot să nu mulţumesc doamnei profesoare Anico Mureşan, pentru pasiunea şi încrederea pe care mi le-a trasmis, fără de care nu ştiu dacă aş fi urmat acelaşi drum. Aş dori de asemenea să mulţumesc câtorva profesori de la Universitatea de Vest Timişoara pentru rolul pe care l-au avut în formarea mea şi pentru sprijinul acordat: Claudia Zaharia, Radu Moleriu, Dorel Miheţ, Petre Birtea, Petre Jebeleanu. În afară de cadrele de care am avut parte în învăţământ, am avut norocul de a da şi peste alte persoane a căror pasiune pentru matematică m-a inspirat: Alex şi Iulian.

Pentru că fiecare persoană este definită de suma experienţelor pe care le trăieşte, aş dori să mulţumesc familiei mele, pentru tot sprijinul şi dragostea oferită, fără de care nu mă pot imagina: mami şi tati, Ama şi Andrei (mulţumesc şi pentru lecturi), Mamă, Tată, Tanti Jana, Momo, Tatiana, Cornel, Andra, Razvan, Lau şi Alexandra. Şi în avans, pentru toate bucuriile pe care o să ni le (mai) ofere, lui Victor şi lui Tudor. Mulţumesc şi lui Dia şi Dragoş, alături de care am împărtăşit nenumărate momente frumoase în ultimii 16 ani. Bineînteles, mulţumesc şi tuturor oamenilor deosebiţi pe care i-am întâlnit la cercetaşi şi pe care îi port în continuare în suflet.

# Vue d'ensemble de la thèse

Le point de départ de cette thèse a été la notion de barycentre sur des graphes pondérés. Pour présenter le cadre plus en détail, prenons l'exemple d'un graphe de citations. Les interactions, qui vont être décrites par des arêtes, entre les auteurs, qui vont être représentés par des sommets, sont modélisées de manière naturelle : on considère que deux auteurs sont directement liés si et seulement si ils ont publié ensemble. Pour mieux traduire leur relation, on munit l'arête associée d'un poids proportionnel au nombre de collaborations. De manière équivalente, on peut considérer une longueur de l'arête, inversement proportionnelle aux nombre de publications communes. Autrement dit, plus deux auteurs ont publié ensemble, plus ils sont proches. Un tel graphe est appelé pondéré. Le graphe considéré est également simple, dans le sens où entre deux sommets il y a au maximum une arête. On définit une mesure (de probabilité) sur les sommets pour modéliser le nombre de citations de chaque auteur : la mesure de chaque nœud est proportionnelle au nombre total de citations cumulées par l'auteur. Nous supposons que le graphe est connexe, c'est-à-dire qu'il n'existe pas deux groupes de personnes qui n'ont pas de collaborations entre eux. Un exemple est illustré en Figure 1.3 .

Une première difficulté rencontrée dans notre démarche est le fait qu'une notion de centralité ou moyenne sur des espaces métriques généraux n'est pas forcement évidente, car les opérations habituelles d'addition peuvent ne pas être définies comme dans le cadre euclidien. Cependant, cette question a été beaucoup étudiée et une réponse possible a été fournie par Fréchet en 1948 [61], qui a étendu la notion de médiane et de moyenne euclidienne aux espaces métriques en introduisant les moyennes qui aujourd'hui portent son nom.

La définition d'une moyenne de Fréchet est basée sur la remarque que, dans un espace euclidien, l'espérance d'une variable aléatoire $Y$ peut être vue comme le point $x^*$ qui minimise $\mathbb{E}[|x - Z|^2]$. En utilisant cette approche, on définit le barycentre d'un graphe $G$ comme étant un élément minimiseur de $U_\nu$, avec :

$$U_\nu(x) = \sum_{y \in V} d^2(x, y)\nu(y).$$

Pour revenir à notre exemple, le barycentre est l'auteur qui minimise la somme des carrés des distances, entre lui et les autres personnes, pondérées par leur nombre respectif de citations. On peut donc supposer qu'il a une grande influence sur les principales connections dans le graphe, et possiblement qu'il relie des auteurs importants (qui ont beaucoup de citations).

On voit donc que trouver le barycentre d'un graphe revient à optimiser une fonction. Si en plus on considère qu'on ne connait pas le nombre de citations de chaque auteur, et qu'on peut seulement observer chaque fois qu'un article est cité, notre problème devient un problème d'optimisation globale sous incertitude. A travers cet aspect, la question de trouver un barycentre sur un graphe a aussi ouvert des pistes de recherche dans ma thèse, qui ne sont pas forcément en lien avec des graphes. C'est la raison pour laquelle ce manuscrit est séparée en deux parties : une première partie concentrée autour des graphes et une deuxième concernant un problème général d'optimisation globale sous incertitude, dans un espace métrique fini. Les principaux objets étudiés et utilisés le long de cette thèse sont présentés plus en détails dans un premier chapitre introductif. Dans ce qui suit, on présente la structure globale du reste du manuscrit.

## Première partie

**Chapitre 2** Le premier chapitre de cette partie est dédié à l'étude de barycentres sur un graphe pondéré, muni d'une mesure de probabilité sur ses sommets. Comme on vient de le voir, cette étude se traduit par un problème d'optimisation globale. On se tourne donc vers une méthode classique : le recuit simulé. Pour contourner le fait que la mesure de probabilité n'est accessible que par des réalisations indépendantes (on peut seulement observer quand un article est cité), on utilise une technique d'homogénéisation sur une formulation de Langevin du recuit simulé. Ce choix nous permet d'utiliser les observations une par une, contrairement aux techniques de type Metropolis Hastings qui nécessite l'utilisation de « mini-lots ». Notre choix nous amène à considérer une version continue du graphe (aussi appelée quantique). L'algorithme proposé pour estimer le barycentre est donc une diffusion markovienne sur le graphe quantique. Dans le but d'établir la convergence en probabilité du processus, on étudie l'évolution de l'entropie relative de sa loi par rapport à une mesure de Gibbs bien choisie. En utilisant des inégalités fonctionnelles (Poincaré et Sobolev) et le lemme de Grönwall, on montre ensuite que l'entropie relative tend vers zéro. On montre que la méthode utilisée pour estimer le barycentre peut être adaptée pour optimiser d'autres fonctions, et en particulier pour estimer des $p$-moyennes de Fréchet pour $p \geq 1$. On applique notre méthode pour estimer le barycentre d'un graphe de citations, en utilisant des données fournies par ZBmath et on présente également des résultats obtenus sur des graphes provenant de réseaux sociaux et sur le graphe constitué des stations de métro de Paris.

**Chapitre 3** Les applications nous ont montré que la méthode proposée dans le chapitre 2 est très précise et relativement rapide en terme de temps de calculs, mais ils nous ont aussi permis de voir ses défauts. Pour simuler l'évolution du processus de Markov associé au recuit simulé, on est obligé de calculer des distances sur le graphe chaque fois qu'on utilise une observation. Pour limiter le nombre de calculs, on préfère stocker, dès le début, les distances entre toutes les paires de nœuds dans une matrice. Cette stratégie est tout à fait raisonnable sur des graphes de taille moyenne (notamment, le graphe de citations considéré a 13000 nœuds), mais devient vite coûteuse en terme de mémoire si on augmente encore plus la taille du graphe. Une possible réponse à cette difficulté est présentée au chapitre 3. On propose une heuristique basée sur la propriété d'associativité et dissociativité du barycentre dans un espace euclidien. Le graphe initial est séparé en plusieurs sous-graphes de taille plus petite, en utilisant des méthodes de clustering existantes, et un nouveau graphe est formé à partir des centres estimés de chaque sous-graphe. Ensuite, une estimation du barycentre de ce graphe à grande échelle est effectuée. Pour obtenir plus de precision, on peut effectuer une nouvelle estimation du barycentre sur un graphe multi-echelle obtenu en réinsérant le cluster central dans le graphe à grande échelle.

**Chapitre 4** Le chapitre 4 est dédié à une étude plus approfondie de la structure du graphe. En effet, jusqu'à maintenant nous ne sommes intéressés qu'au barycentre, qui dans un certain sens est l'équivalent d'une espérance. Pour découvrir les interactions qui synthétisent la structure du graphe en prenant en compte l'importance donnée aux sommets par la mesure de probabilité attachée, on se tourne vers le concept d'analyse en composante principale (ACP). Une première difficulté est, encore une fois, le fait que cet outil a été développé pour un espace euclidien et non pas pour un espace métrique quelconque. Pour le généraliser aux graphes, on utilise la même méthode que pour la moyenne, *i.e.* une formulation variationnelle. Cela nous permet de nous ramener à un problème d'optimisation. On s'intéresse particulièrement à la première composante principale. L'espace sur lequel on veut minimiser est l'espace des géodésiques du graphe. On se retrouve donc face à une deuxième difficulté : comment paramétriser l'espace pour faciliter la définition d'un processus de Markov qui suit une dynamique donnée par un recuit simulé homogénéisé. Après avoir établi une paramétrisation convenable, nous proposons un algorithme pour déterminer la première composante principale et conjecturons la convergence du processus de Markov associé vers l'ensemble ciblé, en présentant également des pistes de preuve.

### Deuxième partie

Dans cette partie, on change de cadre et les graphes ne font plus l'objet de notre intérêt. Notre but est de résoudre un problème d'optimisation stochastique globale sur un espace d'états fini. Les applications d'une telle démarche sont multiples. Prenons comme exemple l'optimisation d'une trajectoire de vol pour un avion par rapport à son coût (en terme de durée et consommation de carburant). Le coût d'une trajectoire fixée n'est pas connu à l'avance. Il peut seulement être estimé en utilisant des prédictions pour certains paramètres, comme les conditions météorologiques ou le poids des passagers. Donc, le choix d'un plan de vol est un problème d'optimisation stochastique.

**Chapitre 5** Le chapitre 5 propose une solution à un problème de ce type. Notre approche est inspirée du domaine général des méthodes Monte-Carlo et repose sur une chaîne de Markov dont la probabilité de transition à chaque étape est définie à l'aide de « mini-lots » de taille croissante (aléatoire). On montre la convergence en probabilité de l'algorithme vers l'ensemble optimal. La preuve est séparée en trois étapes. D'abord on calcule explicitement le générateur infinitésimal de notre recuit simulé bruité, ensuite on le compare à celui du recuit non-bruité et finalement on utilise l'inégalité de trou spectral fournie par [76] et le lemme de Grönwall pour conclure.

Une fois la convergence établie, nous nous sommes demandé quelle est la vitesse de convergence de l'algorithme et comment elle est influencée par le choix du schéma de température et le nombre d'observations utilisées à chaque itération. Pour répondre à cette question, on calcule une borne supérieure pour la vitesse de convergence. Ensuite on propose un choix de paramètres optimisés pour assurer un nombre minimal d'évaluations pour une précision donnée et un intervalle de confiance proche de 1. Ce travail est complété par un ensemble de simulations numériques qui illustrent la performance pratique de notre algorithme à la fois sur des fonctions tests et sur des données réelles issues de cas concrets, notamment de données liées à l'optimisation de trajectoires d'avions.

# Contents

# Chapter 1

# Introduction

In the first part of this chapter, we introduce the main objects and concepts that lay at the center of this thesis: graphs, Fréchet means and global optimization under uncertainty through simulated annealing. Then we also give a brief presentations of the main tools that will be used throughout this present work.

The starting point of this thesis was the question of finding Fréchet means on a graph. We shall see this as a first step in better understating a network. For example, we could expect that a barycenter in a citation network may reveal important historical thoughts hidden behind it. Seems thus natural to start by presenting some details on this concept.

## 1.1 Fréchet Means

At the foundation of this thesis lays the question of establishing a notion of centrality on graphs, namely a barycenter with respect to a given probability measure, and then proposing an algorithm that determines it.

For a probability measure defined on an Euclidean space, there are two classical notions of centrality: the median and the Euclidean mean. There are many mathematical objects that cannot be embedded in a Euclidean space. The geometry of graphs is not plane and thus they constitute an example of such an object. This is also true for other metric spaces like general Riemannian manifolds, the Wasserstein space of probability measures on $\mathbb{R}$, *etc.* It is thus natural to want to adapt these notions of centrality to a wider context. A first difficulty that one encounters when trying to extend these notions to a general metric space is the absence of the natural addition and averaging operations. In 1948 M. Fréchet presented in [61], a possible answer to this problem, not only for the median and the mean of a probability measure, but for moments of all orders. He introduced a notion of typical position of order $p$, for a random variable $Z$, defined on a general metric space $(\mathcal{E}, d)$ and distributed according to any probability measure $\nu$. This is now known as $p$-Fréchet mean (or simply $p$-mean) and is defined as:

$$M_\nu^{(p)} := \arg\min_{x\in\mathcal{E}} \mathbb{E}_{Z\sim\nu}[d^p(x, Z)].$$

It is important to notice that the only requirement for the definition of such a mean element is the specification of a metric on the space of interest. This also means that considering a different metric would define a different mean.

At first sight, this definition may seem slightly counter-intuitive, but at a closer look one can see that it is a rather natural one. We illustrate the idea that lies behind by a summary of the explanation that Fréchet himself presented in [61]. When we have a large data set, for example a table of numbers, and we want to determine a representative element, we first establish a way to measure the gap between two elements and then try to find an element that minimizes the sum of such gaps. If the gap between two numbers $a, b$ is defined as the absolute value $|a-b|$, we obtain the median of the set of numbers and if the gap is defined as the square of the difference, $(a-b)^2$, we obtain the classical arithmetic mean. This variational formulation also holds for real random variables. For example, if $Z$ is a random variable distributed according to a distribution $\nu$ on $\mathbb{R}^d$, its expected value given by $m_\nu = \int_{\mathbb{R}^d} x d\nu(x)$ is the point that minimizes:

$$x \longmapsto \mathbb{E}_{Z\sim\nu}[|x-Z|^2].$$

The most commonly studied $p$-mean is the one corresponding to the Euclidean mean, *i.e.* the case $p=2$. From now on we refer to it simply as Fréchet mean and denote it $M_\nu$.

### Application fields and concrete examples

The notion of Fréchet mean appears in numerous fields and we do not intend to present an exhaustive list, but rather to give a general idea of its potential uses. One of the fields of application is signal averaging. In [26], the author proposes an approached based on Fréchet means to estimate the average heart cycle from electrocardiogram (ECG) records.

Another example occurs in the field of image analysis. It is rather natural to consider closer two images that differ only by a smooth deformation than two images that differ by a chaotic deformation, even though they might be at an equal Euclidean distance. Thus it is necessary, once more to renounce the Euclidean setting. To emphasize the fact that the Euclidean mean is not always adequate in this setting, we provide an illustration in Figure 1.1. The Fréchet mean can be considered as a basic tool for analyzing the properties of an image([111, 9]). Image analysis is used among others, in medical research. The construction of average models of anatomy are key issues, for example in the study of brain development and disease progression. In [48] a mean anatomical image is defined using a deformation-based metric via the Fréchet mean. A concrete application regarding Fréchet means is also provided in [43], namely a problem of handwritten signature authentication. The problem consists in developing an automatic procedure that authenticates the identity of an individual using its signature. A signature is seen not only as a drawing, but rather as a procedure, taking into account the hand movement of the individual and its tempo. This improves the authentications since a signature cannot be reproduced using only a written sample. Two signatures of the same person are never identical and thus they are seen as randomly perturbed versions of an ideal average

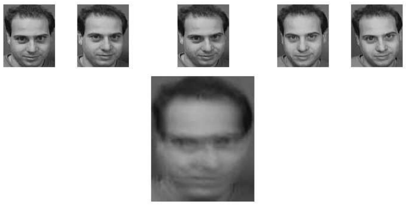


Figure 1.1: Empirical mean of 5 faces, taken from the Olivetti database [115]. We can see that, due to a slight shift in the initial pictures, the average one is all blurred.

one. Taking into account the natural variations that may appear, a set of possible deformations is defined and signatures are now identified on a quotient space, up to a such deformation. The quotient space losses the Euclidean structure and to overcome this issue, a Fréchet mean is used to define a mean signature from a sample provided by the user.

Fréchet mean can also be used to define a barycenter on the Wasserstein space of probabilities measure on $\mathbb{R}^d$ (in [6]) which was later generalized to the notion of regularized barycenter (see for example [27] or [47]). It was also used as a preliminary in defining a principal component analysis on Wasserstein space of real probability measures in [32].

All these fields imply models of data in non-Euclidean metric spaces and thus the Fréchet mean is used to define an average object or position. For this very reason, this particular case is also of special interest on our work.

**Existence and uniqueness** Now that we have seen what Fréchet means are and some of the ways in which they can be used, a natural question that arises is the existence and uniqueness of such means. Without additional assumptions this cannot be guaranteed in the general case of a metric space, since even in the case of distributions on the real line, these properties do not always hold. A simple example for the non-uniqueness is the median of the Bernoulli distribution with parameter $1/2$, for which any $x \in [0, 1]$ is a median. As for the existence part, a classical example of distribution that has no mean, variance or moments of higher order is the Cauchy distribution (of density $f(x) = 1/\pi(1 + x^2)$). The existence and uniqueness of $p$-means for a probability distribution on a Riemann manifold have been the subject of many studies (see for example [5] and [43] for more details on the related literature), but are not of interest in the present work. We will see that in our framework the existence of Fréchet means is immediate and the uniqueness is not a reasonable assumption.

**Algorithms related to Fréchet mean** Besides the existence and uniqueness of such a mean, another important aspect is developing algorithms that can compute it. There are several deterministic algorithms that have already been proposed in a Riemannian setting, like [93] and [70]. Different algorithms have also been developed in the stochastic setting. For example, under conditions ensuring existence and uniqueness of the $p$-mean (provided by Afsari [5]), Arnaudon *et al.* propose in [12] a stochastic gradient descent algorithm that converges almost surely to the $p$-mean (for $p \geq 1$). Other stochastic algorithms have been developed for finding $p$-means on the circle (*e.g.* [15]) or on compact Riemanian manifolds (*e.g.* [14], even for $p > 0$). Moreover, most of the works mentioned in the paragraph related to application fields also propose a method for estimating the Fréchet mean in their specific framework.

**Estimating a Fréchet mean** Many works related to Fréchet means are concerned with the convergence of the M-estimator, $M_{\nu_n}$, towards $M_\nu$ when $\nu_n$ is an empirical measure produced by $n$ i.i.d. samples of $\nu$. In [24], the authors show that on a complete Riemannian manifold, when Fréchet mean exists (uniqueness is not necessary, the mean might be represented by a mean set), almost surely all measurable choices from the sample mean set $M_{\nu_n}$, converge uniformly to the mean $M_\nu$. This is an extension of a previous result of consistency obtained in [124] for a compact manifold. The asymptotic distribution of the sample mean was also studied and central limit theorems are established, for example in [24] and [23]. This kind of results, in particular, implies that if the probability measure

$\nu$ is not explicitly known, the Fréchet mean can be estimated using online arrivals of independent random variables distributed according to it.

However, the consistency of Fréchet mean estimators is not to be taken as granted. For example, in the framework of image analysis, the point that minimizes the empirical variance in the quotient space (the space of equivalence classes of observations under the action of a set of deformations) does not always correspond to the equivalence class of the original mean template. In [50] the authors give sufficient conditions for which inconsistency appears. In [28] the authors use a Fréchet mean of smoothed data to estimate a mean pattern of curves or images satisfying a non-parametric regression model including random deformations and study under which condition the Frécht mean estimator is consistent, namely under which conditions it converges in probability to the true mean pattern. They show that their estimator is consistent when both the number of observed curves (or images) and the number of design points tend to infinity, but the estimator does not converge to the true mean patters if the number of design points is fixed and only the number of images goes to infinity.

## 1.2 Graphs

Graphs model complex structures that are studied in a wide variety of domains and play a central role in many fields. We start by presenting some general facts related to graphs and their applications, some of the main mathematical questions they raise and then introduce some definitions of graph related terminology needed in the sequel.

### 1.2.1 Overview of graphs and their application fields

The beginnings of graph theory go back to the 18*th* century, to the problem of *Seven Bridges of Königsberg*. The city of Königsberg (in Prusia), was crossed by a river that not only split the city in two mainland portions, but also formed two large islands. These 4 portions of the city were connected by 7 bridges. The problem consisted in finding a walk through the city that would cross each bridge exactly one time. In [59], Euler modelled this layout as a graph, with 4 vertices (each vertex corresponds to a part of the city) and 7 edges (each edge represents a bridge), in order to prove that the problem has no solution. This graph is represented in Figure 1.2.

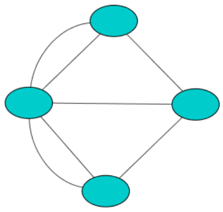


Figure 1.2: The city of Königsberg and its 7 bridges represented as a graph.

**One of the firsts fields of applications: Sociology** One of the oldest application of graphs, that is still very popular today, is in the field of sociology and social psychology. Some interesting examples are the study of human interactions made by Moreno [100] in the 1930′s, or the empirical studies of Milgram in the 1960′s [97] that gave birth to the well known term *Small World Phenomenon* (that refers to the fact that the average path length in a social network is generally small).

**Random graphs** Around the same time (1960) the field of random graphs was instigated by Erdös and Rényi, see [56] and [55]. These seminal works have incited a great amount of works. At first, random graphs were used to prove deterministic properties, these methods being referred to as *probabilistic method*, for details see [10]. The model proposed by Erdös and Rényi is rather simple: we consider a graph with $n$ vertices and place an edge between any pair of distinct vertices with a fixed probability $p$, independently for all pairs. Both the main advantages and disadvantages of this model lie exactly in its simplicity. On one hand, it is easy to study, and thus there are many theoretical results regarding its main properties (connectivity, degree distribution, diameter, *etc.*). On the other hand it is not suited to model real world examples. The analyze of real-world networks showed that they generally have some similar properties like *small world* (typical distances between nodes are small) or *power law degree* (the degrees are highly variable, namely there are a lot of vertices whose degree greatly exceeds the average), both properties that are not fulfilled by the Erdös-Rényi graphs. For a detailed review of real networks and their properties we refer the reader to [34] and [7].

Several random models have been developed, especially after 1999, in order to better represent some specific properties desired in a network like: stochastic block model (the vertices are separated into several groups, each group having its own inner connectivity probability, such that two vertices of the same group are more likely to be connected than nodes from two different ones), configuration models (the degree of each vertex is fixed in advance) or preferential attachment model (vertices are added in a iterative way and at each step the existing vertices with a high degree have a higher probability of connection), and many others. For more details on this type of models see [120].

**Networks nowadays** The increasing amount of available data and computational performances in recent years, boosted the interest on the field of random graphs, but also on the study of networks in general. The range of fields of applications is very wide: computer science (Web understanding [109] and representation), biology (neural or protein networks, genes), social sciences (analysis of citations graphs, social networks [75]), statistical or quantum physics [57], marketing (consumers preference graphs) and computational linguistics (the use of semantic networks in speech recognition software and machine translation programs). We refer to [83] and [69] for additional applications in economics and machine learning. This is not an exhaustive list, but merely some of the main fields of application.

**Mathematical questions raised by graphs** The escalation of the number of fields in which graphs have applications makes it impossible to give a complete or detailed survey on the general domain of networks. However this brief and rather shallow introduction already gave us a glimpse to some of the mathematical questions raised by models involving networks: the study of geometrical or statistical properties, like the degree distribution or typical distances between nodes, and the definition of a suitable random graph model. Other important questions are linked to graph geometry visualization (some popular methods are presented in [117]) or the evolution of a dynamical system over a graph (see [4] and the references therein). In statistics, an important field of interest is clustering. Clustering aims at detecting groups of nodes that have significant common features, and can be a helpful tool in better understanding the mechanisms underlying the graph (*e.g*, [89], [19], [106] or [105]).

In this work we interest ourselves in two statistical objects that are related to the graph's geometry: barycenter and principal components. We aim at defining these objects and then estimating them using random processes that evolve over the graph. We will see more details about this later on (in Chapter 2 and Chapter 4). In what follows we introduce some basic terminology and notions related to graphs.

### 1.2.2 Basic notions

Although, usually graphs are considered as discrete structures, we will see that we can also associate to it a continuous version, by identifying every edge with a real segment. This type of structure, endowed with a differential operator is called quantum graph. For us, the advantage of such a structure is that we can define on it a diffusion process. We start by defining some terminology in the discrete case and then go on to the continuous structure. We will also define the Fréchet mean on graphs (both discrete and continuous). The definition of principal components is postponed to Section 1.4.

#### Discrete graphs

As we have seen in the context of the problem of seven bridges, a graph $G$ can be understood as a collection of vertices (or nodes) $V$, that model objects (or persons, gens, *etc.*) and a collection of edges, $E$, that model interactions between the objects. We usually denote it $G = (V, E)$.

Graphs can be directed or undirected. In a *directed graph* every edge is seen as an ordered pair $(u, v) \in E$, indicating that $u$ is directly connected to $v$. This means that the relations are not mutual. The fact $u$ is directly connected to $v$, does not imply that the reverse is true: $[(u, v) \in E] \nrightarrow [(v, u) \in E]$. A well known example of directed network is the World-Wide-Web, where each vertex is a web page and each edge is a hyperlink from one page to another. A citations network can also be build on the same principle: each vertex is an article and edges represent citations from one article to another.

The graphs we will consider from now on are *undirected.* This means that an edge is an unordered pair $\{u, v\} \in E$, that indicates that the nodes $u, v \in V$ are directly connected. All relations are mutual, if $u$ is directly connected to $v$, then $v$ is also directly connected to $u$. This type of structure can be encountered in the case of a citation network, if vertices represent the authors and edges represent collaborations among them. Another popular example are the social networks, were nodes represent individuals and edges the friendships among them.

Graphs can be *finite* or *infinite*, depending on the cardinality of the vertex set $V$. Since networks are finite, in the present work, we consider only graphs that have a finite number of vertices. We also assume that the graph is *simple*, meaning that there are no self loops or multiple edges. This means there is no edge connecting a vertex $v$ to himself, $\{v, v\} \notin E$, and there is at most one edge between any two different nodes. The graph represented in Figure 1.2 is not a simple graph because it has multiple edges. A simple graph with $N$ vertices, has at most $N(N-1)/2$ edges. When an edge exists between any two nodes, the graph is called *complete.*

Graphs can also be connected or disconnected. We say that two nodes $v$ and $u$ are connected if there exists a path that links them. A *path* between two vertices $u$ and $v$ is a sequence of vertices $(x_i)_{0 \leq i \leq n}$ such that:

$$x_0 = v, \ x_n = u, \qquad \text{and} \qquad \{x_i, x_{i+1}\} \in E, \ 0 \leq i < n.$$

A path $(x_i)_{0\leq i\leq n}$, is also denoted $x_0 \to x_1 \to \ldots \to x_n$. If any two different vertices $u, v \in V$ are connected we say that $G$ is *connected.* Otherwise, we call it *disconnected.* From now on, unless specified otherwise, we will only concern ourselves with connected graphs.

Let $e$ be an edge. If $v$ is one of its extremities ($v \in e$), we say that $v$ *is adjacent to* $e$ and write $v \sim e$. The *degree* of a vertex $v$ represents the number of edges adjacent to $v$:

$$\deg(v) = \sum_{e\in E} \mathbb{1}_{e\sim v}.$$

Another important notion is the one of *weighted* graph, meaning that each edge has an associated positive real value, called a weight (or length). In applications, the weight of an edge can represent the length of a route, the closeness of a friendship in a social network or be the inverse of the number of papers published together by two authors in a citation network.

The structure of a simple weighted graph $G$, can be summarized in an *adjacency matrix* $W = (w_{i,j})_{1\leq i,j\leq N}$, where $N$ is the number of vertices. Supposing that the nodes are labeled from 1 to $N$, each entry $w_{i,j} \geq 0$ represents the weight of the edge $\{i, j\}$, when such an edge exists:

- if $i$ and $j$ are directly connected, $\{i, j\} \in E$, then $0 < w_{i,j} < +\infty$;
- if $i = j$, $w_{i,j} = 0$;
- if $i \neq j$ and there is no direct link between them, $\{i, j\} \notin E$, then $w_{i,j} = +\infty$

If $G$ is undirected, for all $i, j \in V$, $w_{i,j} = w_{j,i}$ and thus $W$ is a symmetric matrix. If the graph is not weighted, generally we associated to it an adjacency matrix $A$, such that $a_{i,j} = 1$ if there is an edge between nodes $i$ and $j$, and 0 otherwise.

A simple graph is naturally endowed with a metric $d : V \times V \to \mathbb{R}_+$. For any two vertices $i, j \in V$, the *distance* $d(i, j)$ is defined as the length of the shortest path between them. The length of this path is given by the addition of the length of traversed edges:

$$\forall (i,j) \in V \times V \qquad d(i,j) = \min_{i=x_1\to x_2\to\ldots\to x_n=j} \sum_{\ell=1}^{k-1} w_{i_\ell,i_{\ell+1}}. \tag{1.1}$$

When the graph is not weighted, the length of the edges is considered constant and equal to 1, and the distance simply corresponds to the number of traversed edges.

To sum up, in this thesis we deal with finite, simple, undirected, connected and weighted graphs, endowed with a distance $d$, as defined in (1.1). Unless mentioned otherwise, a graph $G = (V, E)$ is always supposed to have these properties. Now that we have introduced some graph related terminology and established a framework, we have everything we need to define the barycenter of a graph.

### Fréchet means: Barycenter of a discrete graph

Let $G = (V, E)$ and $\nu$ be a probability measure on the vertex set $V$. This probability distribution is used to measure the influence of each node on the graph. In a citation network this can represent the number of citations of an author or the number of times an article was downloaded. An illustration is provided in Figure 1.3. On a public transport network it can represent the monthly average number of individuals using a particular station. As expected, we call *barycenter* of a graph $G$ with respect to $\nu$, its *Fréchet mean*

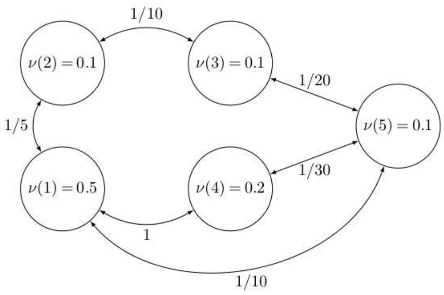


Figure 1.3: Example of a weighted graph between five authors, where the probability mass is $\{0.5, 0.1, 0.1, 0.2, 0.1\}$. In this graph, Author 1 shares five publications with Author 2 and ten publications with Author 5, and so on.

$M_\nu$, defined by:

$$M_\nu := \arg\min_{x \in V} U_\nu(x) \qquad \text{where} \qquad U_\nu(x) = \frac{1}{2} \sum_{y \in V} d^2(x,y)\nu(y). \tag{1.2}$$

Since $G$ is a connected graph that has a finite number of edges the existence of Fréchet means is immediate. At the same time we cannot ensure uniqueness without further assumptions on the graph's structure or on the probability measure $\nu$. An easy way to see this is to consider the cyclic graph with $n$ vertices, all edges of length one and the uniform probability measure on the vertex set. In this case $M_\nu = V$. An illustration of such a graph is presented in Figure 1.4

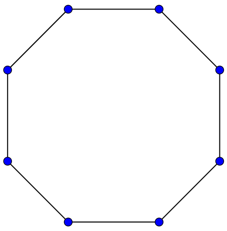


Figure 1.4: Consider the uniform probability on the vertex set and all edges of length 1. For all vertices $U_\nu$ has the same value and thus all vertices are Fréchet means.

### Quantum graphs

To a discrete graph $G$, we associate a continuous version $\Gamma_G$, called *metric graph* that corresponds to the set of points living inside the edges $e \in E$ of the initial graph. This is illustrated in Figure 1.5. Each edge $e$ is now seen as a set homeomorphic to the a real segment. For each edge, we fix a parametrization $x_e : e \to [0, L_e]$, where $L_e$ represents the length of the edge, and $x_e^{-1}(0), x_e^{-1}(L_e) \in V$ are the two vertices adjacent to it.

The continuous graph, $\Gamma_G$, is homeomorphic to the union of segments $[0, L_e]$, for $e \in E$, seen as disjoints sets except for the end points corresponding to the same vertex, which are identified. Metric graphs can also be considered as Riemannian manifolds of dimension one with singularities.

Another remark is that metric graphs are not necessarily defined through a discrete one. A general definition of metric graphs is provided in [20]. However, our definition is not restrictive since to each metric graph we can associate in a natural way several

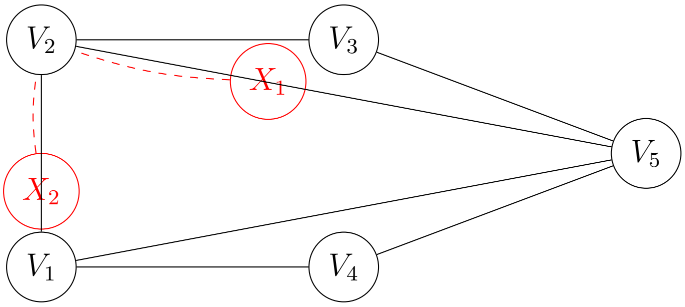


Figure 1.5: An example of a metric graph: $X_1$ and $X_2$ may live inside the edges and not only on the nodes of the graph. Here, $X_1 \in \Gamma_G$ is on $[V_2, V_5]$ and $X_2 \in \Gamma_G$ is on $[V_1, V2]$ but $X_1 \notin G$ and $X_2 \notin G$. The geodesic (shortest) path between $X_1$ and $X_2$ is shown in dashed red line.

discrete weighted graphs $G$.

We extend all notions defined above for the underlying discrete graph $G$ to $\Gamma_G$. From now on, let $\Gamma_G$ be a finite, undirected, simple, connected and weighted metric graph, that has an underlying discrete graph $G$ with all these properties.

The terminology of metric graph comes from the fact that this continuous version is endowed with a natural metric. The distance between two points is the length of the shortest path between them, where lengths are measured along the edges, using the parametrization $x_e$. A *path* going from $x \in \Gamma_G$ to $y \in \Gamma_G$ is now a sequence $x = x_0 \to x_1 \to \ldots \to x_n = y$ such that for $1 \leq i \leq n-1$, $x_i$ are vertices and for all $i < n$, there exists an edge $e_i \in E$ such that $x_i$ and $x_{i+1}$ are on the same edge $e_i$ (the fact that the graph is simple implies that if such an edge exists, it is unique). The distance $d$ is now defined as:

$$\forall (x,y) \in \Gamma_G \times \Gamma_G \qquad d(x,y) = \min_{x=x_0 \to x_1 \to \ldots \to x_n = y} \sum_{i=1}^{n-1} |x_{e_i}(x_i) - x_{e_{i+1}}(x_{i+1})|.$$

*Quantum graphs* are metric graphs equipped with a differential operator acting on functions on the graph. Differential operators are defined in a natural way, using the parametrisation $(x_e)_{e \in E}$. We say that a function $f : e \to \mathbb{R}$ is $\mathcal{C}^k$ on $e$ if and only if $f \circ x_e^{-1}$ is in $\mathcal{C}^k([0, L_e])$. For $f \in \mathcal{C}^1(e)$ we define:

$$\forall x \in e \quad \nabla f(x) = \nabla \left( f \circ x_e^{-1} \right)_{|x_e(x)} \tag{1.3}$$

One can remark that the value of the derivative is influenced by the parametriazation, but only up to a sign. The procedure used in (1.3) can be reiterated to define differential operators of any order. In particular, for all $f \in \mathcal{C}^2(e)$, we denote:

$$\Delta f(x) = \Delta \left( f \circ x_e^{-1} \right)_{|x_e(x)}$$

These two operators are used in Chapter 2 in order to define a diffusion process inside the edges. Now that we have defined differential operators inside edges, we also need to define boundary conditions. There are two classical types of matching conditions. Let $f : \Gamma_G \to \mathbb{R}$. The *Dirichlet boundary conditions* impose that $f$ be null on the boundary of each edge. This can be written as:

$$\forall v \in V \quad f(v) = 0.$$

A function $f$, smooth on each edge ($f \in C^1(e)$ for all $e \in E$), that respects the Dirichlet boundary conditions is continuous over $\Gamma_G$. From now on, if $f$ is continuous over $\Gamma_G$ and $f \in \mathcal{C}^k(e)$ for all $e \in E$ we say that $f$ is $\mathcal{C}^k$ on $\Gamma_G$.

Another type of conditions, that will be used in Chapter 2, are the Neumann boundary conditions that are related to the outgoing (or directional) derivatives at a vertex. To be more exact, for any function $f$ on $\Gamma_G$, at any point $v \in V$, we define the *directional derivative of $f$ with respect to an edge $e \sim v$*, denoted $d_e f(v)$, by:

$$d_e f(v) = \begin{cases} \lim\limits_{h \longrightarrow 0^+} \dfrac{f(x_e^{-1}(h)) - f(x_e^{-1}(0))}{h} & \text{if} \qquad x_e^{-1}(0) = v \\ \\ \lim\limits_{h \longrightarrow 0^-} \dfrac{f(x_e^{-1}(L_e + h)) - f(x_e^{-1}(L_e))}{h} & \text{if} \qquad x_e^{-1}(L_e) = v. \end{cases}$$

If the degree of a vertex $v$ is $n_v$, then we obtain $n_v$ directional derivatives. Directional derivatives could also be defined inside the edges, since any interior point of $e$ can be viewed as a vertex of degree 2. If a function is $\mathcal{C}^1$ on $e$, then its directional derivatives at any point $p$ (inside $e$) would sum to 0.

For a fixed collection of strictly positive constants $\mathcal{A} = \{(a_{e,v})_{e \sim v} \in \mathbb{R}_+^{n_v} | \ \forall v \in V\}$, a function $f$, continuous on the whole graph $\Gamma_G$, respects the *Neumann boundary conditions* if and only if:

$$\forall v \in V \qquad \sum_{e \sim v} a_e d_e f(v, y) = 0. \tag{1.4}$$

A first remark is that the Neumann conditions have the advantage of enabling an integration by parts formula, *i.e.*, for all $f, g \in \mathcal{C}^2(\Gamma_G)$ such that (1.4) holds, we have:

$$\int_{\Gamma_G} f \Delta g = \int_{\Gamma_G} \nabla f \ \nabla g.$$

Besides being convenient for computations, as we will see in Chapter 2, when used in defining a Markov diffusion over the quantum graph, these boundary conditions also have a probabilistic interpretation. When it reaches a vertex, the process is instantly reflected on an adjacent edge $e$ with a probability proportional to $a_e$.

### Fréchet means: Barycenter of a quantum graph

Once more, we consider a probability $\nu$ on the vertex set $V$ and we call barycenter of the quantum graph $\Gamma_G$, with respect to $\nu$, the Fréchet mean:

$$\mathcal{M}_\nu := \arg \min_{x \in \Gamma_G} U_\nu(x) \qquad \text{where} \qquad U_\nu(x) = \frac{1}{2} \int_{\Gamma_G} d^2(x, y) \nu(y). \tag{1.5}$$

The fact that the underlying graph $G$ is simple, finite and connected, implies the that its continuous version is compact. Thus, a Fréchet mean exists. However, it is still not unique. The cyclic graph with $n$ vertices, all edges of length one and the uniform probability measure on the vertex set (Figure 1.4), is still a counter example, since in this case all vertices are barycenters. This is not really surprising. Since we cannot guaranty the uniqueness of geodesic paths form one point to another, $U_\nu$ is not necessarily convex and thus, in order to recover the uniqueness, we would need to make additional assumptions on the graph's structure and the probability measure $\nu$. This question was already studied in other contexts, for example necessary and sufficient conditions for the existence of a Fréchet mean on the circle are provided in [44].

A natural question that arises now, is the link between the barycenter of the discrete graph $G$ and the one of its continuous version $\Gamma_G$. One can show that in some special configurations of the graph (for example a square with edges of lengths 1), independently on the probability measure $\nu$, every element of $\mathcal{M}_\nu$ (barycenter of the quantum graph) is on the same edge as an element of $M_\nu$ (barycenter of the discrete one). This however does not hold in general.

## 1.3 Simulated annealing and optimization under uncertainty

The main topics addressed in this thesis involve a global optimization under uncertainty problem (see chapters 2, 4, 5). The motivations, the state space and the applied methods may vary from chapter to chapter, but in each case, the general framework can be summarized in the following way.

We are given two state spaces $\mathbf{E}$ and $\mathbf{F}$ and a probability measure $\nu$ over $\mathbf{F}$. We are interested in minimizing a function $U_\nu : \mathbf{E} \to \mathbb{R}_+$, of the form:

$$U_\nu(x) = \mathbb{E}_{Y\sim\nu}[U(x,Y)],$$

where $U$ is a positive bounded cost function and $Y$ is a random variable on $\mathbf{F}$ distributed according to $\nu$. We suppose that the probability measure $\nu$ is known only through a sequence of independent random variables $(Y_n)_{n\geq 0}$ distributed according to it. Our problem is thus two fold: firstly, we want to find the global minimum of $U_\nu$, secondly we are not able to access its exact value. To address the first part, we start by presenting a classical global optimization method, that plays a significant role in the present work: simulated annealing. For the second part of the problem, we turn to the field of stochastic algorithms. Finally, we present a way of merging these two concepts in order to give a possible answer to our problem.

### Simulated annealing

Simulated annealing is a probabilistic method that aims at finding a global minimum of a given function $U$ over a state space $\mathbf{E}$. Its name comes from metallurgy, where the annealing procedure consists in heating metals (increasing thus the diffusion rate of atoms inside the material) and then cooling them in a controlled manner (letting the material reach its equilibrium state) in order to reduce their defects.

This method was introduced by Kirkpatrick, Gelatt and Vecchi in 1983 [88]. Several convergence proofs were given shortly after. A rather well known one is provided by Hajek in 1988 [73]. The same year, another interesting proof of convergence using modern semigroup representation of Markov processes has been obtained by Holley and Stroock in [77].

#### How does it work?

The main idea of the simulated annealing consists in constructing a Markov process that converges to the quantity of interest, a global minimum of $U$. In other words, its aim is to sample uniformly from $S_{opt}$, the set of global minima of $U$:

$$\mu^\star = \frac{\mathbb{1}_{S_{opt}}}{|S_{opt}|},$$

where $|.|$ denotes the cardinality of a set. At first sight, this is in no way easier than the initial problem. However, the main idea of the Laplace method could encourage us to rewrite this probability distribution using Gibbs measures associated to the potential (or energy) $U$ as the following limit:

$$\forall x \in \mathbf{E},\ \mu^\star(x) = \lim_{\beta \to +\infty} \frac{e^{-\beta U(x)}}{Z_\beta}\ , \tag{1.6}$$

where $Z_\beta = \int_{\mathbf{E}} e^{-\beta U(y)} \mathrm{d}y$ is a normalizing constant. The constant $1/\beta$ is called temperature.

The next step is to introduce an inhomogeneous Markov process on $E$, such that at each time $t$, the invariant measure of the corresponding instantaneous time homogeneous Markov process, is the Gibbs measure associated to $U$ with temperature $1/\beta_t$. By setting the inverse temperature such that $\beta_t \to +\infty$ as $t$ goes to infinity, one might hope to force the Markov process to concentrate on $S_{opt}$, the optimal set.

It is now obvious that there are two competing goals: the convergence of the Markov process toward its invariant measure and the concentration of Gibbs measures around the set of minimums. This is why calibrating the temperature schedule is crucial: if it decreases too fast, the process might not have the time to approach its equilibrium. On the other hand, we want it to decrease as fast as possible in order to draw the Gibbs measure to the set of minima (1.6). Hajek in [73] gave a necessary and sufficient condition for the convergence in probability of the process towards the optimal set, specifically that the inverse temperature schedule needs to be of the form:

$$\beta_t = b \log t \quad \text{with} \quad 0 < b < 1/c^\star(U),$$

where $c^\star(U)$ is a constant depending on some properties of $U$. To be more explicit, it represents the minimal energy barrier that one needs to cross, when going from any local minimum to a global one. This constant is rather important as it sets a limit to the convergence speed, but in practice it is generally not known in advance and quite difficult to estimate. More details about it are presented in Chapter 2 (Section 2.2.3) and Chapter 5.

Simulated annealing methods can be separated in two large classes depending on the nature of the used Markov process. One approach, sometimes referred to as Metropolis Hastings method, consists in building a random walk (mostly when $\mathbf{E}$ is a discrete space). The other one, also called Langevin simulated annealing, generates a drifted diffusion process (generally when $\mathbf{E}$ is a continuous space).

### Continuous setting.

In a continuous setting, the Markov dynamic of the simulated annealing is given by:

$$\forall t \geq 0 \qquad dX_t = -\beta_t \nabla U(X_t) dt + dB_t, \tag{1.7}$$

where $B_t$ is a standard Brownian motion and the drift coefficient $\beta_t$ is the inverse of the temperature schedule. In an equivalent manner, one could choose to control the variance of the Brownian motion instead of the drift coefficient, set $\epsilon_t = 1/\sqrt{\beta_t}$ and write: $dX_t = -\nabla U(X_t)dt + \epsilon_t dB_t$.

Equation (1.7) indicates that the simulated annealing procedure is a modified version of the classical gradient descent (that would correspond to a temperature $\epsilon_t$ equal to 0). The added randomized moves are what help the process escape local traps, turning it into a global optimizer. The convergence of this dynamic to the optimal set was proved, under some regularity assumptions on the potential $U$ and for $(\beta_t)_{t\geq 0}$ increasing slow enough, for example in [96].

### Discrete setting.

The state space $\mathbf{E}$ is usually considered finite and its exploration is done through a random walk $(X_k)_{k\geq 0}$, or eventually a continuous time one, $(X_t)_{t\geq 0}$.

In a few words, at each step, the random walk chooses a potential new state $x'$ according to a proposition law $L(X_k,\cdot)$. If the value of the potential $U$ in $x'$ is lower, the move is accepted with probability one. Otherwise the move is accepted with a probability that depends on the temperature schedule:

$$X_{k+1} = \begin{cases} x' \text{ with probability } p_k, \\ X_k \text{ with probability } 1-p_k, \end{cases} \tag{1.8}$$

where the probabilities $p_k$ are defined as:

$$p_k = 1 \wedge \left\{ e^{\beta_k [U(X_k)-U(x')]} \right\}, \tag{1.9}$$

and $(\beta_k)_{k\geq 0}$ is of course the inverse of the temperature schedule.
Once more, we see that small values of $\beta_k$ encourage exploratory moves and high ones penalize them. The convergence of such random walks to the set of minima of $U$, when $L$ is reversible and $(\beta_k)_{k\geq 0}$ goes to infinity slow enough, is a well known result (see for example [73]).

In our framework, the function we are trying to minimize cannot be directly accessed, but only through unbiased estimators. As we have seen previously, classical simulated annealing techniques need precise evaluations of the gradient $\nabla U$, in the continuous setting, and of $U$ itself, in the discrete one. A natural question that arises is if (and how) can they be adapted to overcome this type of problems. In order to better understand how this can be done, we turn to the field of optimization under uncertainty.

### Optimization under uncertainty

There are a lot of real-world examples that imply the optimization (or study of other properties) of functions that have a partially random input. This occurs, among others, in the area of on-line procedures or real-time estimation and control. Consider for example the problem of optimizing commercial aircraft trajectories: the cost, in terms of fuel consumption and flight duration, of a given trajectory is not known in advance and its estimation relies on some predicted flight conditions including atmospheric ones. Hence, real-flight costs can deviate substantially from their predictions and some uncertainty propagation method must be applied to obtain an accurate estimate of the expected flight costs. An application of stochastic optimization for Aircraft trajectories is presented in Chapter 5.

The field of stochastic algorithms was developed in answer to such problems. In 1951 Robbins and Monro [113] introduced one of the first stochastic algorithms, now known as Robbins–Monro algorithm. They proposed a method to find a root of a function that cannot be directly evaluated, but can be written as an expected value of observable random variables. Assuming that the solution is unique, their algorithm solves the following equation:

$$M(x) = \alpha \quad \text{where} \quad M(x) = \mathbb{E}[m(x,Y)]. \tag{1.10}$$

For a well chosen sequence of positive constants $(a_n)_{n\geq 0}$, they define the following Markov Chain:

$$X_{n+1} - X_n = a_n(\alpha - m(X_n, Y_n)),$$

and then prove its convergence in probability, to the unique solution of Equation (1.10), independently of the starting point $X_0$. A sufficient set of conditions for the stepping sequence $a_n$, that is often cited, is:

$$\sum_{n=1}^{+\infty} a_n = +\infty \quad \text{and} \quad \sum_{n=1}^{+\infty} a_n^2 < +\infty.$$

This algorithm can be used as an optimization technique by replacing $M$ with the gradient of $U_\nu$, the function we want to minimize, and setting $\alpha = 0$. We obtain thus the stochastic gradient descent algorithm, that can be written as:

$$X_{n+1} = X_n - \gamma_n \nabla U(X_n, Y_{n+1}), \tag{1.11}$$

where $(Y_n)_{n\geq 0}$ is a sequence of independent variables distributed according to $\nu$. Although its name comes from its resemblance with the classical gradient descent given by: $x_{n+1} = x_n - \gamma_n \nabla U_\nu(x_n)$, the randomness renders the two dynamics quite different. It is well known that when $\nabla U_\nu = 0$ has a unique solution, the algorithm (1.11) converges almost surely to it. Even if the function $U$ has more than one critical point, under some regularity assumptions on the function $U_\nu$ and some additional properties on the step sequence $(a_n)_{n\geq 0}$, it is shown in [37] that the algorithm converges almost surely to a local minimum. However, as soon as $U_\nu$ has a local minimum, this is not a global optimization method. We are thus encouraged to mix it with simulated annealing to help it escape local traps. Once more, we treat separately the two different setting.

### Continuous Setting: Homogenization Technique

In our framework, the classical formulation (1.7) of simulated annealing can be written as:

$$\forall t \geq 0 \qquad dX_t = -\beta_t \nabla \mathbb{E}_{Y\sim\nu}[U(X_t, Y)]dt + dB_t. \tag{1.12}$$

The homogenization technique uses the same idea as the stochastic gradient descent, namely it replaces the true value of the gradient in the classical formulation by realizations of independent random variables.

Suppose we can access a sequence $(Y_n)_{n\geq 0}$ of independent random variables distributed according to $\nu$ and that the gradient and the expectancy in Equation (1.12) commute. Consider $(\alpha_t)_{t\geq 0}$, a positive, continuous, increasing function such that $\lim_{t\to+\infty} \alpha_t = +\infty$. We define an accelerated Poisson process $(N_t^\alpha)_{t\geq 0}$ of intensity $\alpha$, that will translate at what rate we use the sequence of random variables. It is standard to represent $N^\alpha$ through a homogeneous Poisson process $H$ of intensity 1 using the relationship:

$$\forall t \geq 0 \qquad N_t^\alpha = H_{h(t)}, \text{ where } h(t) = \int_0^t \alpha_s \mathrm{d}s.$$

The homogenized simulated annealing can now be described as:

$$dX_t = -\beta_t \nabla U_{Y_{N_t^\alpha}}(X_t)dt + dB_t.$$

This type of algorithm was used for example in [13], for finding Fréchet means on a compact Riemannian manifold, or for generalized means in [14].

To ensure the convergence in probability of the process $(X_t)_{t\geq 0}$, one needs to find a balance between the decrease of temperature given by $\beta_t$ and $\alpha_t$, the rate at which the observations are used. Intuitively, $\alpha_t$ needs to increase rapidly in order to provide a fine estimation of the gradient. However, in practice, one would like to limit the number of used observations, and thus to choose a slowly increasing function. For a logarithmic inverse temperature schedule, in general, setting $\alpha_t$ as a linear function of time, is sufficient to ensure the convergence of the process (under regularity assumptions on $U$ and $\nu$).

### Discrete setting: Monte Carlo methods

In the discrete setting, the main idea is to replace, in the computation of the probability transition (1.9), the true value of the function $U_\nu$ by an unbiased estimator $\widetilde{U}_\nu$, and expect that the new (also called noisy) Markov chain stays "close" to its corresponding classical Metropolis Hasting Markov chain.

Gelfand and Mitter, [67], were probably the first ones to introduce the concept of simulated annealing with noisy measurements. They assumed an additive Gaussian noise independent of the evaluation point and gave a sufficient condition for the decrease of the variance $\sigma_k^2$ of this noise, to ensure convergence of the algorithm to the optimal set. This result was extended in [72] to distributions that are more peaked around zero than the Gaussian distribution. In the framework of Gaussian noise, another version of the algorithm was proposed in [8], where the authors modify the transition probability using confidence intervals. They show that in practice, their method performs better than the one of [72].

In our framework, a natural approach would be to replace the true values of $U$ in the computation of the transition probability by Monte Carlo estimations:

$$\widehat{U}_{\nu,N}(x) := \frac{1}{N}\sum_{i=1}^{N} U(x, Y_i),$$

for any given i.i.d. sample $(Y_i)_{1\leq i\leq N}$. An interesting question that arises at this step, is how to choose $N$, or in other words how precise does this estimation need to be in order to assure convergence towards the optimal set. For example the condition imposed on the variance of the noise by Gutjahr and Pflug [72], roughly corresponds to the use of $N = k^\alpha$ observations at each iteration (also called mini-batches), with $\alpha > 2$. In Chapter 5 we show that by choosing a slower temperature schedule, we can obtain the convergence even for $\alpha < 2$. However, by choosing a smaller $\beta_t$, we also decrease the algorithm's convergence speed. Thus, this strategy is not necessarily optimal. More details can be found in Chapter 5.

The Noisy Metropolis Hastings algorithm has received a lot of attention in the field of Bayesian statistics, since it can also be used to sample from a given distribution. For example, by replacing in the definition of the probability transition of noisy-less Metropolis Hasting (1.9) the sequence of decreasing temperatures by a fixed $\beta$, the new Markov chain will converge to the corresponding Gibbs measure $\mu_\beta$. Numerous variants of Noisy Metropolis Hastings were developed, proposing different types of estimators for the value of the probability transition. See for example [11]. In this setting also, the size of mini-batches used at each iteration is of paramount importance since is directly related to the computational cost. In [53] the authors present a guideline on how to choose $N$ depending on the standard deviation of the log-likelihood estimator. The strategy we use in the convergence proof is based on the study of the process' distribution and thus could be adapted to problems of law sampling.

## 1.4 Principal Component Analysis

Principal component analysis (PCA) is a multivariate statistical analysis method. It was first formulated by Pearson [110] in 1901 as a way to "represent a system of points in plane, three or higher dimensional space, by the "best fitting" straight line or plane." The term of principal component was introduced later on, in 1933, by Hotelling [80], who also formalized the concept in its modern form. For a more detailed historical review of the development of PCA we refer to [85].

### What is PCA

The main idea of PCA is to synthesize a data set consisting in a large number of correlated variables, while retaining as much as possible the variation present in it. To achieve this, PCA computes new uncorrelated variables, called principal components. In an Euclidean space, the new variables are obtained as linear combinations of the original ones. The first principal component is chosen as the linear combination that retains most of the variance present in all of the original variables. The second principal component is computed again in order to retain most of the variation, but under the additional constraint of being orthogonal to the first one. The other components can then be computed using this procedure iteratively (see for example [3]).

In an Euclidean framework, PCA can be reduced to the study of eigenvalues and eigenvectors of a positive-semidefinite symmetric matrix, and is generally treated in this manner. However, principal components can also be described as a sequence of nested affine sub-spaces of increasing dimension that maximize the variance of the projections or minimize the sum of norms of projection residuals. For example, consider the case of $m$ points $x_1, \dots x_m$, in $\mathbb{R}^n$ and for all $v \in \mathbb{R}^n$ denote:

$$S_v = \{\bar{x}_m + tv, t \in \mathbb{R}\}, \text{ where } \bar{x}_m = \frac{1}{m}\sum_{i=1}^{m} x_i \text{ is the Euclidean mean.} \tag{1.13}$$

A first principal component of the data can be described as $S_{v_1}$, where $v_1$ is such that:

$$v_1 \in \underset{\|v\|=1}{\arg\min} \sum_{i=1}^{m} d^2(x_i, S_v).$$

This last perspective on PCA, as we will see later on, is at the root of most generalizations to non-Euclidean frameworks.

### Applications

PCA is used in a wide range of scientific fields, sometimes under different names, like electrical engineering (Karhunen-Loéve expansion), chemistry (principal factor analysis) or image analysis (Hotelling transformation). For more details on PCA's fields of application see for example [122].

In what follows, we present some of the various ways in which PCA can be used. More details about these applications and a rather complete survey is given by Joliffe in [85]. If most of the variation of the data can be reproduced by a small number of uncorrelated variables that have a natural interpretation, then the principal components give a much simpler description of the data than the original variables. However, if a natural interpretation does not exist, and one prefers to work with subsets of the original variables, PCA can also be used to suggest suitable subsets. PCA is sometimes employed as a preliminary or in conjunction with other statistical techniques like regression, cluster analysis or discriminant analysis. Even though, generally, studies are interested in a few first principal components, it has been suggested that the few last ones can be useful in uncovering outliers in a data set (see for example Chapter 10 of [85]).

### Extensions of PCA

As we have seen before, nowadays, we often deal with metric spaces that are not necessarily Euclidean. PCA was thus generalized to more general metric frameworks. For example, functional principal component analysis (FPCA) was developed to deal with data where a whole function corresponds to an observation. FPCA was used in analyzing the main modes of variability of a set of probability densities in [49]. In [32], the authors highlight some of the drawbacks of the FPCA and suggest that they are due to the fact that the Euclidean distance in $L^2(\mathbb{R})$ is not always appropriate to perform PCA on density functions. They introduce a notion of geodesic principal component analysis (GPCA), by relying on the formal Riemannian structure of the Wasserstein space of probability measures over $\mathbb{R}$. They also introduce a strategy of convex constrained PCA, in a general Hilbert space.

A survey of methods employed in the Riemannian setting can be found in [81]. A method of particular interest for us is GPCA, that can be seen as PCA based on minimization of intrinsic residual distances to geodesics and where principal components are minimizing geodesics of GPCA. Another one is manifold PCA, which is described as PCA based on *non-nested* geodesic submanifolds of increasing dimension, determined by minimizing intrinsic residual distances. In the present work, we interest ourselves in graphs and in what follows we present a definition of PCA on metric graphs.

## 1.4.1 PCA on graphs

The extension of PCA on metric graphs provided in this thesis is based on a variational formulation (1.13) and is similar to the strategies of manifold PCA and GPCA.

In what follows, we define principal components on a metric graph $\Gamma_G$, endowed with a probability measure $\nu$ on its vertex set $V$. In order to highlight the main ideas, we will sometimes favor an informal description of the objects involved and postpone a formal definition to Chapter 4.

### Variational formulation

Keeping in mind that principal components minimize the sum of the norms of the projection residuals, a good starting point seems to be to look for subsets of $\Gamma_G$ that minimize the following functional:

$$U_\nu(C) = \mathbb{E}[d^2(X, C)] = \sum_{i=1}^{n} d^2(x_i, C)\nu(x_i), \tag{1.14}$$

where $d(x, C)$ is defined in a natural way as:

$$d(x, C) = \inf_{y \in C} d(x, y).$$

Of course, we still need to establish on what subset of the power set of $\Gamma_G$ we want to minimize (1.14). As mentioned before, we take our inspiration from previous methods proposed for Riemannian manifold, and thus we minimize over subsets of $\widetilde{\mathcal{G}}$, the space of the graphs' geodesics. A subset $\gamma \subset \Gamma_G$ is called a *geodesic* if and only if for any two points $x, y \in \gamma$, there exists a shortest path $\gamma_{x,y}$ connecting the two, included in $\gamma$.

### Relative degree: "dimension" of a geodesic in a metric graph

In order to define principal components, we also need an equivalent to the notion of dimension in $\Gamma_G$. This is provided by the notion of relative degree. If $A$ is a connected subset of $\Gamma$ and $x$ a fixed point of $\Gamma$, we call *the degree of $x$ relative to $A$*, the number of connected components in which the set $A$ is split, only into an arbitrarily small neighborhood of $x$, when we remove $x$ from it:

$$\deg^A(x) := \inf_{\epsilon>0} N_c((A \setminus \{x\}) \cap B_\epsilon(x))$$

where $N_c$ represents the number of connected components. This can also be seen as the number of directions on which one can go from $x$ while staying in $A$. A representation is given in Figure 1.6.

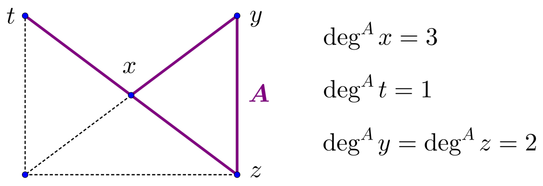


Figure 1.6: Examples of relative degrees.

We denote $\widetilde{\mathcal{G}}_d$ the subsets of geodesics that contain only points whose relative degree is at most $d+1$:

$$\widetilde{\mathcal{G}}_d := \{\gamma \in \widetilde{\mathcal{G}} \text{ s.t. } \forall x \in \gamma, \deg^\gamma(x) \leq d+1\}.$$

It is rather natural to interpret the relative degree as a notion of dimension, since any element of $\widetilde{\mathcal{G}}_d$ is locally isomorphic to a subset of $\mathbb{R}^d$. For example $\widetilde{\mathcal{G}}_0 = \{\{x\}; x \in \Gamma\}$ is the set of all singletons of the quantum graph and $\widetilde{\mathcal{G}}_1$ is the set of geodesic paths and geodesic cycles. An example of such elements is illustrated in Figure 1.7.

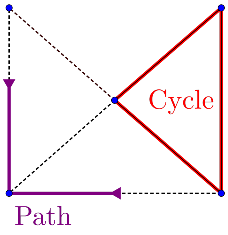


Figure 1.7: Examples of elements of $\mathcal{G}_1$.

### Definition of principal components

We can finally define a *principal component of order $d$*, as an element $g_d^* \in \widetilde{\mathcal{G}}_d$ such that:

$$g_d^* \in \arg\min_{g \in \widetilde{\mathcal{G}}_d} U_\nu(g). \tag{1.15}$$

We denote $\mathcal{M}_\nu^d$ the set of minimizers of $U_\nu$ over $\widetilde{\mathcal{G}}_d$. A principal component of order $d$ always exists, since $\widetilde{\mathcal{G}}_d$ are compact and $U_\nu$ is continuous, but is not necessarily unique (see Figure 1.9). A first observation is that by minimizing $U_\nu$ over $\widetilde{\mathcal{G}}_0$, one retrieves the set $\mathcal{M}_\nu$, representing the classical Fréchet mean.

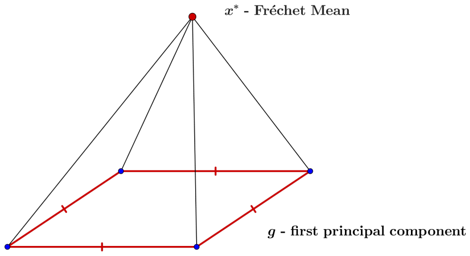


Figure 1.8: Consider the uniform probability on the vertex set and all edges of length 1. The top of the pyramid is the Fréchet mean and the edges forming the base are the first principal component.

A second observation is that as opposed to the Euclidean framework, the principal components are not necessarily nested. In Figure 1.8, we illustrate an example of graph where the Fréchet mean is not included in the first principal component. In order to retrieve this property, we introduce a second notion, of nested-principal components. We present here the definition of a nested-principal component of order one. This definition is extended in an iterative way for all orders in Chapter 4.

In all generality, for $x \in \Gamma_G$, we call *$x$-principal component of order* 1, a set $\mathfrak{g}_1^* \in \widetilde{\mathcal{G}}_1^x$ such that:

$$\mathfrak{g}_1^* \in \underset{g \in \widetilde{\mathcal{G}}_1^x}{\arg\min}\, U_\nu(g).$$

A *nested principal component of order* 1 is a $x$-principal component, when $x$ is a Fréchet mean. Such an element always exists, but is not necessarily unique (see Figure 1.9).

The two types of principal components defined above, have slightly different properties and could be of interest in different applications.

### An illustration of first principal components

To conclude this section, we present an illustration of first principal components, namely of order one, on a Facebook sub-graph in Figure 1.9. In this case, the nested principal components of order 1 coincide with the principal components.

The graph represented in Figure 1.9 is composed of two star-shaped clusters. The central nodes of these clusters are connected by 4 shortest paths, represented in red. We can interpret it as the set of friends of two persons that have only four common friends. Adding a vertex at each extremity of a path connecting the two (in a way that preserves the geodesic property) gives a (nested) principal component. Even though the extreme vertices are not important, we can see that a first principal component includes the center of each community and an individual that connects them, summarizing thus rather well the graph's structure.

## 1.5 Main tools

In this section we define the main mathematical tools used throughout the thesis. Every optimization problem approached in this work is formulated as the study of an evolution of a Markov process. Thus we have chosen to start this section by presenting some brief reminders on this category of stochastic processes.

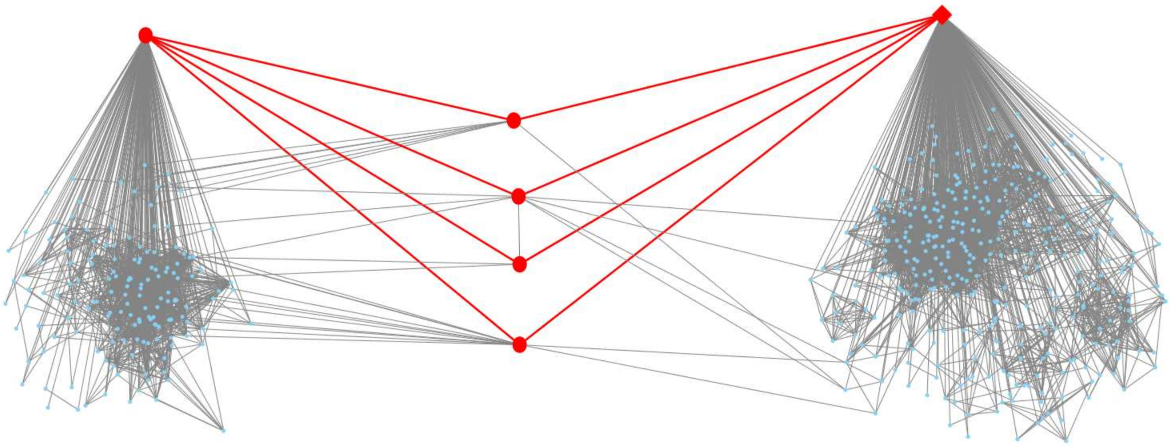


Figure 1.9: Consider the uniform probability on the vertex set and all edges of length 1. The diamond shaped node represents the unique Fréchet mean and the red paths represent the "core" of (nested) principal components.

### 1.5.1 Markov Processes

Stochastic processes model systems that evolve in time according to a random mechanism. Markov processes are one of their most popular class. Roughly speaking, they can be described as stochastic processes whose future is independent of its past. We present only some basic definitions and properties without including any proofs. For more details see for example [58].

**General definition.** Let $\mathbf{E}$ be a complete, metric, separable space (Polish space), equipped with its Borel $\sigma$-field $\mathscr{E}$. A *Markov process* $X = (X_t)_{t\geq 0}$ is a family of random variables constructed on some probability space $(\Omega, \mathcal{F}, \mathbb{P})$ with values on $(\mathbf{E}, \mathscr{E})$ such that for all bounded measurable functions $f$:

$$\mathbb{E}[f(X_t)|\mathcal{F}_s^X] = \mathbb{E}[f(X_t)|X_s], \quad 0 \leq s \leq t, \tag{1.16}$$

where $\mathcal{F}_s^X$ is the filtration naturally associated to the process:

$$\mathcal{F}_s^X = \sigma(X_u, 0 \leq u \leq s).$$

**Semigroups.** The *semigroup* of a Markov process is defined on the set of measurable bounded functions by:

$$P_{s,t}f(x) = \mathbb{E}[f(X_t)|X_s = x]. \quad \forall 0 \leq s \leq t.$$

**Infinitesimal generators.** Given a semigroup, we can define an associated family $(\mathcal{L}_t)_{t\geq 0}$ of *infinitesimal generators*:

$$\mathcal{L}_t f(x) = \lim_{s\to 0^+} \frac{P_{t,t+s}f(x) - P_{t,t}f(x)}{s} \quad \forall\ f \in \mathcal{D}(\mathcal{L}_t),$$

where $\mathcal{D}(\mathcal{L}_t)$ is defined as the set of measurable bounded functions for which this limit exists. It referred to as the generator's *domain*. We denote $\mathcal{D}(\mathcal{L})$ the set of functions $f$ that are in the domain at all times, namely:

$$\mathcal{D}(\mathcal{L}) = \{f \mid t \geq 0\ f \in \mathcal{D}(\mathcal{L}_t)\}.$$

**Kolmogorov equations.** Let $f$ be a function in $\mathcal{D}(\mathcal{L})$. Then the following *Kolmogorov equations* hold:

$$\partial_t P_{s,t} f = \mathcal{L}_s P_{s,t} f = P_{s,t} \mathcal{L}_s f \qquad 0 \le s \le t.$$

In practice, there are situations when an analytic expression of the semigroup is not available, but an explicit form of the generator is given. Thus, the Kolmogorov equations can be useful when one needs to deal with the derivative of the semigroup.

**Homogeneity.** If the semigroup $(P_{s,t})_{0 \le s \le t}$ depends only on the difference $t - s$, then we say that $X$ is a *homogeneous* Markov process and *inhomogeneous* otherwise. The semigroup of a homogeneous Markov process is simply denoted $(P_t)_{t \ge 0}$.

In this thesis we deal mostly with inhomogeneous Markov processes. However, to each inhomogeneous Markov process $X$, we can associate a homogeneous one $\widetilde{X}$, by setting:

$$\widetilde{X}_t = (t, X_t).$$

The infinitesimal generator $\widetilde{\mathcal{L}}$, associated to $\widetilde{X}$ can be defined on the set of functions $\mathcal{D}(\widetilde{\mathcal{L}})$:

$$\mathcal{D}(\widetilde{\mathcal{L}}) = \Big\{ f \mid f : \mathbb{R}_+ \times \mathbf{E} \to \mathbb{R},\ \forall t \ge 0,\ \ f(t, \cdot) \in \mathcal{D}(\mathcal{L}),\ \forall x \in \mathbf{E},\ \ f(\cdot, x) \in \mathcal{C}^1 \Big\}$$

and its action is given by:

$$\widetilde{\mathcal{L}} f(t,x) = \partial_t f(t,x) + \mathcal{L}_t f_t(x),$$

where for all $t$, $f_t : \mathbf{E} \to \mathbb{R}$ is defined by $f_t(\cdot) = f(t, \cdot)$.

This observation allows us to transfer existing results from the homogeneous case to the inhomogeneous one. This can be of interest because the first case was far more analyzed than the latter.

**Invariant and reversible measure.** We say that $\mu$ is an *invariant measure* for $(P_t)_{t \ge 0}$ if for all measurable bounded functions and for all $t \ge 0$:

$$\int_{\mathbf{E}} P_t f \mathrm{d}\mu = \int_{\mathbf{E}} f \mathrm{d}\mu$$

The measure $\mu$ is called *reversible* for the semi-group $(P)_{t \ge 0}$, if for all $f, g \in L^2(\mu)$, and all $t \ge 0$:

$$\int_{\mathbf{E}} g P_t f \mathrm{d}\mu = \int_{\mathbf{E}} f P_t g \mathrm{d}\mu \tag{1.17}$$

These properties can be translated in terms of infinitesimal generators in the following way: $\mu$ is an invariant measure for $\mathcal{L}$, if for all $f \in \mathcal{D}(\mathcal{L})$:

$$\int_{\mathbf{E}} \mathcal{L} f \mathrm{d}\mu = 0,$$

and the fact that $\mu$ is reversible implies that for all $f, g \in \mathcal{D}(\mathcal{L})$ for which Equation (1.17) holds, we have :

$$\int_{\mathbf{E}} f \mathcal{L} g \ \mathrm{d}\mu = \int_{\mathbf{E}} g \mathcal{L} f \ \mathrm{d}\mu$$

### Martingale Problem approach

Let $\mathcal{L}$ be the generator of a homogeneous Markov process. For all $f \in \mathcal{D}(\mathcal{L})$, we denote $M^f = \left(M_t^f\right)_{t\geq 0}$, the real valued stochastic process given by:

$$M_t^f = f(X_t) - f(X_0) - \int_0^t \mathcal{L}f(X_u)\mathrm{d}u \tag{1.18}$$

For any initial law $m_0$ of the Markov process $(X_t)_{t\geq 0}$, $M^f$ is a martingale (with respect to the filtration $\mathcal{F}_t$). This observation leads us to a different approach, namely trying to characterize a Markov process associated to a given generator using the so called martingale problem.

In what follows we present a somehow simplified version of this approach in order to highlight its ideas. A general and detailed description is provided by Ethier and Kurtz in Chapters 3 and 4 of [58].

**Well posed martingale problem: definition.** Let $\mathcal{L}$ be a linear operator and $\mathcal{D}(\mathcal{L})$ its domain. We say that a measurable stochastic process $X$ with values in $\mathbf{E}$ is a *solution of the martingale problem associated to* $(\mathcal{L}, \mathcal{D}(\mathcal{L}), \mu)$ if $X$ has $\mu$ as initial law and for all $f \in \mathcal{D}(\mathcal{L})$, $M^f$ defined as in (1.18) is a martingale (with respect to the associated filtration $\mathcal{F}_t^X$).

If this is true for any initial probability measure, we simply say that $X$ is a solution of the martingale problem $(\mathcal{L}, \mathcal{D}(\mathcal{L}))$. A martingale problem is *well posed* if it has a unique solution (in law).

**Existence and uniqueness.** A solution of the martingale problem can be obtained as a weak limit of solutions of approximating martingale problems (for example a diffusion process can be approached by discrete processes, such an example is given in the introduction of [58]).

The existence and uniqueness (in law) of a solution can be proved using rather technical analytical properties of the generator that roughly demand two things:

Dis For $\mathcal{L}$ to be dissipative:

$$\|f - \lambda\mathcal{L}f\| \geq \|f\|, \qquad \forall\lambda > 0 \text{ and } f \in \mathcal{D}(\mathcal{L}).$$

Den For its domain $\mathcal{D}(\mathcal{L})$ to contains enough functions (for example a dense subset of the set of continuous bounded functions).

These conditions are similar to the conditions of the Hille-Yoshida theorem that lies behind the existence of a contracting semigroup associated to a generator. In [58] the authors show, that under the conditions mentioned above (Dis and Den), if $X$ is a solution to the martingale problem $(\mathcal{L}, \mathcal{D}(\mathcal{L}))$, then $X$ is the Markov process corresponding to the infinitesimal generator $\mathcal{L}$ (see Theorem 4.1 of Chapter 4 in [58]).

This essentially means that under suitable conditions, a Markov process is the unique solution of the martingale problem for its generator. In particular, if the martingale problem is well posed, its solution is a Markov process.

For example, this kind of approach was used by Freidlin and Wentzell in [64] to prove the existence of a Feller Markov process corresponding to a diffusion generator on a quantum graph.

### 1.5.2 Functional inequalities

Once that an optimization problem is brought to the Markov setting, as we have seen in Section 1.3 (and will see later on in Chapters 2, 4 and 5), it can be resumed to a convergence study. To accomplish that, we try to quantify the gap between the law of the process at a given time and a target distribution. In this thesis we measure the "closeness" using two classical tools: the $\chi^2$-distance and the relative entropy.

Once we establish the quantity of interest (*i.e.* choose a distance), we compute its derivative in order to obtain a differential inequality and to conclude using the Grönwall lemma.

In doing so, depending on the chosen instrument, we will be obliged to deal with some terms using functional inequalities: either the Poincaré inequality (for the $\chi^2$ distances) or the Sobolev one (for the relative entropy). As we will see later on, functional inequalities describe an exponential convergence to equilibrium for the semigroup associated to a Markov process. In what follows, we present their definitions and some properties without including any proofs. The vast subject of functional inequalities and their link with Markov processes is thoroughly presented in [21].

First we need to set a framework and define some auxiliary objects.

**Dirichlet form.** Let $X$ be a Markov process with infinitesimal generator $\mathcal{L}$ and unique invariant measure $\mu$. Suppose that $\mu$ is invariant and that for all $f, g \in \mathcal{D}(\mathcal{L})$:

$$\int_{\mathbf{E}} f\mathcal{L}g \, \mathrm{d}\mu = \int_{\mathbf{E}} g\mathcal{L}f \, \mathrm{d}\mu.$$

Unless specified otherwise, these notations will we used through the rest of this section. The associated *carré du champ operator* is given by:

$$\Gamma(f,g) = \frac{1}{2}\left[\mathcal{L}(fg) - f\mathcal{L}(g) - g\mathcal{L}(f)\right] \quad f, g \in \mathcal{D}(\mathcal{L}).$$

For all $f, g \in \mathcal{D}(\mathcal{L})$ the *Dirichlet form* is defined as:

$$\mathcal{E}(f,g) = \int_{\mathbf{E}} \Gamma(f,g)\mathrm{d}\mu.$$

We will see that in functional inequalities, a quantity of interest is the Dirichlet form $\mathcal{E}(f,f)$ and in what follows we simply denote it $\mathcal{E}(f)$. Since $\mu$ is the invariant measure, for all $f \in \mathcal{D}(\mathcal{L})$:

$$\mathcal{E}(f) = -\int_{\mathbf{E}} f\mathcal{L}f\mathrm{d}\mu.$$

**An example.** To illustrate these concepts, consider $X$ the Markov process on $\mathbb{R}^d$, associated to the Langevin formulation given in Equation (1.7), at a fixed temperature $\beta$. Its infinitesimal generator is:

$$\mathcal{L}_\beta f(x) = -\beta\langle\nabla f(x), \nabla U(x)\rangle + \frac{1}{2}\Delta f(x).$$

The unique invariant measure is the Gibbs measure $\mu_\beta$ of potential $U$ and temperature $1/\beta$:

$$\mu_\beta = \frac{e^{-\beta U(x)}}{Z_\beta}\mathrm{d}x \quad \text{with} \quad Z_\beta = \int_{\mathbb{R}^d} e^{-\beta U(x)}\mathrm{d}x$$

and its Dirichlet form is:

$$\mathcal{E}(f,g) = \int_{\mathbb{R}^d} \langle\nabla f(x), \nabla g(x)\rangle\mathrm{d}\mu_\beta.$$

In particular, it means that the Dirichlet form of a function $f$ is the squared $L^2(\mu_\beta)$ norm of its gradient:

$$\mathcal{E}(f) = \|\nabla f\|^2_{2,\mu_\beta}.$$

### Poincaré inequality and the $\chi^2$ distance

**$\chi^2$-distance.** Let $\mu_1$ and $\mu_2$ be two probability measures such that $\mu_2$ is absolutely continuous with respect to $\mu_1$ ($\mu_2 \ll \mu_1$), and denote $f$ its Randon-Nikodym derivative, $f \mathrm{d}\mu_1 = \mathrm{d}\mu_2$. The $\chi^2$- *distance* between $\mu_2$ and $\mu_1$ is defined by:

$$\chi^2(\mu_2, \mu_1) = \left( \int_{\mathbb{E}} (f-1)^2 \mathrm{d}\mu_1 \right)^{1/2}.$$

This can be written also as:

$$\chi^2(\mu_2, \mu_1) = \left( Var_{\mu_1}(f) \right)^{1/2},$$

where, of course, $Var_{\mu_1}(f)$ represents the variance of $f$ with respect to $\mu_1$:

$$Var_{\mu_1}(f) = \int_{\mathbf{E}} f^2 \mathrm{d}\mu_1 - \left( \int_{\mathbf{E}} f \mathrm{d}\mu_1 \right)^2.$$

**Poincaré's inequality.** We say that the measure $\mu$ satisfies a *Poincaré inequality with constant* $C_p$ (abbreviated $PI(C_p)$) if for all $f \in \mathcal{D}(\mathcal{L})$:

$$Var_\mu(f) \leq C_p \mathcal{E}(f).$$

The smallest constant $C_p$ such that this inequality holds is often referred to as the spectral gap and we will also refer to it as the optimal Poincaré constant. The above inequality depends not only on $\mu$ but also on the generator $\mathcal{L}$ and its domain. Thus, when saying only that a measure $\mu$ respects a Poincaré inequality, we make a slight abuse. It would be more accurate to say that $\mu$ respects a Poincaré inequality with respect to the Dirichlet form $\mathcal{E}$, on the class of function given by $\mathcal{D}(\mathcal{L})$.

The Poincaré inequality also implies the exponential convergence of the semigroup $(P_t)_{t \geq 0}$ in $L^2(\mu)$ (see for example Theorem 4.2.5 of [21]):

$$\forall f \in L^2(\mu) \quad Var(P_t f) \leq e^{-2t/C_p} Var(f). \tag{1.19}$$

This highlights the importance of the optimal Poincaré constant since a smaller $C_p$ implies a faster convergence rate. The proof of this theorem is based on the Grönwall Lemma and the fact that:

$$\frac{\mathrm{d}}{\mathrm{d}t} \left[ Var(P_t f) \right] = -2\mathcal{E}(P_t f).$$

**Link between the Poincaré's inequality and the spectral gap.** We have mentioned that the Poincaré inequality is also called the spectral gap inequality. To see why, suppose $\mathcal{L}$ is symmetric in $L^2(\mu)$ (ensuring thus that eigenvalues are real) and that $f$ is an is an eigenfunction of $-\mathcal{L}$ with eigenvalue $\lambda$. If the Poincaré inequality holds, then:

$$\int_{\mathbf{E}} f^2 \mathrm{d}\mu = Var_\mu(f) \leq C_P \mathcal{E}(f) = C_P \lambda \int_{\mathbf{E}} f^2 \mathrm{d}\mu.$$

This means that every non-zero eigenvalue of $-\mathcal{L}$ is greater than or equal to $1/C_p$, and thus the Poincaré inequality describes a *gap* in the generator's spectrum.

### Relative entropy and Logarithmic Sobolev inequality

The notion of entropy can be defined for a general measure $\mu_1$ (not necessarily finite). However in this thesis we only deal with probability measures and thus we restrain ourselves to this particular setting. Let $\mu_1$ be a fixed probability measure. For all positive integrable functions $f$ such that $\int_{\mathbf{E}} f|\log f| \mathrm{d}\mu_1 < +\infty$, we define the *entropy* of $f$ with respect to $\mu_1$ as:

$$\mathrm{Ent}_{\mu_1}(f) = \int_{\mathbf{E}} f \log(f) \mathrm{d}\mu_1 - \int_{\mathbf{E}} f \mathrm{d}\mu_1 \log\left(\int_{\mathbf{E}} f \mathrm{d}\mu_1\right),$$

In the above definition, $0 \log 0 = 0$. A first remark is that as a consequence of Jensen's inequality, $\mathrm{Ent}(f) \geq 0$ for all $f$ and $\mathrm{Ent}(f) = 0$ if and only if $f$ is constant $\mu_1$-almost everywhere. Also for all $c \geq 0$, $\mathrm{Ent}(cf) = c\mathrm{Ent}(f)$.

**Relative entropy.** Let $\mu_2$ be a probability measure such that $\mu_2 \ll \mu_1$, and let $f$ be its Randon-Nikodym derivative. The *relative entropy of $\mu_2$ with respect to $\mu_1$* is given by:

$$KL(\mu_2 \| \mu_1) = \mathrm{Ent}_{\mu_1}(f) = \int_{\mathbf{E}} \log f \mathrm{d}\mu_2$$

The relative entropy is a good way of quantifying the difference between two probability measures. Namely, if the relative entropy is null, then the two measures coincide. The *Csiszár-Kullback-Pinsker* inequality states that the total variation distance ($d_{TV}$) is upper bounded by the relative entropy, in the following sense:

$$d_{TV}(\mu_2, \mu_1) \leq \frac{1}{2}\sqrt{\mathrm{Ent}_{\mu_1}(f)}.$$

**Logarithmic Sobolev inequality** The measure $\mu$ respects a *logarithmic Sobolev inequality with constant $C_s$* (shortened as LSI($C_s$)) if for all $f \in \mathcal{D}(\mathcal{L})$:

$$\mathrm{Ent}_\mu(f^2) \leq 2C_s\mathcal{E}(f)$$

As in the case of the Poincaré inequality, the above inequality implies that if $\mathcal{E}(f)$ is zero, then $f$ is constant.

The logarithmic Sobolev inequality implies an exponential decay in entropy for the semigroup $(P_t)_{t\geq 0}$ (see for example Theorem 5.2.1 of [21]):

$$\forall f \in L^1(\mu) \quad \mathrm{Ent}_\mu(P_t f) \leq e^{-2t/C_s}\mathrm{Ent}_\mu(f) \tag{1.20}$$

The proof is based on the following de Bruijn identity (Proposition 5.2.2. of [21]):

$$\frac{\mathrm{d}}{\mathrm{d}t}\left[\mathrm{Ent}_\mu(P_t f)\right] = -\int_{\mathbf{E}} \frac{\Gamma\left(P_t f\right)}{P_t f} \mathrm{d}\mu =: -I_\mu(P_t f), \tag{1.21}$$

where $I_\mu$ is the Fisher information. Another important element of the proof is the fact that the logarithmic Sobolev inequality can also be written as:

$$\mathrm{Ent}_\mu(f) \leq \frac{C_s}{2} I_\mu(f),$$

and thus using (1.21) we obtain a differential inequality that allows us to conclude using the Grönwall Lemma:

$$\frac{\mathrm{d}}{\mathrm{d}t}\left[\mathrm{Ent}_\mu(P_t f)\right] \leq -\frac{2}{C_s}\mathrm{Ent}_\mu(P_t f).$$

### Poincaré vs Sobolev

A first remark is that the logarithmic Sobolev with constant $C_s$ implies the Poincaré inequality with the same constant:

$$\mathrm{LSI}(C_s) \Rightarrow \mathrm{PI}(C_s),$$

and is thus considered stronger in the hierarchy of functional inequalities.

The interest of the Poincaré inequality lies in the fact that it is generally easier to prove and can be obtained for a larger class of measures. It is sometimes used as a preliminary step in the proof of a Sobolev inequality. Moreover there are cases when both inequalities hold, but the optimal constant of the LSI is strictly larger than the optimal constant of PI. One should notice that the constants are important when comparing the rate of exponential decay in (1.19) and (1.20).

One can also observe that the convergence given in (1.20) holds for a larger class of functions $f$, since $L^2(\mu) \subset L^1(\mu)$ and the inclusion is strict. This is rather natural because there exist functions $f$ such that $\mathrm{Ent}_\mu(f)$ is finite and $Var_\mu = +\infty$. This occurs for example when $\mu$ is the standard normal distribution on $\mathbb{R}$ and $f(x) = e^{cx^2/2}$, with $c \in (0, 1)$.

This brings us to the last question: when trying to prove the convergence of a Markov process, what should we choose? The variance or the relative entropy?

### In practice: variance vs relative entropy

Indeed, going back to our general framework, the Poincaré and the Sobolev inequality are just instruments necessary in dealing with a particular term. To be more precise, one step of the convergence's proof is to measure the gap between the noisy algorithm and its classical version (with no noise) and another is to quantify the convergence of the classical algorithm to the wanted set. Functional inequalities intervene in a natural way in the second part.

Therefore, before choosing a distance, one needs to take several things into consideration:

- What kind of functional inequality can be obtained for the invariant measure of the classical version? If the Poincaré inequality holds but not the Sobolev one, then one needs to choose the variance.
- If both inequalities hold, a next criteria should be the type of convergence we want to obtain. Proving that the relative entropy of two measures goes to zero is stronger, since the relative entropy controls the total variation distance. However, as we will see later on, the fact that the $\chi^2$ distance goes to zero, might be enough to ensure the convergence in probability of the algorithm to the optimal set.
- A final question (and quite important in practice) is which of the two quantities allows us to properly control the gap between the noisy version and the classical one, or in other words, what quantity is easier to deal with from a technical point of view.

# Part I

# On the Subject of Graphs

# Chapter 2

# Fréchet mean on quantum graphs

Discrete structures like graphs make it possible to naturally and flexibly model complex phenomena. Since graphs that represent various types of information are increasingly available today, their analysis has become a popular subject of research. Yet, even an algorithm for locating the *average* position in graphs is lacking although this knowledge would be of primary interest for statistical analysis or representation problems. In this work, we develop a stochastic algorithm for finding the Fréchet mean of weighted undirected metric graphs. This method relies on a noisy simulated annealing algorithm dealt with using homogenization. We then illustrate our algorithm with three examples (subgraphs of a social network, subgraph of a collaboration and citation network and a transport network).

**Note to the reader:** The next nine sections of this Chapter consist in the article [66], written in collaboration with Sébastien Gadat and Laurent Risser. The article is soon to be published under the name *How to calculate the barycenter of a weighted graph*, in Mathematics of Operations Research. Section 2.10 presents some further developments.

## 2.1 Introduction.

### 2.1.1 Generalities

Numerous open questions in a very wide variety of scientific domains involve complex discrete structures that are easily modelled by graphs. The nature of these graphs may be weighted or not, directed or not, observed online or by using batch processing, each time implying new problems and sometimes leading to difficult mathematical or numerical questions. Graphs are the subject of perhaps one of the most impressive growing bodies of literature dealing with potential applications in statistical or quantum physics (see, *e.g.*, [57]) economics (dynamics in economy structured as networks), biology (regulatory networks of genes, neural networks), informatics (Web understanding and representation), social sciences (dynamics in social networks, analysis of citation graphs). We refer to [90] and [104] for recent communications on the theoretical aspects of random graph models, and to [83] and [69] for additional applications in economics and machine learning.

In the meantime, the nature of the mathematical questions raised by the models that involve networks is very extensive and may concern geometry, statistics, algorithms or dynamical evolution over the network, to name a few. For example, we can be interested in the definition of suitable random graph models (see the survey of [95]), or in graph geometry visualization (see some popular methods in [117]). In statistics, a popular topic deals with the estimation of some natural objects like clusters (see, *e.g*, [89] and [19]). A final important field of interest deals with the evolution of a dynamical system over a graph (see [4] and the references therein). Our work will mix several of the aforementioned fields since we will be interested in the estimation of a rather simple statistical object related to the graph geometry. This estimation will be produced by a random dynamical system that evolves on the graph. We address a problem here that may be considered as very simple at first glance: *we aim to define and estimate the barycenter*[1] *of a weighted graph.*

### 2.1.2 Motivation

Surprisingly, as far as we know, the question of locating the barycenter of a graph has received little attention certainly because of the difficulty of its resolution. However, the knowledge of the barycenter of a (weighted) graph may be of interest from several viewpoints.

First, understanding what node of the graph is the barycenter may be used to produce a graphical representation of the graph with the average position in the middle of the representation since it is expected that the barycenter is the central node of the network. Indeed, nowadays graph-visualization algorithms like force-directed methods usually minimize some heuristic criterion that does not calculate and use a center of mass of the graph. This drawback of the graph-visualization algorithms may be annoying because we should expect that a good assessment of the location of a weighted graph barycenter could be used for fair representation issues.

Second, in statistics, an elementary object that permit to compare several datasets is the mean position. This is for example the case with the T-test that makes it possible to compare if two Gaussian distributions are the same (or not) with a simple computation of the mean of two $n$-samples. If we now observe two graphs defined with the same set of nodes, the computation of the two barycenters could be used to test if these two graphs are sampled with the same probabilistic generating model.

In addition, the knowledge of the mean of a dataset is a preliminary information to produce a "second order" moment analysis with a generalization of the Principal Component Analysis (PCA for short in what follows). In a non-Euclidean dataset, this intermediary step was used by [32] to extend the definition of PCA on the space of probability measures on $\mathbb{R}$, and by [31] to develop a suitable geometric PCA of a set of images. For a graph analysis problem, the average position in a graph may be used to produce a PCA that would translate the main statistical fluctuations among the nodes of the network. Such PCA does not exist yet in particular because a definition and algorithm to compute the mean of a graph is an important (and however basic) unsolved problem.

Another motivation can be found in the analysis of traffic networks. For example, subway networks naturally form a connected weighted graph equipped with a distance between stations that are related to the time needed to travel from one station to another. Each station of the network may be weighted by its monthly use rate by all passengers if we are interested in a global understanding of the network. Each station can also be weighted by the individual use of one passenger if we are interested in an analysis of a personal use of the network. In this last case, these weights may be unknown and have to be learned on-line throughout the use of the network by the passenger. In each case, learning the location of the barycenter of such networks is of interest, either for a general firm (for improving the circulation in the main stations of the network) or for individual

[1] We give below a precise mathematical definition of the barycenter of a general weighted graph.

decision related to each passenger.

Lastly, understanding where is the barycenter of the graph permit to identify the most important nodes of the graph. We can also push the argument further and think that recursive computations of some barycenters of some subgraphs produced by a clustering method also make it possible to find the important set of nodes that describe the structure of the whole graph. For example, in a citation network, the barycenters of clusters of nodes are suspected to be the nodes that have a large influence on the whole connections of the graph. In particular, these barycenters may reveal the important historical thoughts hidden behind a citation network.

In the next paragraph, we describe a natural way to define barycenter for weighted graphs.

### 2.1.3 Fréchet means

**Variational formulation of the barycenter** The major difficulty for defining means in a general (and not necessarily Euclidean) metric space $E$ lies in the absence of the natural addition and averaging operations. A popular strategy used to define moments of distributions in general metric spaces is to use the variational interpretation of means, which leads to the introduction of Fréchet (or Karcher) means. This approach has been introduced in the seminal contribution [61] that makes it possible to define $p$-means over any metric probability space. Following the simple remark that the (quadratic) mean of any distribution $\nu$ of $\mathbb{R}^d$, given by $m_\nu = \int_{\mathbb{R}^d} x d\nu(x)$ is the point that minimizes $x \longmapsto \mathbb{E}_{Z\sim\nu}[|x-Z|^2]$, the Fréchet $p$-mean of any metric space $(\mathcal{E}, d)$ weighted by any distribution $\nu$ is defined by (see *e.g.* [61]):

$$M_\nu^{(p)} := \arg\min_{x\in\mathcal{E}} \mathbb{E}_{Z\sim\nu}[d^p(x, Z)].$$

We can then extend this definition to our particular graph structure and only restrict our study to quadratic barycenters, with $p = 2$. We also alleviate our notation using only $M_\nu$ instead of $M_\nu^{(2)}$.

**Definition 1** (Fréchet mean)**.** *Given a weighted graph $(G, \nu)$ equipped with the geodesic distance $d$ w.r.t. $G$, the Fréchet mean of the graph is the set of minimizers of $U_\nu$ defined by:*

$$M_\nu := \arg\min_{x\in V} U_\nu(x) \qquad \textit{where} \qquad U_\nu(x) = \frac{1}{2}\int_G d^2(x, y)\nu(y). \tag{2.1}$$

Note that $M_\nu$ set is not necessarily a singleton and a uniqueness property generally requires some additional topological assumptions (see [13], for example).

**Background on Fréchet means** The use of Fréchet means defined by Equation (2.1) has met with great interest, especially in the field of bio-statistics and signal processing, although mathematical and statistical derivations around this notion constitute a growing field of interest.

Almost all previous works on Fréchet means are interested in the convergence of $M_{\nu_n}$ towards $M_\nu$ when $\nu_n$ is an empirical measure produced by $n$ i.i.d. samples of $\nu$. We list below several works that are related to the convergence of $M_{\nu_n}$ towards $M_\nu$ when $n \longrightarrow +\infty$. Motivated by applications to continuous manifolds that describe shape spaces introduced in [54], many authors have recently proposed limit theorems on $M_{\nu_n}$ in various frameworks. For example, [92] establishes the consistency of the population Fréchet mean and derives applications in the Kendall space. The study of [24] establishes the consistency of $M_{\nu_n}$ and derives its asymptotic distribution. Finer results can be obtained in some non-parametric restrictive situations (see, *e.g.* [29, 30, 35]). Other applications involve signal processing: ECG curve analysis [26] and image analysis [111, 9], to name a few.

Recent works also treat Fréchet means in a discrete setting, especially when dealing with phylogenetic trees that have an important hierarchical structure property. In particular, [22] proposed a central limit theorem in this discrete case, whereas [98] used an idea of Sturm for spaces with non-positive curvature to define an algorithm for the computation of the population Fréchet mean. Other works deal with the averaging of discrete structure sequences such as diagrams using the Wasserstein metric (see, *e.g.*, [101]) or graphs [68]. Our work also deviates from these above-mentioned discrete studies [98, 22, 101, 68] since we build an algorithm that recovers the Fréchet mean of a weighted graph itself, instead of finding the population Fréchet mean of a set of discrete phylogenetic structures.

We discuss below on the computational difficulty required to solve the Fréchet mean computation.

**Computational complexity and on-line procedure** First of all and heuristically, it is highly suspected that finding the Fréchet mean that minimizes $U_\nu$ over the discrete graph and even on the continuous quantum graph that interpolates the discrete one is a difficult problem. Indeed, in the case of a discrete network within a deterministic setting, finding a $p$-mean is known to be a NP-hard problem (see *e.g.* [74] and [118]). Concerning now a continuous embedding of the network, the problem of finding $M_\nu$ involves the minimization of $U_\nu$, which is a non-convex function with the possibility of numerous local traps. To our knowledge, this problem cannot be efficiently solved using either a relaxed solution or using a greedy/dynamic programming algorithm (in the spirit of the Dijkstra method that makes it possible to compute geodesic paths [51]). Moreover, if we consider for example a continuous network supported by the unit cube $C_d$ of $\mathbb{R}^d$, where each node of the cube is weighted by a probability distribution $\nu$, then the minimization of $U_\nu$ turns into a quadratic programming over $C_d$, which is a NP-hard problem in its full generality (see *e.g.* [114, 108]). A rigorous reduction of the minimization Equation (2.1) over a continuous network to a NP-hard problem is an interesting issue, but beyond the scope of this work and deserves further investigations.

Second, since we will be interested in dealing with possibly on-line problems, the availability of the measure $\nu$ over the nodes of the graph that is used in the definition of $U_\nu$ is questionable and only a sequence of nodes i.i.d. according to $\nu$ may be available. This is for example the case if we consider the *zbMATH* dataset where downloaded articles are observed recursively. This is also the case when a passenger travels over a subway network without a priori information of its interest for the whole set of stations. In this kind of situations, we therefore need to learn the distribution $\nu$ all along the iterations of the algorithm in parallel to the location of the minimizer of $U_\nu$. This may be also the case if we consider a probability distribution over $E$ that is cropped while gathering interactive forms on a website…

Lastly, minimizing $U_\nu$ require a computation of an integral w.r.t. $\nu$. Since we plan to handle large graphs, such an integral computation can be a very strong limitation. Therefore, the algorithm we are looking for is also motivated by the legitimate impossibility (from a numerical point of view) to compute such an integral. Thus, our algorithm should work as a first order method that uses unbiased and noisy evaluations of gradients.

**Global optimization procedure with Simulated Annealing** The deterministic discrete problem being NP-hard, it is thus natural to think about the use of a global minimization procedure, and, in particular, the simulated annealing (S.A. for short) method. S.A. is a standard strategy to minimize a function over discrete spaces and its computational cost is generally high. It relies on an inhomogeneous Markov process that evolves on the graph with a transition kernel depending on the energy estimates $U_\nu$ for solving Equation (2.1). The method we are looking for must be compatible with the stochastic on-line issue or stochastic gradient approaches. Our algorithm uses recent contributions on simulated annealing ([13, 14]) where an auxiliary jump-process is introduced to handle on-line observations. It relies on a continuous-time noisy simulated annealing Markov process on graphs, as well as a second process that accelerates and homogenizes the updates of the noisy transitions in the simulated annealing procedure. We also refer to [99] that introduces a simulated annealing procedure with an auxiliary jump process which encodes a piecewise deterministic Markov process dynamical system.

### 2.1.4 Roadmap

The paper is organized as follows: Section 2.2 provides some brief remainders on graphs and then describes our proposed algorithm. In Section 2.3.1, we provide some theoretical backgrounds on diffusion on quantum graphs and Section 2.3.2 states our main convergence results. Simulations and numerical insights are then given in Section 2.4 and a short conclusion proposes some further developments in Section 2.5. The convergence of the algorithm is theoretically established in Section 2.7, whereas Section 2.8 describes functional inequalities in quantum graphs (Poincaré and Log-Sobolev inequalities for Gibbs field at low temperature).

# Acknowledgments.

The authors gratefully acknowledge Laurent Miclo for his stimulating discussions and helpful comments throughout the development of this work, and Nathalie Villa-Vialaneix for her interest and advice concerning simulations. The authors are also indebted to *zbMATH* for making their database available to produce numerical simulations.

## 2.2 Simulated annealing on quantum graphs

We consider $G = (V, E)$ a finite connected and undirected graph with no self-loop, where $V = \{1, \dots, N\}$ refers to the $N$ vertices (also called nodes) of $G$, and $E$ the set of edges that connect some couples of vertices in $G$.

### 2.2.1 Undirected weighted graphs

The structure of $G$ may be described by the adjacency matrix $W$ that gives a non-negative weight to each edge $E$ (pair of connected vertices), so that $W = (w_{i,j})_{1 \leq i,j \leq N}$ while $w_{i,j} = +\infty$ if there is no direct link between node $i$ and node $j$. $W$ indicates the length of each direct link in $E$: a small positive value of $w_{i,j}$ represents a small length of the edge $\{i, j\}$. We assume that $G$ is undirected (so that the adjacency matrix $W$ is symmetric) and connected: for any couple of nodes in $V$, we can always find a path that connects these two nodes. Finally, we assume that $G$ has no self loop. Hence, the matrix $W$ satisfies:

$$\forall i \neq j \quad w_{i,j} = w_{j,i} \qquad \text{and} \qquad \forall i \in V \quad w_{i,i} = 0.$$

We define $d(x, y)$ as the geodesic distance between two points $(x, y) \in V^2$, which is the length of the shortest path between them. The length of this path is given by the addition of the length of traversed edges:

$$\forall (i,j) \in V^2 \qquad d(i,j) = \min_{i=i_1 \to i_2 \to \ldots \to i_{k+1}=j} \sum_{\ell=1}^{k-1} w_{i_\ell, i_{\ell+1}}.$$

When the length of the edges is constant and equal to 1, it simply corresponds to the number of traversed edges. Since the graph is connected and finite, we introduce the definition of the diameter of $G$:

$$\mathcal{D}_G := \sup_{(x,y)\in V^2} d(x,y).$$

To define a barycenter of a graph, it is necessary to introduce a discrete probability distribution $\nu$ over the set of vertices $V$. This probability distribution is used to measure the influence of each node on the graph. According to this former definition of the geodesic distance $d$, we are interested in solving the problem introduced in Equation (2.1): $M_\nu := \arg\min_{x\in E} U_\nu(x)$ where $U_\nu(x) = \frac{1}{2}\int_G d^2(x,y)\nu(y)$.

**Example 1.** *Let us consider a simple scientometric example illustrated in Figure 2.1 and consider a "toy" co-authorship relation that could be obtained in a subgraph of a collaboration network like zbMATH*[2]*. If two authors $A$ and $B$ share $k_{A,B}$ joint papers, it is a reasonable choice to use a weight $w_{A,B} = \phi(k_{A,B})$, where $\phi$ is a convex function satisfying $\phi(0) = +\infty, \phi(1) = 1$ and $\phi(+\infty) = 0$. This means that no joint paper between $A$ and $B$ leads to the absence of a direct link between $A$ and $B$ on the graph. On the contrary, the more papers there are between $A$ and $B$, the closer $A$ and $B$ will be on the graph. Of course, this graph may be embedded in a probability space with the additional definition of a probability distribution over the authors that can be naturally proportional to the number of citations of each author. This is a generalization of the Erdös graph. Note that this type of example can also be encountered when dealing with movies and actors, leading, for example, to the Bacon number and graph (see the website: www.oracleofbacon.org/).*

### 2.2.2 Outline of Simulated Annealing (S.A.)

As pointed in Section 2.1.3, the optimization of Equation (2.1) is a NP-hard problem (see [118, 74, 108]) and it is therefore natural to use a global optimization algorithm like simulated annealing. We briefly detail below a description of the two main methods of S.A.

The optimization with S.A. introduces a Markov random dynamical system that evolves either in continuous time (generally for continuous spaces) or in discrete time (for discrete spaces). When dealing with a discrete setting, S.A. is based on a Markov kernel proposition $L(.,.) : V \times V \longrightarrow [0,1]$ related to the 1-neighborhoods of the Markov chain, as introduced in [73]. It is based on an inhomogeneous Metropolis-Hastings scheme,

[2] https://zbmath.org/

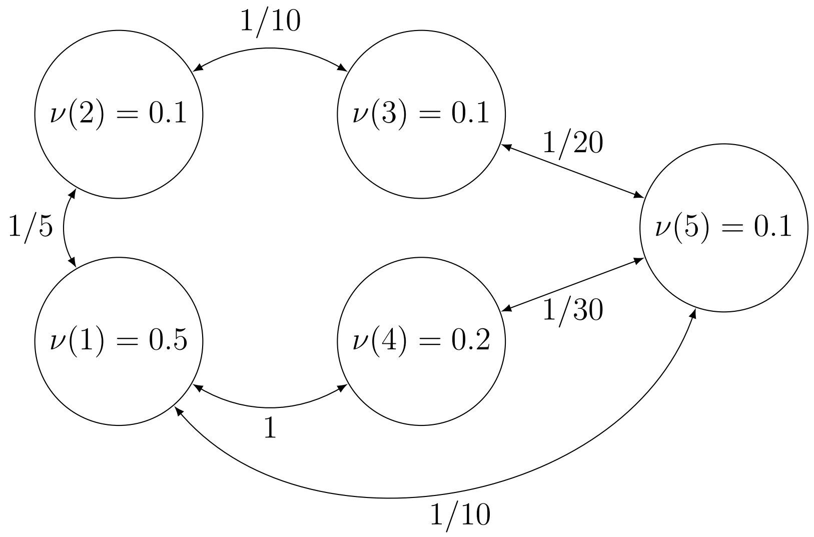


Figure 2.1: Example of a weighted graph between five authors, where the probability mass is $\{0.5, 0.1, 0.1, 0.2, 0.1\}$. In this graph, Author 1 shares five publications with Author 2 and ten publications with Author 5, and so on.

which is recalled in Algorithm 1. We can derive asymptotic guarantees of the convergence towards a global minimum of $U$ as soon as the cooling schedule is well chosen.

```
Algorithm 1: M.-H. Simulated Annealing
  Data: Function $U$. Decreasing temperature sequence $(T_k)_{k\geq 0}$.
1 Initialization: $X_0 \in V$;
2 for $k = 0 \dots +\infty$ do
3     Draw $x' \sim L(X_k, .)$ and compute $p_k = 1 \wedge \left\{ e^{T_k^{-1}[U(X_k)-U(x')]} \frac{L(x',X_k)}{L(X_k,x')} \right\}$.
4     Update $X_{k+1}$ according to $X_{k+1} = \begin{cases} x' \text{ with probability } p_k \\ X_k \text{ with probability } 1 - p_k. \end{cases}$
5 end
6 Output: $\lim_{k \longrightarrow +\infty} X_k$.
```

When dealing with a continuous setting, S.A. uses a drifted diffusion with a vanishing variance $(\epsilon_t)_{t\geq 0}$ over $V$, or an increasing drift coefficient $(\beta_t)_{t\geq 0}$. We refer to [76, 96] for details and we recall its Langevin formulation in Algorithm 2.

```
Algorithm 2: Langevin Simulated Annealing
  Data: Function $U$. Increasing inverse temperature $(\beta_t)_{t\geq 0}$.
1  Initialization: $X_0 \in V$;
2      $\forall t \geq 0 \qquad dX_t = -\beta_t \nabla U(X_t)dt + dB_t$.
3 Output: $\lim_{t \longrightarrow +\infty} X_t$.
```

In both cases, we can see that S.A. with $U = U_\nu$ given by Equation (2.1) involves the computation of the value of $U$ in Line 4 of Algorithm 1, or the computation of $\nabla U$ in Line 2 of Algorithm 2. These two computations are problematic for our Fréchet mean problem: the integration over $\nu$ is intractable in the situation of large graphs and we are naturally driven to consider a noisy version of S.A. A possible alternative method for this problem is to use a homogenization technique: replacing $U_\nu$ in the definition of $p_k$ by $U_y(\cdot) = d^2(\cdot, y)$, *where $y$ is a value from an i.i.d. sequence $(Y_n)_{n\geq 0}$ distributed according to $\nu$* . Such methods have been developed in [72] as a modification of Algorithm 1 with an additional Monte-Carlo step in Line 4, when $U_Y(x)$ follows a Gaussian distribution centered around the true value of $U(x) = \mathbb{E}_{Y\sim\nu} U_Y(x)$. This approach is still problematic in our case since the Gaussian assumption on the random variable $d^2(x, Y)$ is unrealistic here. Another limitation of this MC step relies on the fact that it requires a batch average of several $(U(x, Y_j))_{1\leq j\leq n_k}$ where $n_k$ is the number of observations involved at iteration $k$, although we also plan to develop an algorithm that may be adapted to on-line arrivals of the observation $(Y_n)_{n\geq 0}$. Lastly, it is important to observe that the non-linearity of the exponential prevents the use of only *one* observation $Y_k$ in the acceptation/reject ratio involved in the S.A. since it does not lead to an unbiased evaluation of the true transition:

$$\mathbb{E}_{Y\sim\nu}\left[e^{T_k^{-1}[U_Y(X_k)-U_Y(x')]}\right] \neq e^{T_k^{-1}\mathbb{E}_{Y\sim\nu}[U_Y(X_k)-U_Y(x')]}\,.$$

This difficulty does not arise in the homogenization of the simulated annealing algorithm in the continuous case since the exponential is replaced by a gradient, *i.e.*, the process we will use is a Markov process of the form:

$$\mathrm{d}X_t = -\beta_t \nabla U_{Y_t}(X_t)dt + dB_t.$$

Let us describe the main ingredient we will need to introduce below:

- $dB_t$ is a Brownian increment on the graph $G$ and will correspond either to a standard Brownian increment inside the edges or to a random walk displacement near the nodes of the graph. A more precise construction will be obtained through the martingale problem representation of Markov semi-group, in Theorem 2 of Section 2.3.1.

- $\beta_t^{-1}$ refers to the inverse of the temperature, which is a cornerstone of the theoretical studies on simulated annealing procedures.

- $Y_t$ is a continuous time Markov process obtained from the sequence $(Y_n)_{n\in\mathbb{N}}$.

This point motivates the introduction of the quantum graph induced by the initial graph. Of course, dealing with a continuous diffusion over a quantum graph $G$ deserves special theoretical attention, which will be given in Section 2.3.1.

### 2.2.3 Homogenized S.A. algorithm on a quantum graph

We now present the proposed algorithm for estimating Fréchet means. To do so, we first introduce the quantum graph $\Gamma_G$ derived from $G = (V, E)$ that corresponds to the set of points living inside the edges $e \in E$ of the initial graph. Once an orientation is arbitrarily fixed for each edge of $E$, the location of a point in $\Gamma_G$ depends on the choice of an edge $e \in E$ and on a coordinate $x_e \in [0, L_e]$ where $L_e$ is the length of edge $e$ on the initial graph. The coordinate 0 then refers to the initial point of $e$ and $L_e$ refers to the other extremity. Figure 2.2 briefly describes an example of the relaxation from a discrete network to a quantum graph. In the discrete graph, the points $X_1$ and $X_2$ do not belong to the graph, while in the quantum settings these two points belong to $\Gamma_G$ and the geodesic path between $X_1$ and $X_2$ travels through the node $V_2$.

While considering the quantum graph $\Gamma_G$, it is still possible to define the geodesic distance between any point $x \in \Gamma_G$ and any node $y \in G$ (see dashed red line in Figure 2.2). In particular, when $x \in V$, we use the initial definition of the geodesic distance over the discrete graph, although when $x \in e \in E$ with a coordinate $x_e \in [0, L_e]$, the geodesic distance between $x$ and $y$ is:

$$d(x, y) = \{x_e + d(e(0), y)\} \wedge \{L_e - x_e + d(e(L_e), y)\}\,.$$

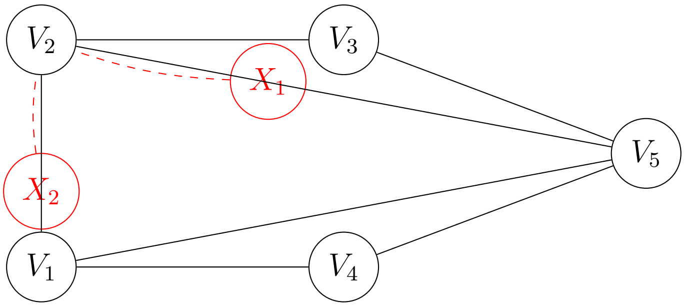


Figure 2.2: An example of a quantum graph: $X_1$ and $X_2$ may live inside the edges and not only on the nodes of the graph. Here, $X_1 \in \Gamma_G$ is on $[V_2, V_5]$ and $X_2 \in \Gamma_G$ is on $[V_1, V2]$ but $X_1 \notin G$ and $X_2 \notin G$. The geodesic (shortest) path between $X_1$ and $X_2$ is shown in dashed red line.

This definition can be naturally generalized to any two points of $\Gamma_G$, enabling us to consider the metric space $(\Gamma_G, d)$. Finally, it is worth saying that the orientation of any edge will not have any influence on the proposed algorithm since the definition of the geodesic distance on $\Gamma_G$ is kept invariant when we change one orientation of one edge. Note that such a modification of orientation will introduce a change of sign in the definition of the first order derivative, but the location of the minimizer of $U_\nu$ remains unchanged.

**Regularity issues** A first remark is that $U_\nu$ is continuous and piecewise $\mathcal{C}^\infty$ on $\Gamma_G$, since it can be written as a weighted sum of functions $U_y$, that are themselves continuous piecewise $\mathcal{C}^\infty$. Indeed, for a fixed vertex $y$, the function $U_y$ might have points on which its derivative is not well defined.

- This can be the case inside an edge $x \in e$ with $x_e \in (0, L_e)$ when at least two different geodesic paths from $x$ to $y$ start in opposite directions.
- This can also be the case at a node $x \in V$ when several geodesic paths start from $x$ to $y$.

Since the graph has a finite number of nodes and edges, $U_\nu$ is smooth almost everywhere on $\Gamma_G$. However its irregularities raise some technical issues and we are forced to approach it by suitable functions. To that end, for all $t \geq 0$, let $\mathbf{d}_t : \Gamma_G \times V \to \mathbb{R}$ be such that when $t$ goes to infinity $\mathbf{d}_t$ converges uniformly to the restriction of the distance $d$ to $\Gamma_G \times V$:

$$\mathbf{d} : \mathbb{R}_+ \times \Gamma_G \times V \to \mathbb{R} \quad \text{such that} \quad \mathbf{d}_t \xrightarrow{u} d, \tag{2.2}$$

and for all $t \geq 0$ and all $y \in V$, $\mathbf{d}_t(\cdot, y)$ is smooth enough in $x$. What we mean precisely by smooth enough and what exactly motivates this regularization procedure will be explained in Section 2.6.

Since our goal is to approach $U_\nu$, we define:

$$\mathbf{U}_{\nu,t}(x) = \sum_{y \in V} \mathbf{d}_t^2(x, y)\nu(y),$$

and denote $\mathbf{U}_{y,t} = \mathbf{d}_t^2(\cdot, y)$. An important remark is that, due to the uniform convergence of $\mathbf{d}_t$, we also have:

$$\mathbf{U}_{\nu,t} \xrightarrow{u} U_\nu, \qquad M_{\nu,t} \longrightarrow M_\nu \tag{2.3}$$

where $u$ denotes the uniform convergence and $M_{\nu,t}$ the set of minimizing points of $\mathbf{U}_{\nu,t}$.

**Dynamic of the homogenized simulated annealing** Consider a positive, continuous and increasing function $t \longmapsto \alpha_t$ such that:

$$\lim_{t \longmapsto +\infty} \alpha_t = +\infty \qquad \text{and} \qquad \forall t \geq 0 \quad \beta_t = o(\alpha_t).$$

We introduce $(N_t^\alpha)_{t \geq 0}$, an inhomogeneous Poisson process over $\mathbb{R}_+$ with intensity $\alpha$. It is standard to represent $N^\alpha$ through a homogeneous Poisson process $H$ of intensity 1 using the relationship:

$$\forall t \geq 0 \qquad N_t^\alpha = H_{h(t)}, \text{ where } h(t) = \int_0^t \alpha_s \mathrm{d}s.$$

**Definition 2** (Regularized homogenized simulated annealing over $\Gamma_G$)**.** *Given an accelerated process $(N_t^\alpha)_{t \geq 0}$ and an i.i.d. sequence of nodes $(Y_n)_{n \geq 1}$ distributed according to $\nu$, our regularized homogenized S.A. solves the following S.D.E. over $\Gamma_G$:*

$$\begin{cases} X_0 \in \Gamma_G \\ dX_t = -\beta_t \nabla \mathbf{U}_{Y_{N_t^\alpha}, t}(X_t) dt + dB_t. \end{cases} \tag{2.4}$$

Using the definition of $N_t$ and the basic properties of a Poisson Process, it can be observed that for all $\epsilon > 0$ and all $t \geq 0$:

$$\mathbb{E}\left[\frac{N_{t+\epsilon} - N_t}{\epsilon}\right] = \frac{1}{\epsilon} \int_t^{t+\epsilon} \alpha_s \mathrm{d}s.$$

Hence, $\alpha$ should be understood as the speed of new arrivals in the sequence $(Y_n)_{n \in \mathbb{N}}$.

We will describe an efficient discretization of Equation (2.4) that makes it possible to derive our practical optimization algorithm.

**Algorithm 3:** Homogenized Simulated Annealing over a quantum graph

```
Data: Function $U$. Increasing inverse temperature $(\beta_t)_{t \geq 0}$. Intensity $(\alpha_t)_{t \geq 0}$
Initialization: Pick $X_0 \in \Gamma_G$. ;
$T_0 = 0$ ;
for $k = 0 \ldots + \infty$ do
    while $N_t^\alpha = k$ do
        $X_t$ evolves as a Brownian motion, relatively to the structure of $\Gamma_G$
        initialized at $X_{T_k}^-$.
    end
    $T_{k+1} := \inf\{t : N_t^\alpha = k+1\}$;
    At time $t = T_{k+1}$, draw $Y_{k+1} = Y_{N_t^\alpha}$ according to $\nu$.
    The process $X_t$ jumps from $X_t^-$ towards $Y_{k+1}$:
        $$X_t = X_t^- + \beta_t \alpha_t^{-1} \overrightarrow{X_t Y_{N_t^\alpha}}, \tag{2.5}$$
    where $\overrightarrow{X_t Y_{N_t^\alpha}}$ represents the shortest (geodesic) path from $X_t$ to $Y_{N_t^\alpha}$ in $\Gamma_G$.
end
Output: $\lim_{t \longrightarrow +\infty} X_t$.
```

Algorithm 3 could be studied following the road map of [15]. Nevertheless, this implies serious regularity difficulties on the densities and the Markov semi-group involved. Furthermore, they propose a method for finding Fréchet means on a circle and thus adapting it to the graph's geometry raises additional difficulties. Hence, we have chosen to consider Algorithm 3 as a natural Euler explicit discretization of our Markov evolution (2.4): for a large value of $k$, the average time needed to travel from $T_k$ to $T_{k+1}$ is approximately $\alpha_{T_k}^{-1} \longrightarrow 0$ as $k \longrightarrow +\infty$. On this short time interval, the drift term in (2.4) is the gradient

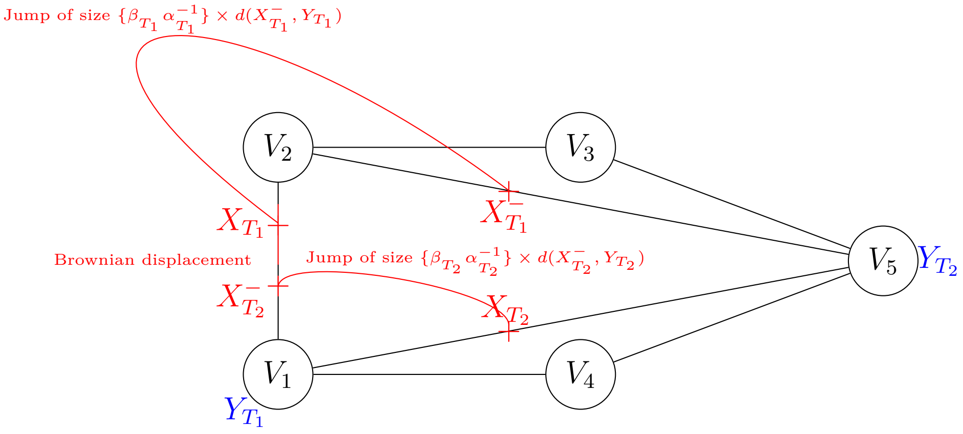

Jump of size $\{\beta_{T_1}\alpha_{T_1}^{-1}\} \times d(X_{T_1}^-, Y_{T_1})$
$V_2$
$V_3$
$X_{T_1}$
$X_{T_1}^-$
Brownian displacement
Jump of size $\{\beta_{T_2}\alpha_{T_2}^{-1}\} \times d(X_{T_2}^-, Y_{T_2})$
$V_5$
$Y_{T_2}$
$X_{T_2}^-$
$X_{T_2}$
$V_1$
$V_4$
$Y_{T_1}$

Figure 2.3: Schematic evolution of the Homogenized S.A. described in Algorithm 3 over the quantum graph. We first observe a jump at time $T_1$ towards $Y_{T_1} = V_1$ and a Brownian motion on $\Gamma_G$ during $T_2 - T_1$. A second node is then sampled according to $\nu$: here, $Y_{T_2} = V_5$ and a jump towards $Y_{T_2}$ occurs at time $T_2$.

of the squared geodesic distance between $X_{T_k}$ and $Y_{N^\alpha_{T_k}}$, except in small neighborhoods of cut-locus points (where the drift is a regularized version of the gradient). This is approximated by our vector $\overrightarrow{X_tY_k}$, multiplied by $\beta_{T_k}$, leading to Equation (2.5). $X_t$ now evolves as a Brownian motion over $\Gamma_G$ between two jump times and this evolution can be simulated with a Gaussian random variable using a (symmetric) random walk when the algorithm hits a node of $\Gamma_G$. Figure 2.3 proposes a schematic evolution of $(X_t)_{t\geq 0}$ over a simple graph $\Gamma_G$ with five nodes. We will prove the following result.

**Theorem 1.** *A constant $c^\star(U_\nu)$ exists such that if $\alpha_t = 1+t$ and $\beta_t = b\log(1+t)$ with $b < \{c^\star(U_\nu)\}^{-1}$, then $(X_t)_{t\geq 0}$ defined in (2.4) converges in probability to $M_\nu$ defined in Equation (2.1), when $t$ goes to infinity.*

### About the constant $c^\star(U_\nu)$

The constant $c^\star(U_\nu)$ plays a central role in the statement of Theorem 1. To precisely define $c^\star(U_\nu)$, we first introduce some useful notations. For any couple of vertices $(x, y)$ of $\Gamma_G$, and for any path $\gamma_{x,y}$ that connects them, we define $h(\gamma_{x,y})$ as the highest value of $U_\nu$ on $\gamma_{x,y}$:

$$h(\gamma_{x,y}) = \max_{s\in\gamma_{x,y}} U_\nu(s).$$

We define $H(x, y)$ as the smallest value of $h(\gamma_{x,y})$ obtained for all possible paths from $x$ to $y$:

$$H(x,y) = \min_{\gamma:x\to y} h(\gamma)$$

Now, for any pair of vertices $x$ and $y$, the notation $\gamma_{x,y}$ will be reserved for the path that attains the minimum in the definition of $H(x, y)$. Such a path exists for any $x, y$ because $\Gamma_G$ is connected and possesses a finite number of paths that connect any two given vertices.

Finally, we introduce the quantity that will mainly determine the size of the spectral gap involved in the functional inequality satisfied by $\mu_\beta$ when $\beta \longrightarrow +\infty$ (see the seminal works of [63] for an interpretation as a large deviation principle and [76] for a functional analysis point of view):

$$c^\star(U_\nu) := \max_{(x,y)\in\Gamma_G^2} [H(x,y) - U_\nu(x) - U_\nu(y)] + \min_{x\in\Gamma_G} U_\nu(x). \tag{2.6}$$

Figure 2.4 proposes a simplified illustration of the value of $c^\star(U_\nu)$ when the state space is of dimension 1. This illustration can be extended to our quantum graph model with the help of a more complex set of possible paths $\gamma_{x,y}$.
The uniform convergence 2.2 also implies:

$$c^\star(\mathbf{U}_{\nu,t}) \longrightarrow c^\star(U_\nu), \tag{2.7}$$

where $c^\star(\mathbf{U}_{\nu,t})$ is the analog of the $c^\star(U_\nu)$ defined in Equation (2.6).

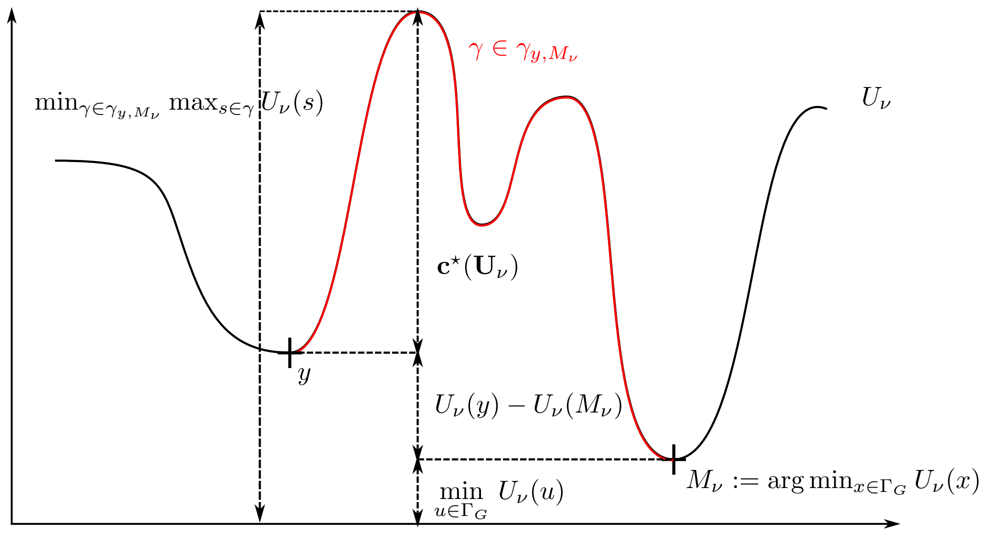

$\min_{\gamma\in\gamma_{y,M_\nu}} \max_{s\in\gamma} U_\nu(s)$
$\gamma \in \gamma_{y,M_\nu}$
$U_\nu$
$\mathbf{c}^\star(\mathbf{U}_\nu)$
$y$
$U_\nu(y) - U_\nu(M_\nu)$
$M_\nu := \arg\min_{x\in\Gamma_G} U_\nu(x)$
$\min_{u\in\Gamma_G} U_\nu(u)$

Figure 2.4: An example of a function $U_\nu$ and of the value $c^\star(U_\nu)$ when the possible paths are restricted to horizontal displacements.

### 2.2.4 Roadmap of our theoretical study

More details about the process defined in Equation (2.4) are presented in Section 2.3.2 and in particular the well-posedness of the Martingale problem is given by Theorem 2 in Section 2.3.1. Then, a more rigorous form of Theorem 1 above is given in the statement of Theorem 3 (see Section 2.3.2). Theorem 3 will involve the entropy convergence of the law of $X_t$ towards an associated Gibbs measure $\mu_{\beta_t}$. Our regularization procedure is properly defined and explained in Section 2.6 and the proof of Theorem 3 is deferred to Section 2.7.

## 2.3 Inhomogeneous Markov process over $\Gamma_G$

This section presents the theoretical background needed to define the Markov evolution Equation (2.4).

### 2.3.1 Diffusion processes on quantum graphs

We adopt here the convention introduced in [62] and fix for any edge $e \in E$ of length $L_e$ an orientation (and parametrization $s_e$). This means that $s_e(0)$ is one of the extremities of $e$ and $s_e(L_e)$ is the other one. By doing so, we have determined an orientation for $\Gamma_G$.

**Dynamical system inside one edge** To precisely define and obtain the weak existence of a couple $(X_t, Y_t)_{t\geq 0}$ solution of Equation (2.4), we use the Markovian approach and define the associated infinitesimal generator of the process.

**Definition 3** (Diffusion operator $\mathcal{L}_e$ over an edge $e$)**.** *Following the parametrization of each edge, we can define the second order elliptic operator $\mathcal{L}_e$ as $\forall(x, y, t) \in e \times V(G) \times \mathbb{R}_+$:*

$$\mathcal{L}_e f(x, y, t) = -\beta_t \langle \nabla_x \mathbf{U}_{y,t}(x), \nabla_x f(x, y)\rangle + \frac{1}{2}\Delta_x f(x, y) + \alpha_t \int_G [f(x, y') - f(x, y)] d\nu(y') + \partial_t f. \tag{2.8}$$

The generator $\mathcal{L}_e$ is associated with Equation (2.4) when $x \in e$: the $x$-component follows a standard diffusion drifted by $\nabla_x \mathbf{U}_{y,t}(.)$ inside the edge $e$, although the $y$-component jumps over the nodes of the initial graph $G$ with a jump distribution $\nu$ and a rate $\alpha_t$.

Since the drift term $\nabla \mathbf{U}_{y,t}(.)$ is measurable w.r.t. the Lebesgue measure over $e$ and the second-order part of the operator is uniformly elliptic, $\mathcal{L}_e$ uniquely defines (in the weak sense) a diffusion process up to the first time the process hits one of the extremities of $e$ (see, *e.g.*, [82]), which leads to a Feller Markov semi-group.

**Dynamical system near one node** We adopt the notation of [65] and write $e \sim v$ when a vertex $v \in V$ is an extremity of an edge $e \in E$. For any function $f$ on $\Gamma_G$, at any point $v \in V$, we can define the directional derivative of $f$ with respect to an edge $e \sim v$ according to the parametrization of $e$. If $n_v$ denotes the number of edges $e$ such that $e \sim v$, we then obtain $n_v$ directional derivatives designated as $(d_e f(v))$:

$$d_e f(v) = \begin{cases} \lim\limits_{h \longrightarrow 0^+} \dfrac{f(s_e(h)) - f(s_e(0))}{h} & \text{if} \quad s_e(0) = v \\[2ex] \lim\limits_{h \longrightarrow 0^-} \dfrac{f(s_e(L_e + h)) - f(s_e(L_e))}{h} & \text{if} \quad s_e(L_e) = v. \end{cases}$$

The dynamical system near one node behaves as follows: when the $x$ component of $(X_t, Y_t)_{t\geq 0}$ hits an extremity $v \in V$, it is instantaneously reflected in one of the $n_v$ edges connected to $v$ (with a uniform probability distribution over the connected edges) while spending no time on $v$. To sum-up, $(X_t)_{t\geq 0}$ is uniformly and instantaneously reflected when he hits any node of $V$.

It is shown in [65] that general dynamics over quantum graphs depend on a set of positive coefficients:

$$\mathcal{A} := \left\{ (a_v, (a_{e,v})_{e\sim v}) \in \mathbb{R}_+^{1+n_v} \text{ s.t. } a_v + \sum_{e\sim v} a_{e,v} > 0 : \forall v \in V \right\}.$$

There then exists a one-to-one correspondence between $\mathcal{A}$ and the set of all possible continuous Markov Feller processes on $\Gamma_G$. More precisely, if the global generator $\mathcal{L}_t$ is defined as

$$\forall (x, y) \in \Gamma_G \times V \qquad \mathcal{L}_t f(x, y) = \mathcal{L}_{e,t}(f)(x, y) \qquad \text{when} \qquad x \in e,$$

while $f$ belongs to the domain:

$$\mathcal{D}(\mathcal{L}) := \left\{ \forall y \in G \quad f(., y) \in \mathcal{C}^2(\Gamma_G) \ : \forall v \in V \qquad \sum_{e\sim v} a_e d_e f(v, y) = 0 \right\}, \tag{2.9}$$

then the martingale problem is well-posed (see [65, 58]).
Such a set of conditions on the partial derivatives of functions is also referred to as Neumann boundary conditions. From the symmetric instantaneous reflections we imposed, we introduce our *gluing conditions*.

**Definition 4** (Gluing conditions on nodes)**.** *For any $v \in V$, we define $(a_v, (a_{e,v})_{e\sim v})$ as*

$$a_v = 0 \qquad \textit{and} \qquad \forall e \sim v \qquad a_{e,v} = \frac{1}{n_v}.$$

Using the uniform ellipticity of $\mathcal{L}_e$ and the measurability of the drift term, Theorem 2.1 of [65] can be adapted, providing the well-posedness of the Martingale problem associated with $(\mathcal{L}, \mathcal{D}(\mathcal{L}))$, and the next preliminary result can then be obtained.

**Theorem 2.** *The operator $\mathcal{L}$ associated with the gluing conditions $\mathcal{A}$ generates a Feller Markov process on $\Gamma_G \times V \times \mathbb{R}_+$, with continuous sample paths on the $x$ component. This process is weakly unique and follows the S.D.E. Equation (2.4) on each $e \in E$.*

For simplicity, $\Delta_x$ will refer to the Laplacian operator with respect to the $x$-coordinate on the quantum graph $\Gamma_G$, using our gluing conditions given in Definition 4 and Equation (2.9) and the formalism introduced in [65].

### 2.3.2 Convergence of the homogenized S.A. over $\Gamma_G$

As mentioned in Section 2.2.2 and 2.2.3, we use a homogenization technique that involves an auxiliary sequence of random variables $(Y_n)_{n\geq 1}$, which are distributed according to $\nu$. More specifically, the stochastic process $(X_t, Y_t)_{y\geq 0}$ described above is depicted by its inhomogeneous Markov generator, which can be split into two parts:

$$\mathcal{L}_t f(x,y) = \mathcal{L}_{1,t} f(x,y) + \mathcal{L}_{2,t} f(x,y).$$

In the equality above, $\mathcal{L}_{1,t}$ is the part of the generator that acts on $Y_t$:

$$\mathcal{L}_{1,t} f(x,y) = \alpha_t \int [f(x,y') - f(x,y)]\nu(\mathrm{d}y'), \tag{2.10}$$

describing the arrival of a new observation $Y \sim \nu$ with a rate $\alpha_t$ at time $t$. Concerning the action on the $x$ component, the generator is:

$$\mathcal{L}_{2,t} f(x,y) = \frac{1}{2}\triangle_x f(x,y) - \beta_t < \nabla_x \mathbf{U}_{y,t}(x), \nabla_x f(x,y) > . \tag{2.11}$$

Since the couple $(X_t, Y_t)_{t\geq 0}$ is Markov with a renewal of $Y$ with $\nu$, it can be immediately observed that the $y$ component is distributed at any time according to $\nu$. We introduce the notation $m_t$ to refer to the distribution of the couple $(X_t, Y_t)$ at time $t$, and we define $n_t$ as the marginal distribution of $X_t$. In the following, we will also need to deal with the conditional distribution of $Y_t$ given the position $X_t$ in $\Gamma_G$. We will refer to this probability distribution as $m_t(y|x)$. To sum up, we have:

$$\mathscr{L}(X_t, Y_t) = m_t \quad \text{with} \quad n_t(x)dx = \int_V m_t(x,y)\mathrm{d}y \quad \text{and} \quad m_t(y\,|x) := \mathbb{P}[Y = y\,|X_t = x]. \tag{2.12}$$

A traditional method for establishing the convergence of S.A. towards the minimum of a function $U_\nu$ consists in studying the evolution of the law of $(X_t)_{t\geq 0}$ and, in particular, its close relationship with the Gibbs field $\mu_{\beta_t}$ with energy $U_\nu$ and inverse temperature $\beta_t$:

$$\mu_{\beta_t} = \frac{1}{Z_{\beta_t}} \exp(-\beta_t \mathbf{U}_{\nu,t}(x)), \tag{2.13}$$

where $Z_{\beta_t}$ is the normalization factor given by:

$$Z_{\beta_t} = \int e^{-\beta_t \mathbf{U}_{\nu,t}(x)} \mathrm{d}x. \tag{2.14}$$

At a fixed temperature $\beta^{-1}$, it is well known (see *e.g.* [21]) that the law of the process $(X_t)_{t\geq 0}$ converges exponentially fast towards its invariant measure. Now, using a slowly decreasing temperature scheme $t \longmapsto \beta_t^{-1}$, it is expected that the invariant measure itself $\mu_{\beta_t}$ does not evolve too much when the temperature $(\beta_t)_{t\geq 0}$ varies, so that the process $(X_t)_{t\geq 0}$ remains close to $\mu_{\beta_t}$. In addition, the Laplace method on the sequence $(\mu_{\beta_t})_{t\geq 0}$, together with (2.3), ensures that the measure is concentrated near the global minimum of $U_\nu$ (see, for example, the large deviation principle associated with $(\mu_\beta)_{\beta\to+\infty}$ in [63]).

Hence, a natural consequence of the convergence "$\mathscr{L}(X_t) \longrightarrow \mu_{\beta_t}$" and of the weak asymptotic concentration of $(\mu_{\beta_t})_{t\geq 0}$ around $M_\nu$ would be convergence in probability of the algorithm towards the Fréchet mean:

$$\lim_{t\longrightarrow+\infty} P(X_t \in M_\nu) = 1.$$

We refer to [76] for further details. In particular, a strong requirement for this convergence can be considered through the relative entropy of $n_t$ (the law of $X_t$) with respect to $\mu_{\beta_t}$:

$$J_t := KL(n_t||\mu_{\beta_t}) = \int_{\Gamma_G} \log\left[\frac{n_t(x)}{\mu_{\beta_t}(x)}\right] dn_t(x). \tag{2.15}$$

The function $\beta_t$ will be chosen as a $\mathcal{C}^2$ function of $\mathbb{R}_+$, and since $\mu_\beta$ is a strictly positive measure over $\Gamma_G$, it implies that $t \longmapsto \mu_{\beta_t}(x)^{-1}$ is $\mathcal{C}^2(\mathbb{R}_+\times\Gamma_G)$. Moreover, $(t,x) \longmapsto n_t(x)$ follows the backward Kolmogorov equation, which induces a $\mathcal{C}^2(\mathbb{R}_+\times\Gamma_G)$ function. Since the semi-group is uniformly elliptic on the $x$-component, we have:

$$\forall t > 0 \qquad \forall x \in \Gamma_G \qquad n_t(x) > 0.$$

On the basis of these arguments, we can deduce:

**Proposition 1.** *Assume that $t \longmapsto \beta_t$ is $\mathcal{C}^2$, then $(t,x) \longmapsto n_t(x)$ defines a positive $\mathcal{C}^2(\mathbb{R}_+^*\times\Gamma_G)$ function and $t \longmapsto J_t$ is differentiable for any $t > 0$.*

If we define:

$$\alpha_t = \lambda(t+1) \text{ and } \beta_t = b\log(t+1),$$

with $b$ a constant strictly smaller than $c^\star(U_\nu)^{-1}$, where $c^\star(U_\nu)$ is the maximal depth of a well not containing a fixed global minimum of $U_\nu$, defined in Equation (2.6), then our main result can be stated as follows:

**Theorem 3.** *For any constant $\lambda > 0$ such that $\alpha_t = \lambda(t+1)$ and $\beta_t = b\log(1+t)$ with $b < \{c^\star(U_\nu)\}^{-1}$, then:*

$$\lim J_t = 0 \qquad \textit{as} \qquad t \longrightarrow +\infty.$$

This ensures that the process $X_t$ will converge in probability towards $M_\nu$ and, therefore, towards a global minimum of $U_\nu$.

The idea of the proof is to obtain a differential inequality for $J_t$, which implies its convergence towards 0. It is well known that the Gibbs measure $\mu_{\beta_t}$ is the unique invariant distribution of the stochastic process that evolves only on the $x$ component, whose Markov generator is given by:

$$\widehat{\mathcal{L}}_{2,t}(f)(x) = \frac{1}{2}\Delta_x f - \beta_t\langle\nabla_x f, \nabla_x \mathbf{U}_{\nu,t}\rangle. \tag{2.16}$$

Therefore, a natural step of the proof will be to control the difference between $\mathcal{L}_{2,t}$ and $\widehat{\mathcal{L}}_{2,t}$ and to use this difference to study the evolution of $m_t$ and $n_t$. It can be seen that

$\widehat{\mathcal{L}}_{2,t}$ may be written as an average action of the operator thanks to the linearity of the gradient operator:

$$\widehat{\mathcal{L}}_{2,t}(f) = \frac{1}{2}\Delta_x f - \beta_t \mathbb{E}_{y\sim\nu}\langle \nabla_x f, \nabla_x \mathbf{U}_{y,t}\rangle.$$

When $X_t = x$, we know that $Y_t$ is distributed according to the distribution $m_t(y|x)$. Consequently, the average action of $\mathcal{L}_{2,t}$ on the $x$ component is:

$$\widetilde{\mathcal{L}}_{2,t}(f) = \frac{1}{2}\Delta_x f - \beta_t \mathbb{E}_{y\sim m_t(.|x)}\langle \nabla_x f, \nabla_x \mathbf{U}_{y,t}\rangle = \frac{1}{2}\Delta_x f - \beta_t \int_V \langle \nabla_x f, \nabla_x \mathbf{U}_{y,t}\rangle m_t(y|x)\mathrm{d}y, \tag{2.17}$$

whose expression may be close to that of $\widehat{\mathcal{L}}_{2,t}$ if $m_t(.|x)$ is close to $\nu$.

Thus, another important step is to choose appropriate values for $\alpha_t$ and $\beta_t$, *i.e.*, to find the balance between the increasing intensity of the Poisson process and the decreasing temperature schedule, in order to quantify the distance between $\nu$ and $m_t(.|x)$. The main core of the proof brings together these two aspects and is detailed in Section 2.7.

A necessary condition for the method presented above is:

$$\Gamma_G \times V \ni (x,y) \longmapsto \log \frac{n_t(x)}{\mu_{\beta_t}(x)} \in \mathcal{D}(\mathcal{L}). \tag{2.18}$$

This is the technical detail that demands a regularized version of $U_\nu$ and $\mathbf{d}_t$ is defined precisely to ensure that (2.18) holds. Namely, $\mathbf{d}_t(\cdot, y)$ is $\mathcal{C}^2(\Gamma_G)$ for all $y \in V$ and $t \geq 0$, and all of its directional derivatives are null. In Section 2.6 we explain why this is enough and prove that such a function $\mathbf{d}_t$ always exists.

Another important step will be the use of functional inequalities (Poincaré and log-Sobolev inequalities) over $\Gamma_G$ for the measure $\mu_\beta$ when $\beta = 0$ and $\beta \longrightarrow +\infty$. The proof of these technical results are given in Section 2.8.3.

**Corollary 1.** *Assume that $\beta_t = b\log(t+1)$ with $b < c^\star(U_\nu)^{-1}$ and $\alpha_t = \lambda(t+1)^\gamma$ with $\gamma \geq 1$, then for any neighborhood $\mathcal{N}$ of $M_\nu$:*

$$\lim_{t\longrightarrow+\infty} \mathbb{P}[X_t \in \mathcal{N}] = 1.$$

*Proof:* The argument follows from Theorem 3. Consider any neighborhood $\mathcal{N}$ of $M_\nu$. The continuity of $U_\nu$ shows that:

$$\exists \delta > 0 \quad \mathcal{N}^c \subset \{x \in \Gamma_G \, : U_\nu(x) > \min U_\nu + 2\delta\}.$$

Using the uniform convergence of $\mathbf{U}_{\nu,t}$, one can show that there exists $T_\delta > 0$ such that for all $t \geq T_\delta$:

$$\mathcal{N}^c \subset \{x \in \Gamma_G \, : \mathbf{U}_{\nu,t}(x) > \min \mathbf{U}_{\nu,t} + \delta\} =: \mathcal{N}^c_{\delta,t}$$

Hence,

$$\begin{aligned}
\mathbb{P}[X_t \in \mathcal{N}^c] &\leq \mathbb{P}[X_t \in \mathcal{N}^c_{\delta,t}] \\
&\leq \mathbb{P}[\mathbf{U}_{\nu,t}(X_t) > \min \mathbf{U}_{\nu,t} + \delta] \\
&= \int_{\Gamma_G} \mathbf{1}_{\mathbf{U}_{\nu,t}(x) > \min \mathbf{U}_{\nu,t}+\delta} \mathrm{d}n_t(x) \\
&= \int_{\Gamma_G} \mathbf{1}_{\mathbf{U}_{\nu,t}(x) > \min \mathbf{U}_{\nu,t}+\delta} \mathrm{d}\mu_{\beta_t}(x) + \int_{\Gamma_G} \mathbf{1}_{\mathbf{U}_{\nu,t}(x) > \min \mathbf{U}_{\nu,t}+\delta}[n_t(x) - \mu_{\beta_t}(x)]\mathrm{d}x \\
&\leq \mu_{\beta_t}\{\mathbf{U}_{\nu,t} > \min \mathbf{U}_{\nu,t} + \delta\} + 2d_{TV}(n_t, \mu_{\beta_t}) \\
&\leq \mu_{\beta_t}\{\mathbf{U}_{\nu,t} > \min \mathbf{U}_{\nu,t} + \delta\} + \sqrt{2J_t},
\end{aligned}$$

where we used the variational formulation of the total variation distance and the Csiszár-Kullback-Pinsker inequality $d_{TV} \leq \sqrt{2J}$ (see [46, 91, 112]). As soon as $\lim_{t \longrightarrow +\infty} J_t = 0$, we can conclude the proof observing that $\mu_{\beta_t} \{\mathbf{U}_{\nu,t} > \min \mathbf{U}_{\nu,t} + \delta\} \longrightarrow 0$ as $\beta_t \longrightarrow +\infty$. □

It can actually be proven (not shown here) that $\lim_{t \longrightarrow +\infty} \mathbb{P}[X_t \in \mathcal{N}] = 1$ *if and only if* the constant $b$ is chosen to be lower than $c^\star(U_\nu)^{-1}$. This means that this algorithm does not allow faster cooling schedules than the classical S.A. algorithm. Nevertheless, this is a positive result since this homogenized S.A. can be numerically computed quickly and easily on large graphs. Finally, we should consider this result to be theoretical. However, in practice, the simulation of this homogenized S.A. is performed during a finite horizon time and efficient implementations certainly deserve a specific theoretical study following the works of [42] and [119].

## 2.4 Numerical results

This experiment section presents some practical considerations on how to use our algorithm, and then describe our numerical results obtained on (i) social network subgraphs of *Facebook*, (ii) a graph of the *Parisian subway*, and (iii) a large citation subgraph of *zbMATH*. In order to compare the statistical accuracy and numerical complexity of our strategy with another reference algorithm, we also estimated the barycenter of the *Facebook* subgraphs, first with a bruteforce method to determine the true barycenter and therefore assess the accuracy of our method, and then using an adaptation of the algorithm of [72] to evaluate the performance of our method in comparison with the one of [72].

### 2.4.1 Practical insights

It is widely recognized that the algorithmic complexity of numerical strategies dealing with graphs can be an issue. The number of vertices and edges of real-life graphs can indeed be quite large. For instance, the *zbMATH* subgraph of Section 2.4.3 has 13000 nodes and approximately 48000 edges. In this context, it is worth justifying that the motion of $(X_t)_{t\geq 0}$ on the graph $\Gamma_G$ across the iterations of Algorithm 3 is reasonably demanding in terms of computational resources. We recall that this motion is driven by a Brownian motion (lines 4 to 6 of Algorithm 3) and an attraction towards the vertex $Y_{N_t^\alpha}$ (line 9 of Algorithm 3) at random times $(T_k)_{k\geq 0}$. The number of vertices and undirected edges with non-null weights in $\Gamma_G$ is also $N$ and $|E|$, respectively. Note finally that $N - 1 \leq |E| \leq N(N-1)/2$ if $\Gamma_G$ has a unique connected component.

**Neighborhood structure and sparse representation of the graph** A first computational issue that can arise when moving $X_t$ is to find all possible neighbors of a specific vertex $v$. This is indeed performed every time $X_t$ moves from one edge to another and directly depends on how $\Gamma_G$ is encoded in the memory.

(a) Encoding $\Gamma_G$ in a list of edges with non-null weights is common practice. In that case, the computer checks the vertex pairs linked by all edges to find those containing $v$, so the algorithmic cost is $2|E|$.

(b) A second classic strategy is to encode the graph in a connectivity matrix. In this case, the computer has to go through all the $N$ indexes of the columns representing $v$ to find the non-null weights, whith a cost $N$.

(c) We instead sparsely encode the graph in a list of lists: the main list has a size $N$ and each of its elements lists the neighbors of a specific vertex $v$. If the graph only contains edges with strictly positive weights, this strategy has a computational cost $N$, and in all of the other cases, the cost is $< N$. On average, the computational cost is $2\frac{|E|}{N}$.

Strategy (c) is particularly efficient for sparse graphs, where $|E| << N(N-1)/2$, which are common for the targeted applications. We therefore have chosen to handle the neighborhood structure of our graphs with this sparse list of list representation. For instance, $|E| = 4\,.10^3$ and $N(N-1)/2 = 1.24\,10^5$ on the smallest *Facebook* subgraph of Section 2.4.3, and $|E| = 4.8\,10^4$ and $N(N-1)/2 = 8.44\,10^7$ on the *zbMATH* subgraph of Section 2.4.3. Hence, following Strategy (c) for the smallest *Facebook* subgraph yields an average number of 16 operations (resp. 7.4 on the *zbMATH* subgraph) to choose a neighbor of any node, which is much more efficient than $8\,10^3$ (resp. $9.6\,10^5$) operations with Strategy (a) or 500 (resp. $1.3\,10^5$) operations with Strategy (b).

**Geodesic paths** Another potential issue with our strategy is that it seeks at least one optimal path between the vertices $X_t$ and $Y_{N_t^\alpha}$ at each jump time (line 9 of Algorithm 3). This may be efficiently done using a fast marching propagation algorithm (see, *e.g.*, [51] for details), where:

(i) the distance to $X_t$ is iteratively propagated on the whole graph $\Gamma_G$ until no more optimal distance to reach $X_t$ is updated, and

(ii) considering the shortest path between $X_t$ and $Y_{N_t^\alpha}$.

In this case, step (i) is particularly time-consuming and *cannot be reasonably performed at each step of Algorithm 3*. Fortunately, the algorithmic cost to compute the distance between all pairs of vertices is equivalent to that of computing step (i). We then compute these distances once and for all at the beginning of the computations and store the result in a $N \times N$ matrix in the RAM of the computer [3]. We can therefore very quickly use these results at each iteration of the algorithm and, in particular, deduce the useful part of the geodesic paths involved to travel from $X_t$ to $Y_{N_t^\alpha}$. The only limitation of this strategy is that the distance matrix can be memory-consuming. It can therefore be used on small to large graphs but not on huge graphs (typically when $N > 10^5$) on current desktops. An extension to deal with this scalability issue is a current subject of research, and we will therefore not describe applications of Algorithm 3 on huge graphs in this work.

**Practical calculation** Our method is described in Algorithm 3. We then briefly explain how to run this algorithm from a practical point of view. The simulation of some Poisson arrivals is straightforward and only Lines 5 and 9 (with Equation (2.5)) deserve some particular attention.

- Between two jumps in L.5, we discretize a Brownian motion on the edge where $(X_t)_{t\geq 0}$ is living and when it hits one node (the extrimity of the edge), we use a uniform reflexion among the neighbor of this node according to the neighborhood structure we encode by method (c) above.

- When a jumping time occurs, $t = T_{k+1}$ and we observe a new node $Y_{k+1}$. In that case, we obtain the geodesic distance between $X_t^-$ and $Y_{k+1}$ using the recorded matrix distance. This matrix also encodes the geodesic path between $X_t^-$ and $Y_{k+1}$ and we pick $X_t$ on this path at distance $\alpha_t^{-1}\beta_t d(X_t^-, Y_{k+1})$ from $X_t^-$.

[3] In our experiments, we used the *all-shortest-paths* function of the Python library NetworkX.

### 2.4.2 Parameter tuning

**Final time of computation** Several parameters influence the behavior of the simulated process $X_t$ in Algorithm 3. Some of them are directly introduced in the theoretical construction of the algorithm, *i.e.*, the intensity of the Poisson process $\alpha_t^\star$ or the temperature schedule $\beta_t^\star$. Other ones come from the practical implementation of the algorithm, *i.e.*, the maximal time $T_{\max}^\star$ up to which we generate $X_t$. The theoretical result given in Corollary 1 gives an upper bound for the probability of $X_t$ to be not too distant from the set of global minima. This bound depends on $\beta_t$ and $\alpha_t$ as well as on different characteristics of the graph such as its diameter and number of nodes. We then propose an empirical strategy for parameter tuning. For a graph $G$ with $N$ nodes and a diameter $\mathcal{D}_G$, our choice of the size of $T_{\max}^\star$ is driven by a linear relationship between the time of simulation needed and the number of nodes in the graph. Heuristically, the more nodes in the graph, the more we need to explore until the convergence is attained. We therefore choose

$$T_{\max}^\star = 100 + 0.1N.$$

**Cooling scheme** $(\beta_t)_{t\geq 0}$ Concerning now the temperature scheme, a good calibration of $t \longmapsto \beta_t$ should depend on the graph geometry. From a theoretical point of view (see Theorem 1) $\beta_t$ has to be chosen like $b\log(1+t)$ with $b$ small enough (lower than $c^\star(U_\nu)^{-1}$). Hence, it it necessary to find a reasonnable upper bound of $c^\star(U_\nu)$. It is straightforward to show that $c^\star(U_\nu) \leq \{\mathcal{D}_G\}^2$ (see Equation (2.6)), and in a same time understand that the elevation involved in Equation (2.6) should increase with the diameter of the graph. But it is much more difficult (and subject of our current investigation) to provide a tight upper bound of $c^\star(U_\nu)$. Such a bound is related to the underlying stochastic graph model for large graphs and the asymptotic study of $c^\star(U_\nu)$ is beyond the scope of this present work. In our simulations, we have found that a less stringent dependency of $c^\star(U_\nu)$ with the diameter provides some good results: $b$ is chosen linearly with $\{\mathcal{D}_G\}^{-1}$ instead of quadratically and we define $\beta_t^\star$ as:

$$\beta_t^\star = \frac{2}{\mathcal{D}_G}\log(t+1).$$

**Poisson intensity** $(\alpha_t)_{t\geq 0}$ We then choose the intensity of the Poisson process so that it will generate a reasonably large amount of $Y_n$ sampled with the discrete probability distribution $\nu$ at the end of the algorithm (one thousand below). More specifically, if we denote by $S^*$ the average number of jumping times between $T_{\max}^\star - 1$ and $T_{\max}^\star$ (last unit time of our simulation), we then need to solve

$$S^\star = \int_{T_{\max}^\star - 1}^{T_{\max}^\star} \alpha_s \mathrm{d}s.$$

In the same time, Theorem 3 shows that a minimal linear dependency of $\alpha_t$ with $t$ is required to guarantee the convergence of the method. In order to minimize the computational time, we then chose $\alpha_t = \lambda(t+1)$ and $\lambda$ is calibrated by solving the previous equation:

$$\lambda^\star = \frac{2S^\star}{2T_{\max}^\star + 1} \qquad \text{with} \qquad S^\star = 1000.$$

This strategy will be used later in Section 2.4.3 for reference parameters on graphs having different structures and sizes. Reference results can then be considered as obtained automatically.

**Number of used observations** The expectation of the number of observations used up to a given time $T$, depends only on $(\alpha_t)_{t\geq 0}$ and can be explicitly computed. Using the definition of $N_t^\alpha$ we have:

$$\mathbb{E}\left[N_t^\alpha\right] = \mathbb{E}\left[H_{\int_0^t \alpha_s \mathrm{d}s}\right] = \int_0^t \alpha_s \mathrm{d}s.$$

This implies that the number of observations used during a simulation with the set of parameters defined as above is of the order $\lambda^\star\left(1/2T_{\max}^{\star 2} + T_{\max}^\star\right)$. Taking the value of $\lambda^\star$ in the acount this roughly equal to $S^\star T_{\max}^\star/2$.

### 2.4.3 Results

This section presents results obtained on graphs of different sizes and using different parameters. We used the empirical methods of Section 2.4.2 to define default parameters and altered them to quantify the sensitivity of our algorithm to parameter variations.

Since the process $X_t$ lives on a quantum graph, its location at time $t$ is between two vertices, on an edge of $\Gamma_G$. However, our primary interest is to study the properties of the initial discrete graph and therefore, the output of the algorithm will be the vertex considered as the graph barycenter. To achieve this, we associate a frequency to each vertex. For a vertex $v$, this frequency is the portion of time during which $v$ was the closest vertex to the simulated process $X_t$. In our results, we consider frequencies computed over the last 10% of the iterations of the algorithm.

#### Facebook subgraphs

**Experimental protocol** We first tested our algorithm on three subgraphs of Facebook from the Stanford Large Network Dataset Collection[4]: **(FB500)** has 500 nodes and 4337 edges and contains two obvious clusters; **(FB2000)** has 2000 nodes and 37645 edges and fully contains (FB500); and **(FB4000)** has 4039 nodes and 88234 edges and fully contains (FB2000). For each of these subgraphs, we considered the probability measure $\nu$ as the uniform distribution over the graph's vertices and used a length of 1 for all edges. We also explicitly computed the barycenter of these graphs using an exhaustive search procedure (both for the discrete graphs and their continuous versions). We found that for the examples considered in this subsection, the Fréchet mean of the continuous and discrete version are the same. We are able to do these computations because we considered a simplified case where the distribution over the nodes is uniform (and thus explicit). For example, this exhaustive search procedure required approximately 2 hours for the **(FB4000)** subgraph. Nevertheless, it allowed us to compare our method with the one described in [72] since we can compute the ground-truth results with the bruteforce method on the **(FB4000)** subgraph.

We used the strategy of Section 2.4.2 to define default parameters adapted to each subgraph. We also tested different values for parameters $\beta$, $S$ and $T_{\max}$ in order to quantify their influence. We repeated our algorithm 100 times for each parameter set to evaluate the algorithm stability.

In the tables representing quantitative results, *Error* represents the number of times, out of 100, that the algorithm converged to a node different from the ground-truth barycenter. It is a rough indicator of the ability of the algorithm to locate the barycenter of the graph, that could be replaced by a measure of the average distance between the

[4] https://snap.stanford.edu/data/

last iterations of the algorithm and the ground-truth barycenter (not shown in this work).

For each subgraph and parameter set, column *Av. time* contains the average times in seconds for the barycenter estimation, keeping in mind that the Dijkstra algorithm was performed once for all before the 100 estimations. This preliminary computation requires approximately 1, 30 and 80 seconds, and 0.1, 0.6, and 1.5GB of memory on an Intel Core i7 Ubuntu laptop at 2.60GHz with 16GB memory.

**Effect of $\beta$** From a theoretical point of view, $(\beta_t)_{t\geq 0}$ should be chosen in relation to the constant $c^\star(U_\nu)$, which is unknown in practice. Therefore, the practical choice of $(\beta_t)_{t\geq 0}$ is a real issue to obtain a good behavior of the algorithm. Table 2.1 gives the results obtained on our algorithm with different values of $\beta_t$. We show the accuracy of the algorithm (percentage of error) and average time needed to compute until the ending time when the algorithm is launched on a private dedicated server.

| | FB500 | | | FB2000 | | | FB4000 | | |
|---|---|---|---|---|---|---|---|---|---|
| $\beta$ | Error | Med. Freq. | Av. time | Error | Med. Freq. | Av. time | Error | Med. Freq. | Av. time |
| $\frac{1}{4}\beta^*$ | 15 % | 0.6042 | 0.80 s | 28 % | 0.3519 | 5.26 s | 27 % | 0.3314 | 17.01 s |
| $\frac{1}{2}\beta^*$ | 2 % | 0.4184 | 0.83 s | 1 % | 0.7268 | 5.48 s | 5 % | 0.6534 | 17.58 s |
| $\beta^*$ | 0 % | 0.8008 | 0.91 s | 0 % | 0.9418 | 6.17 s | 1 % | 0.8913 | 18.98 s |
| $2\beta^*$ | 0 % | 0.8321 | 1.00 s | 0 % | 0.9892 | 7.71 s | 0 % | 0.9647 | 22.22 s |
| $4\beta^*$ | 0 % | 0.8233 | 1.13 s | 0 % | 0.9930 | 12.92 s | 0 % | 0.9824 | 26.02 s |
| $8\beta^*$ | 0 % | 0.7717 | 1.31 s | 0 % | 0.9750 | 12.23 s | 0 % | 0.9445 | 29.85 s |

Table 2.1: Experiments on the three Facebook subgraphs, while varying the values of $(\beta_t)_{t\geq 0}$.

We can observe that when $(\beta_t)_{t\geq 0}$ is too small, then the behavior of the algorithm is deteriorated, revealing the tendancy of the process $(X_t)_{0\leq t\leq T^\star_{\max}}$ to have an excessively slow convergence rate towards its local attractor in the graph. Roughly speaking, in such a situation, the process does not learn fast enough. When the value of $\beta$ is chosen in the range $[\beta^*; 2\beta^*]$, we can observe a really good behavior of the algorithm: it almost always locates the good barycenter in a quite reasonable time of computation (less then 20 seconds for the largest graph). Finally, we can observe in the column, Med. Freq., that in most of the last iterations of the algorithm (more than 80%), the process evolves around its estimated barycenter, so that the decision to produce an estimator is quite easy when looking at an execution of the algorithm.

**Effect of $S$ and $T_{\max}$** Table 2.2 gives the results obtained with our algorithm while using different values of $S$ and $T_{\max}$. As expected, we observe that increasing the ending time of simulation always improves the convergence rate (column Error in Table 2.2) of the algorithm towards the right node. The behavior of the algorithm is also improved by increasing the value of $S$, which quantifies the number of arrivals of nodes observed along the averaging procedure. Of course, the counterpart of increasing both $S$ and $T_{\max}$ is an increasing cost of simulation (see column Av. time).

As an illustration of the (small) complexity of the *Facebook* sub-graphs used for benchmarking our algorithm, we provide a representation of the FB500 graph in Figure 2.5. This representation has been obtained with the help of *Cytoscape* software[5] and is not a result of our own algorithm. In Figure 2.5, the red node is the estimated barycenter, which is also the ground-truth barycenter located by a direct exhaustive computation. The blue nodes are the “second rank" nodes visited by our method.

[5] http://www.cytoscape.org/

|  |  | FB500 |  |  | FB2000 |  |  | FB4000 |  |  |
|---|---|---|---|---|---|---|---|---|---|---|
| $S$ | $T_{\max}$ | Error | Med. Freq. | Av. time | Error | Med. Freq. | Av. time | Error | Med. Freq. | Av. time |
| $\frac{1}{2}S^*$ | $T^*_{\max}$ | 2 % | 0.7667 | 0.49 s | 0 % | 0.9344 | 3.43 s | 0 % | 0.8614 | 10.17 s |
| $S^*$ | $2T^*_{\max}$ | 0 % | 0.8049 | 1.85 s | 0 % | 0.9610 | 12.77 s | 0 % | 0.8970 | 38.72 s |
| $S^*$ | $4T^*_{\max}$ | 0 % | 0.8101 | 3.80 s | 0 % | 0.9677 | 26.35 s | 0 % | 0.9222 | 78.96 s |
| $2S^*$ | $T^*_{\max}$ | 1 % | 0.8345 | 1.74 s | 0 % | 0.9512 | 11.49 s | 0 % | 0.9062 | 36.17 s |
| $2S^*$ | $2T^*_{\max}$ | 0 % | 0.8361 | 3.53 s | 0 % | 0.9586 | 23.52 s | 0 % | 0.9121 | 73.47 s |
| $2S^*$ | $4T^*_{\max}$ | 0 % | 0.8423 | 7.23 s | 0 % | 0.9735 | 48.2s | 0 % | 0.9366 | 149.16 s |

Table 2.2: Experiments on the three Facebook subgraphs, while varying the values of $S$ and $T_{\max}$.

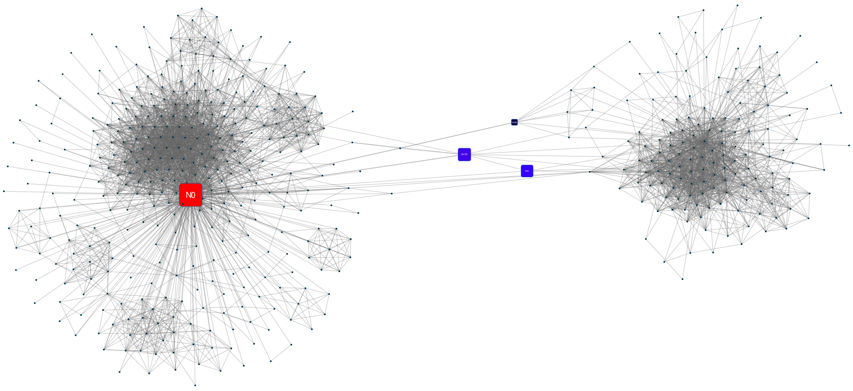


Figure 2.5: Region of interest presenting the results obtained on the FB500 graph.

### Parisian subway

In order to study a graph having a very intuitive interpretation of its barycenter, we also used our algorithm on the Parisian subway network. The 296 nodes and 353 weighted vertices of the graph were found in http://perso.esiee.fr/~coustyj/EnglishMorphoGraph/PS3.html, where the nodes obviously represent the metro stations while the length of each vertex is given by the average time spent to travel between two connected stations. In order to use a meaningful weights distribution on the nodes (*i.e.* metro stations), we also get from https://data.ratp.fr the metro stations attendance in 2015 and use a probability distribution $\nu$ proportionnal to this attendance.

Barycenter estimation was first performed using reference parameters, following the tuning strategy of Section 2.4.2. In order to assess the stability of our results we also run our algorithm using different parameters: ($i$) reference parameters, ($ii$) $10T^\star_{max}$ instead of $T^\star_{max}$ ($iii$) $2\beta^\star$ instead of $\beta^\star$, and ($iv$) $2S^\star$ instead of $S^\star$. Computations were moreover run 100 times for each parametrization. Using the reference parameters defined in Section 2.4.2, each barycenter estimation required about 0.6 seconds and a negligible amount of memory on an Intel Core i7 Ubuntu laptop at 2.60GHz. A typical result is given Fig. 2.6.

Subway station *Chatelet*, which is widely recognized as the central subway station in Paris, was estimated 98%, 100%, 100% and 97% of the times as the barycenter using the parameters ($i$), ($ii$), ($iii$), ($iv$), respectively. As shown in Table 2.3, we obtained subway stations which are either connected to *Chatelet* (i.e. *Hotel de Ville*) or close to *Chatelet* and with connections (i.e. *Bastille*, *Opéra* and *Madeleine*) in the rare cases where other stations were found.

Note finally that our graph could also be weighted with different properties than the travel time between connected subway stations and their attendance (for example with the average income of passengers). This could emphasize the subway stations in which specific groups of persons are likely to go through, which has many applications.

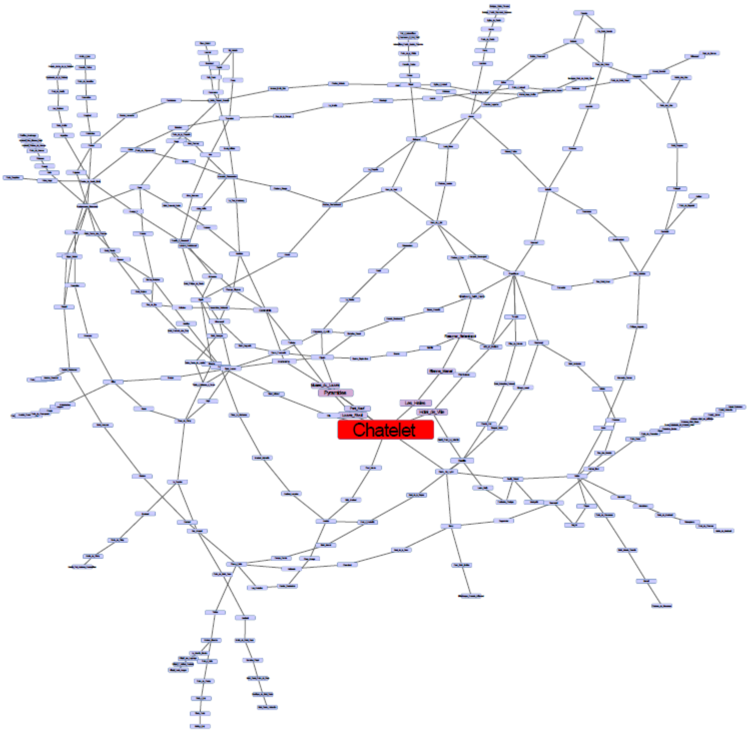


Figure 2.6: Barycenter estimate on the Parisian subway graph following the tuning strategy of Section 2.4.2.

| Parametrization | Chatelet | Bastille | Hotel de Ville | Opéra | Madeleine |
|---|---|---|---|---|---|
| $(i)$ | 98 | 0 | 1 | 1 | 0 |
| $(ii)$ | 100 | 0 | 0 | 0 | 0 |
| $(iii)$ | 100 | 0 | 0 | 0 | 0 |
| $(iv)$ | 97 | 2 | 0 | 0 | 1 |

Table 2.3: Estimated barycenters of the Parisian subway using 100 Monte-Carlo replication runs of Algorithm 3 for each of the four parametrizations of section 2.4.3.

### zbMATH subgraph

The zbMATH subgraph built from zbMATH[6] has been obtained by an iterative exploration of the co-authorship relationship in an alphabetical order. We thus naturally obtained a connected graph. This graph has been weighted by the complete number of citations obtained by each author, leading to the distribution probability $\nu \propto \sharp(\text{number of citations})$. Finally, all the edges in the graph are fixed to have a length of 1. This exploration was initialized on the entry of the first author's name (*i.e.*, S. Gadat) and we stopped the process when we obtained 13000 nodes (authors) on the graph. This stopping criterion in the exploration of the zbMATH database corresponds to a technical limitation of 40 GB memory required by the distance matrix obtained with the Dijkstra algorithm. In particular, this limitation and the starting point of the exploration induce an important bias in the community of authors used to build the subgraph from the zbMATH dataset: the researchers obtained in the subgraph are generally French and applied mathematicians. The graph is more or less focused on the following research themes: probability, statistics and partial differential equations. Consequently, the re-

[6] https://zbmath.org/authors/

sults provided below should be understood as an illustration of our algorithm and not as a bibliometric study!

Our experiments rely on the same choice of reference parameters $(\beta_t^\star)_{t\geq 0}$, $S^*$, $T^\star_{\max}$ and $\alpha^\star_T$ as above, *i.e* those following the strategy of 2.4.2. As in section 2.4.3, we assessed the stability of our algorithm by running our algorithm using different parameters: $(i)$ reference parameters $(ii)$ $2T^*_{max}$ instead of $T^*_{max}$ $(iii)$ $2\beta^*$ instead of $\beta^*$, and $(iv)$ $2S^*$ instead of $S^*$. Computations were moreover run 10 times for each parametrization. Using our reference parameters, each barycenter estimation required about 90 minutes and 40GB memory on an Intel Xeon E5-2660 Debian server at 2.60GHz with 128GB memory.

In Figure 2.7, we present a representation of the subgraph obtained with *Cytoscape* software, and a zoom on a Region Of Interest (ROI for short) in Figure 2.7. Again, the main nodes visited by our algorithm are represented with a red square and the size of the used square is larger when the node is frequently visited (using reference parameters here). Note that due to the large size of the zbMATH subgraph, computing its ground-truth barycenter was however technically impossible. According to the results obtained on the *Facebook* subgraphs and the *Parisian subway graph*, we then assume that the largest red square is the barycenter of the zbMATH subgraph, *i.e.* the most frequent node over the last 10% of the iterations.

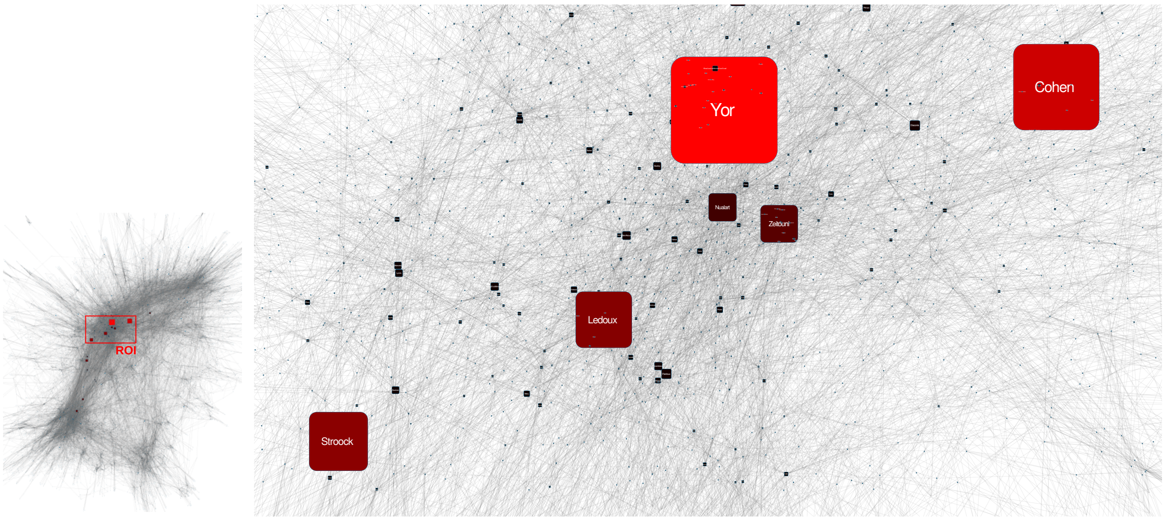


Figure 2.7: Left: General overview of the zbMATH subgraph extracted for our experiments, containing approximately 13000 nodes. Right: Region of interest presenting the main results obtained on the zbMATH subgraph.

Although the large graph size, barycenter estimates were stable with respect to the Monte Carlo runs of our algorithm. We have first produced some boxplots for each of the main authors located in the subgraph, according to the occupation measure of the process over the last 10% of the iterations with the 10 Monte-Carlo replications run using the parameters $(i)$, *i.e.* the reference parameters. These “violin" plots are represented in Figure 2.8. The node *M. Yor* was also generally estimated as the graph barycenter using the different parametrizations: He was estimated 7, 8, 5 and 5 times as the barycenter using the 10 Monte Carlo run with parametrizations $(i)$, $(ii)$, $(iii)$ and $(iv)$, respectively. As shown in Table 2.4, other estimated barycenters are close to *M. Yor* in the graph and have a strong bibliography too. The proposed algorithm therefore seems to produce reliable conclusions concerning the top nodes visited all along the ending iterations, even on large social networks. Nevertheless, it appears to be necessary to extend our investigations in order to obtain a scalable method for handling larger graphs.

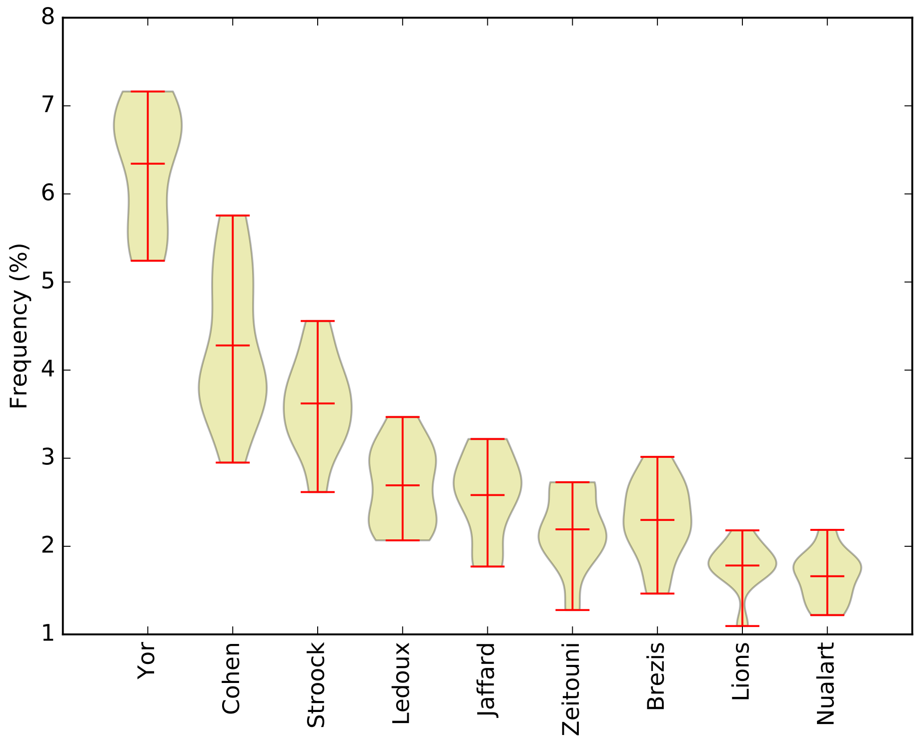


Figure 2.8: “Violin" plot of the occupation measure of the algorithm on the zbMATH subgraph for 10 MC replications. The average frequency is located in the middle of the “violin" plot, although the minimal and maximal values are shown in the extremity of the representation.

| Parametrization | M. Yor | S. Cohen | D. W. Stroock | P.-L. Lions | I. A. Ibragimov | O. Zeitouni |
|---|---|---|---|---|---|---|
| *(i)* | 7 | 1 | 0 | 1 | 1 | 0 |
| *(ii)* | 8 | 2 | 0 | 0 | 0 | 0 |
| *(iii)* | 5 | 3 | 2 | 0 | 0 | 0 |
| *(iv)* | 5 | 2 | 2 | 0 | 0 | 1 |

Table 2.4: Estimated barycenters of the zbMATH subgraph using 10 Monte-Carlo replication runs of Algorithm 3 for each of the four parametrizations of section 2.4.3.

### 2.4.4 Comparison with Gutjahr and Pflug [72]

#### Adaptation of the method

Finally, we also estimated the barycenter of the Facebook subgraphs using an adaptation of the algorithm of [72]. This algorithm is similar to the one described in Algorithm 1, the main difference being that at each step $k$ the true value of the potential $U$ is replaced by an estimation with $n_k$ independent observations $(Y_i)_{i\geq 0}$ of law $\nu$. These mini-batch samples obtained by Monte-Carlo simulations then produce an estimate at iteration $k$:

$$\widetilde{U}_k(x) = \frac{1}{2}\sum_{i=1}^{n_k} d^2(x, Y_i). \tag{2.19}$$

This algorithm was created for finite discrete spaces, with a symetric neighborhood structure in [72]. In a more general situation of unbalanced neighborhood structure, the transition is then induced by the Markov kernel proposition $L(.,.) : V \times V \to [0,1]$, as mentioned in the description of Algorithm 1. In order to adapt it to a graph, it seems natural to consider the Markov kernel corresponding to the intrinsic structure of the graph itself. This means that $L(x,y) > 0$ if and only if $x$ and $y$ are neighbors. Furthermore in this case $L(x,y) = 1/S_x$, where $S_x$ is the number of neighbors of a vertex $x$. In other words this means that when the algorithm is located at a state $X_k$, it proposes a new state, $x'$, uniformly among the neighbors of $X_k$. This state is then accepted with probability

$$p_k = 1 \wedge \left\{ e^{T_k^{-1}[\widetilde{U}(X_k)-\widetilde{U}(x')]} \frac{L(x',X_k)}{L(X_k,x')} \right\},$$

where the true value of $U$ is then replaced by the estimation produced by Equation (2.19) in the Metropolis-Hastings acceptation rate.

Before describing the numerical results, we emphasize that actually no result of convergence is known for the method of [72] for general noise models on $U$, since [72] only deals with Gaussian noise. Moreover, the good behaviour of the mini-batch strategy requires that at least $n_k \geq O(k^2)$ samples are used at each step. Hence, at iteration $N$, the number of observations $Y$ used is of the order $O(N^3)$.

### Results

In order to fairly compare Algorithm 3 with the one of [72], we tuned the number of iterations $N$ of [72] so that both strategies required approximately the same number of observations $Y_n$ and had therefore a similar algorithmic cost. We therefore used $\{T^\star\}^{2/3}$ iterations (60, 76 and 90 for FB500, FB2000 and FB4000, respectively). We also used the same reference $\beta^*$ values for both methods.

| | FB500 | | FB2000 | | FB4000 | |
|---|---|---|---|---|---|---|
| $\beta$ | Error | Comp. T. | Error | Comp. T. | Error | Comp. T. |
| $4\beta^*$ | 48 % | 0.67 s | 40% | 5.4 s | 71 % | 14.06 s |
| $8\beta^*$ | 54 % | 0.77 | 29 % | 5.42 | 65% | 13.75 s |
| $32\beta^*$ | 45 % | 0.73 | 45 % | 5.11 | 72% | 14.54 s |

Table 2.5: Experiments on the three Facebook sub-graphs using [72], while varying the value of $(\beta_t)_{t\geq 0}$, using the same amount of observations $Y$ as Algorithm 3. Equivalent results with Algorithm 3 are given Table 2.1.

**FB subgraph**. First, we observe that the computing time is not influenced by the variation of $\beta$, as opposed to Algorithm 3. This implies that for large values of $\beta$ this algorithm is a bit faster than Algorithm 3. However, one can easily suspect that the number of iterations $\{T^\star\}^{2/3}$ is too small and that the algorithm of [72] does not have time to converge. Therefore, in Table 2.6, we present results on the graph FB2000 with some highest number of iterations. We can easily see that the number of used observations, and therefore the computational time, increase very rapidly with respect to the number of iterations, which

| Iterations | Error | Av. Time | $\beta$ | Observations |
|---|---|---|---|---|
| 100 | 21 % | 11.66 s | 8.0 | $33\,10^4$ |
| 100 | 34 % | 11.50 s | 16.0 | $33\,10^4$ |
| 200 | 20 % | 136.17 s | 8.0 | $26\,10^5$ |
| 200 | 12 % | 134.22 s | 16.0 | $26\,10^5$ |
| 400 | 3 % | 1077.78 s | 8.0 | $21\,10^6$ |

Table 2.6: Experiments on the FB2000 sub-graph using [72], while varying the number of iterations and $(\beta_t)_{t\geq 0}$.

is a common feature of mini-batch strategies. Even though the number of iterations is not very high, one can see that for a reasonnably low error rate, the computational time needs to be rather large, and in particular much larger than the one of Algorithm 3.

**Parisian subway graph**. We also estimated the barycenter of the Parisian Metro using the algorithm of [72] with time distances between stations in minutes. We consider the set of parameters described in 2.4.3 and use the equivalent number of iterations with respect to the number used observations. Table 2.7 shows that the algorithm of [72] performs better on the Parisian subway graph than on the Facebook sub-graphs. This is to be expected, since the degree distribution of this graph is more homogeneous than in the social-network case, and thus the probability of acceptance is more influenced by the fluctuation of $U$ than by the Markov kernel proposition (see next paragraph, Section 2.4.4). However, from a computational time, our method is still faster than the one of [72].

| Parametrization | Error | Av.time | Iterations |
|---|---|---|---|
| $(i)$ | 16 % | 0.93 s | 57 |
| $(ii)$ | 2 % | 8.80 s | 124 |
| $(iii)$ | 3 % | 1.03 s | 57 |
| $(iv)$ | 4 % | 2.03 s | 72 |

Table 2.7: Estimated barycenters of the Parisian subway using 100 Monte-Carlo replication runs of Algorithm [72] for each of the four parametrizations of section 2.4.3.

### Bias induced by the neighborhood structure

One reason why the algorithm of [72] is clearly outperformed in the FB datasets may be the fact that the kernel proposition influences not just the proposal, but also the probability of acceptance. This is especially the case when the ratio $\frac{L(x',X_k)}{L(X_k,x')}$ is very high, or very low. In the case of social networks, this is often the case, because the variance of the degree of the vertices is high. For example, in the FB2000 sub-graph, the method of [72] oscillates between two nodes 1 and 2 with a very close energy level: $U_\nu(1) \leq U_\nu(2) = U_\nu(1) + \delta$ and node 1 possesses much more neighbors than node 2. Therefore, the transition 1 → 2 happens with a probability of the order $n_1/n_2 e^{-\delta T_k^{-1}}$ at time $k$. The number of iterations $k$ needed to reject this transition with probability $1-\epsilon$ should be such that

$$\beta \log(1+k) \geq \frac{\log(n_1/n_2 \epsilon^{-1})}{\delta},$$

and it is immediate to see that when $n_1 >> n_2$, we have to wait a long time before this transition becomes unlikely. Oppositely, the transition 2 → 1 appears to difficult in this settings while 1 is the minimizer of $U_\nu$. To sum-up, the algorithm of [72] needs more time in order to overcome the bias induced by the neighborhood structure of the graph.

## 2.5 Concluding remarks

In this paper, we have introduced a global optimization procedure that makes it possible to estimate the center of mass of a weighted graph when some i.i.d. nodes $(Y_k)_{k\geq 1}$ of law $\nu$ are observed sequentially, even if $\nu$ is unknown. Our algorithm uses an stochastic homogenization strategy coupled with simulated annealing and appears to work well on reasonably large graphs. For example, it requires only few seconds on FB subgraphs with an almost perfect recovery rate.

This algorithm is tightly linked to the size of the constant $c^\star(U_\nu)$ introduced in Equation (2.6), which defines a maximal size for the function $\beta_t$ at time $t$. This constant is classically related to the entropic barrier that separates any couple of basins of attraction of local minima of $U_\nu$. From a theoretical point of view, if $\beta_t$ is too large, then the homogenized S.A. may fail to converge to the good target point, while when $\beta_t$ is too small, the convergence of the algorithm may be too slow. It is therefore important, from a practical point of view, to relate $c^\star(U_\nu)$ to some tractable geometrical quantities that describe $G$ and in particular it is of primary interest to find a not too large upper bound of $c^\star(U_\nu)$. We can expect that $c^\star(U_\nu)$ increases when the size of the graph increases, and this intuition is translated in our setting $\beta_t^\star \propto \{\mathcal{D}_G\}^{-1}\log(1+t)$ through the diameter of the graph $\mathcal{D}_G$. Nevertheless, we do not have at the moment any theoretical justification of why such a choice is reasonable for network analysis. It is certainly necessary to introduce a random network model (for example the stochastic block model that may describe $K$ communities of $n$ nodes) and understand the size of $c^\star(U_\nu)$ when $n$ increases. Understanding the (asymptotic) behaviour of the spectral gap when the size of the graph is growing is beyond the scope of our paper, and a current subject of our investigations.

Another possible theoretical and practical development concerns the noise dynamics involved by our Brownian motion. Indeed, it would be interesting to use a piecewise deterministic Markov process, instead of a Brownian motion, to create the minimal needed random agitation. This development has been recently proposed by [99] for S.A. procedures in Euclidean settings, and in [25] with Zig-Zag processes for posterior Bayesian probability sampling.

Lastly, we should also say that our algorithm still requires long run of computations for very large graphs. To improve this, a future development will propose to first use a preliminary clustering step (for which efficient sub-optimal algorithms exist) before running our homogenized S.A.

## 2.6 Regularity issues

In this section we deal with all the issues raised by the irregularity of $U_\nu$ and the Neumann conditions. We start by stating precisely what we mean be a smooth version of the distance $d$. We then explain why (2.18) holds, namely why $f_t(x) = \log n_t/\mu_{\beta_t}$ belongs to the generator's domain.

### Regularized version of $d$

Let $\mathbf{d} : \mathbb{R}_+ \times \Gamma \times V$ be such that the following conditions hold:

Reg The function is smooth on $x$:

$$\forall t > 0 \quad \forall y \in V \quad \mathbf{d}_t(\cdot, y) \in \mathcal{C}^2(\Gamma_G)$$

Neu The directional derivative are always null :

$$\forall v \in V,\ \forall e \in E, e \sim v \quad d_e[\mathbf{d}_t(\cdot, y)](v) = 0.$$

Uni When $t$ goes to infinity $\mathbf{d}_t$ converges uniformly to the restriction of the distance $d$ to $\Gamma_G \times V$:

$$\forall \delta > 0 \ \exists\ T > 0 \text{ such that } \forall t \geq T,\ \forall x \in \Gamma,\ \forall y \in V \quad |\mathbf{d}_t(x, y) - d(x, y)| < \delta.$$

### Densities in the generator's domain

To see that this set of conditions is sufficient to ensure that (2.18) holds, consider $\widetilde{m}_t(x,y)$ the density of the Markov process $(X_t, Y_t)_{t\geq 0}$ (given by Theorem 2), with respect to $\widetilde{\lambda}_t$, its invariant measure at time $t$:

$$\widetilde{m}_t(x,y) = m_t(x,y)/\widetilde{\lambda}_t(x,y).$$

One can show using the martingale problem formulation that $\widetilde{m}_t \in \mathcal{D}(\mathcal{L})$. This is independent of the regularization and is true only because we consider the density with respect to the invariant measure. Additional conditions are imposed on the drift to ensure that the invariant measure is in the domain:

$$\widetilde{\lambda}_t \in \mathcal{D}(\mathcal{L}).$$

In particular, the regularity of the drift ensures that, for all $y \in V$, the density in $x$ corresponding to the invariant measure is smooth, $\widetilde{\lambda}_t(\cdot, y) \in \mathcal{C}^2(\Gamma_G)$, while condition Neu implies that its directional derivatives are all null. Now $m_t(x,y)$, the distribution of $(X_t, Y_t)$ with respect to the normalized Lebesgue measure, can be written as the product of two functions belonging to $\mathcal{D}(\mathcal{L})$, and thus it is also in the domain.

As for $n_t$, the density of the marginal law of $(X_t)_{t\geq 0}$ at time $t$, it can be written as:

$$n_t(x) = \int m_t(x,y)\mathrm{d}y \quad \text{and thus} \quad n_t \in \mathcal{D}(\mathcal{L}) \tag{2.20}$$

Since for all $y \in V$ and all $t$, $\mathbf{d}_t(\cdot, y)$ are $\mathcal{C}^2(\Gamma_G)$, the density associated to the Gibbs measure $\mu_{\beta_t} \sim e^{-\beta_t \mathbf{U}_{\nu,t}(x)}$ are also $\mathcal{C}^2(\Gamma_G)$. A direct computation can verify that the Neu condition on $\mathbf{d}_t$, implies that the gluing conditions (4) are satisfied by $\mu_{\beta_t}$, for all $t$. Of course this is not necessarily true for the Gibbs measure associated to $U_\nu$. Using (2.20) we can conclude that (2.18) holds:

$$\Gamma_G \times V \ni (x,y) \longmapsto \log \frac{n_t(x)}{\mu_{\beta_t}(x)} \in \mathcal{D}(\mathcal{L}).$$

### Dealing with $\mathbf{U}_{\nu,t}$

Denote $\mathcal{D}_{G,t}$ the upper bound of $\mathbb{D}$ at time $t$:

$$\mathcal{D}_{G,t} = \sup\{ \ |\ \mathbf{d}_t(x,y)| \ : \ x \in \Gamma_G \text{ and } y \in V\}. \tag{2.21}$$

We use this notation to emphasize the fact that this quantity, especially for large times, is close to the graph's diameter. In particular, Uni implies:

$$\lim_{t\to\infty} \mathcal{D}_{G,t} = \mathcal{D}_G.$$

Moreover, once we have a function $\widetilde{\mathbf{d}}_t$ that respects Uni, for all $\delta > 0$ there exists a $T > 0$ such that if $\mathbf{d}_t = \widetilde{\mathbf{d}}_{t+T}$ then:

$$\|d - \mathbf{d}_t\|_\infty < \delta \quad \text{and thus} \quad |\mathcal{D}_{G,t} - \mathcal{D}_G| < \delta,$$

where the infinity norm is considered over $\Gamma_G \times V$ (we make a slight abuse and use the same notation for the distance defined on $\Gamma_G \times \Gamma_G$ and its restriction to $\Gamma_G \times V$). If, in addition, Neu and Reg hold for $\widetilde{\mathbf{d}}_t$, then they also hold for $\mathbf{d}_t$. This is why $\mathcal{D}_{G,t}$ can be considered as being close to the graph's diameter $\mathcal{D}_G$ at all times.

At time $t$ the gap between $\mathbf{U}_{\nu,t}$ and $U_\nu$ is controlled by the graph's diameter and the gap between $\mathbf{d}_t$ and $d$:

$$\|\mathbf{d}_t - d\|_\infty < \delta \quad \Rightarrow \quad \|\mathbf{U}_{\nu,t} - U_\nu\|_\infty < (2\mathcal{D}_G + \delta)\,\delta.$$

In order to deal with the relative entropy $J_t$, we compute its derivative and try to obtain a differential equation that implies its convergence towards 0. The computations imply, among others, the time derivative of the function $\mathbf{U}_{\nu,t}$. With that in mind, let $\delta : \mathbb{R}_+ \to \mathbb{R}$, be a function that controls uniformly the time derivative of $\mathbf{U}_{\nu,t}$, for all $(t,x,y) \in \mathbb{R}_+ \times \Gamma_G \times V$:

$$|\partial_t \, \mathbf{U}_{\nu,t}(x,y)| \leq \delta_t \tag{2.22}$$

We also need an upper bound on the second derivative of $\mathbf{U}_{\nu,t}$ in $x$ and thus let $\varphi_t$ be such that, for all $(t,x,y) \in \mathbb{R}_+ \times \Gamma_G \times V$ :

$$|\partial_x^2 \mathbf{U}_{\nu,t}(x,y)| \leq \varphi_t \tag{2.23}$$

**Proposition 2.** *There exists a function $\mathbf{d} : \mathbb{R}_+ \times \Gamma_G \times V \to \mathbb{R}$ continuously derivable in time, complying with the set of conditions* Uni, Neu *and* Reg *and such that when $t$ goes to infinity:*

$$\delta_t \sim (\log^4 t)/t \qquad \textit{and} \qquad \varphi_t \sim \log t$$

*where $\delta_t$ is defined in (2.22) and $\varphi_t$ in (2.23). Moreover, for all $(t,x,y) \in \mathbb{R}_+ \times \Gamma_G \times V$:*

$$|\partial_x \mathbf{d}_t(x,y)| \leq 6 \tag{2.24}$$

To impose upper bounds on the partial derivatives of $U_\nu$ is necessary in the convergence proof of the entropy. We will see that the upper bound of the time derivative is related to the temperature schedule $\beta_t = b\log(t+1)$, namely $\delta_t$ has tot be such that for large times:

$$\delta_t = o\left(t^{-bc^\star(U_\nu)}\right).$$

The bound on the first derivative seems quite arbitrary, and such a strong condition is not necessary, but having a finite uniform bound facilitates the proof. The constant 6 can be easily obtained in our case and it could probably be improved, but doing so would have no influence on our result. In particular, the upper bound on $|\partial_x \mathbf{d}_t|$ implies that:

$$|\partial_x \mathbf{U}_{\nu,t}(x,y)| \leq 12\mathcal{D}_{G,t}.$$

The proof of this proposition follows from the construction of such a function using polynomial interpolation. It is thus rather technical and in itself does not bring a better understanding of our method. This is why we postpone it to Appendix 2.9.

## 2.7 Proof of the main result (Theorem 3)

We establish a differential inequality that will imply the convergence of $J_t$. The computations actually lead us to a system of two differential inequalities. We therefore introduce another quantity $I_t$ that measures the average closeness (w.r.t. $x$) of the conditional law of $y$ given $x$ at time $t$ to $\nu$, defined as:

$$I_t := \int KL(m_t(.|x)||\nu)\mathrm{d}n_t(x) = \int \left[\int \log \frac{m_t(y|x)}{\nu(y)} m_t(y|x)dy\right].\mathrm{d}n_t(x) \tag{2.25}$$

The next proposition links the evolution of $J_t$ (in terms of an upper bound of $\partial_t J_t$) with the spectral gap of $\mu_{\beta_t}$ over $\Gamma_G$, the upper bound of $\mathbf{d}_t$, $\mathcal{D}_{G,t}$ and the divergence $I_t$.

### 2.7.1 Study of $\partial_t J_t$

**Proposition 3.** *Let $c^\star(U_\nu)$ be the term defined in Equation (2.6) and $C_{\Gamma_G}$ be the constant given in Proposition 6. We then have:*

$$\partial_t J_t \leq \mathcal{D}_{G,t}^2 \beta_t' + \beta_t \delta_t + 2(12\mathcal{D}_{G,t}\beta_t)^2 I_t - \frac{e^{-c^\star(\mathbf{U}_{\nu,t})\beta_t}}{C_{\Gamma_G}(1+\beta_t)} J_t.$$

Proof: First, Proposition 1 provides the differentiability of $(t,x) \longrightarrow n_t(x)$ on $\mathbb{R}_+ \times \Gamma_G$, which is a consequence of the non-degeneracy of the Brownian motion on $\Gamma_G$ involved in Equation (2.4). In the meantime, $(t,x) \longrightarrow \mu_{\beta_t}(x)$ is positive and differentiable on $\Gamma_G$. Moreover, the time derivatives of $n_t$ and $\mu_{\beta_t}$ are continuous w.r.t. $x$. The graph $\Gamma_G$ being a finite union of segments, these smoothness results lead to the integrability of $\partial_t \log(n_t/\mu_{\beta_t})$, and of $\log(n_t/\mu_{\beta_t})\partial n_t$ over $\Gamma_G$.

Second, we compute the derivative of $J_t$ and separately study each of the three terms:

$$\partial_t J_t = \underbrace{\int_{\Gamma_G} \partial_t\{\log(n_t(x))\}\mathrm{d}n_t(x)}_{J_{1,t}} - \underbrace{\int_{\Gamma_G} \partial_t\{\log(\mu_{\beta_t}(x))\}\mathrm{d}n_t(x)}_{J_{2,t}} + \underbrace{\int_{\Gamma_G} \log\left[\frac{n_t(x)}{\mu_{\beta_t}}\right] \partial_t n_t(x)\mathrm{d}x}_{J_{3,t}}. \tag{2.26}$$

**Study of $J_{1,t}$:** This term is easy to deal with:

$$\begin{aligned} J_{1,t} &= \int_{\Gamma_G} \partial_t\{\log(n_t(x))\}n_t(x) = \int_{\Gamma_G} \frac{\partial_t\{n_t(x)\}}{n_t(x)} n_t(x) \\ &= \int_{\Gamma_G} \partial_t\{n_t(x)\}dx = \partial_t\left\{\int_{\Gamma_G} n_t(x)dx\right\} = \partial_t\{1\} = 0. \end{aligned} \tag{2.27}$$

**Study of $J_{2,t}$:** Using the definition of $\mu_{\beta_t}$ given in Equation (2.13), we obtain for the second term:

$$\begin{aligned} J_{2,t} &= -\int_{\Gamma_G} \partial_t\{\log(\mu_{\beta_t}(x))\}\mathrm{d}n_t(x) = -\int_{\Gamma_G} \partial_t\{-\beta_t \mathbf{U}_{\nu,t}(x) - \log Z_{\beta_t}\}\mathrm{d}n_t(x) \\ &= \int_{\Gamma_G} [\beta_t' \mathbf{U}_{\nu,t}(x) + \beta_t \partial_t \mathbf{U}_{\nu,t}(x)]\,\mathrm{d}n_t(x) + \frac{\partial_t\{Z_{\beta_t}\}}{Z_{\beta_t}}. \end{aligned}$$

According to the definition of $Z_{\beta_t}$ given in Equation (2.14), we have:

$$\begin{aligned} \frac{\partial_t\{Z_{\beta_t}\}}{Z_{\beta_t}} &= Z_{\beta_t}^{-1}\partial_t\left\{\int_{\Gamma_G} e^{-\beta_t \mathbf{U}_{\nu,t}(x)}\mathrm{d}x\right\} \\ &= -\int_{\Gamma_G} [\beta_t' \mathbf{U}_{\nu,t}(x) + \beta_t \partial_t \mathbf{U}_{\nu,t}(x)]\, \frac{e^{-\beta_t \mathbf{U}_{\nu,t}(x)}}{Z_{\beta_t}} \mathrm{d}x \\ &= -\int_{\Gamma_G} [\beta_t' \mathbf{U}_{\nu,t}(x) + \beta_t \partial_t \mathbf{U}_{\nu,t}(x)]\, \mu_{\beta_t}(x)\mathrm{d}x. \end{aligned}$$

Hence, we obtain:

$$J_{2,t} = \beta_t' \int_{\Gamma_G} \mathbf{U}_{\nu,t}(x)[n_t(x) - \mu_{\beta_t}(x)]\mathrm{d}x + \int_{\Gamma_G} \beta_t \partial_t \mathbf{U}_{\nu,t}(x)[n_t(x) - \mu_{\beta_t}(x)]\mathrm{d}x$$

For all $t$, $\mathbf{U}_{\nu,t}$ is a continuous function defined on a compact space. We therefore have: $\int_{\Gamma_G} \mathbf{U}_{\nu,t}(x)\mathrm{d}n_t(x) \leq \mathcal{D}_{G,t}^2$. At the same time, the absolute value of the time derivative $\partial_t \mathbf{U}_{\nu,t}(x,y)$ is uniformly controlled by $\delta_t$ and thus:

$$\int_{\Gamma_G} \beta_t\, |\partial_t \mathbf{U}_{\nu,t}(x)|\mathrm{d}n_t(x) \leq \beta_t \delta_t$$

The same inequality holds using the measure $\mu_{\beta_t}$ so that:

$$|J_{2,t}| \leq \mathcal{D}^2_{G,t}\beta'_t + \beta_t\delta_t. \tag{2.28}$$

**Study of $J_{3,t}$:** The last term $J_{3,t}$ involves the backward Kolmogorov equation. First, since $n_t$ is the marginal law of $X_t$, we have: $n_t(x) = \int m_t(x,y)\mathrm{d}y$.

Using the backward Kolmogorov equation for the Markov process $(X_t, Y_t)_{t\geq 0}$ and the Fubini theorem, we have, for any function twice differentiable function $f_t : x \in \Gamma_G \longmapsto \mathbb{R}$:

$$\begin{aligned}\int_{\Gamma_G} f_t(x)\partial_t\{n_t(x)\}\mathrm{d}x &= \int_{\Gamma_G} f_t(x)\partial_t\left\{\int_V m_t(x,y)\mathrm{d}y\right\}\mathrm{d}x = \int_{\Gamma_G}\int_V f_t(x)\partial_t\{m_t(x,y)\}\mathrm{d}x\mathrm{d}y\\ &= \int_{\Gamma_G}\int_V \mathcal{L}_t(f_t)(x)m_t(x,y)\mathrm{d}x\mathrm{d}y\\ &= \int_{\Gamma_G}\int_V [\mathcal{L}_{1,t}+\mathcal{L}_{2,t}](f_t)(x)m_t(x,y)\mathrm{d}x\mathrm{d}y,\end{aligned}$$

where $\mathcal{L}_{1,t}$ and $\mathcal{L}_{2,t}$ are defined in Equation (2.10) and Equation (2.11). Since the function $f_t$ is independent of $y$, we have $\mathcal{L}_{1,t}(f_t) = 0$. For the part corresponding to $\mathcal{L}_{2,t}$, we have:

$$\begin{aligned}&\int_{\Gamma_G}\int_V \mathcal{L}_{2,t}(f_t)(x)m_t(x,y)\mathrm{d}x\mathrm{d}y\\ &= \int_{\Gamma_G}\int_V \left[\frac{1}{2}\Delta_x f_t(x) - \beta_t\nabla_x \mathbf{U}_{y,t}(x)\nabla_x f_t(x)\right] m_t(x,y)\mathrm{d}x\mathrm{d}y\\ &= \int_{\Gamma_G}\frac{1}{2}\Delta_x f_t(x)n_t(x)\mathrm{d}x - \beta_t\int_{\Gamma_G}\int_V \nabla_x\mathbf{U}_{y,t}(x)\nabla_x f_t(x)n_t(x)m_t(y|x)\mathrm{d}x\mathrm{d}y.\end{aligned}$$

because $n_t$ is the marginal distribution of $X_t$ and $m_t(y|x) \times n_t(x) = m_t(x,y)$.

Thus, for any function $f$ such that for all $t$, $f_t(\cdot)$ respects the Neumann conditions defined in (4) and is in $\mathcal{C}^2(\Gamma_G)$, using the operator introduced in Equation (2.17), we have:

$$\int_{\Gamma_G} f_t(x)\partial_t n_t(x)\mathrm{d}x = \int_{\Gamma_G}\widetilde{\mathcal{L}}_{2,t}(f_t)(x)n_t(x)\mathrm{d}x.$$

This point of the proof demands that condition (2.18) holds and $\mathbf{U}_{\nu,t}$ was defined precisely to ensure that. Hence, replacing $f_t(x)$ by $\log\frac{n_t(x)}{\mu_{\beta_t}(x)}$, we obtain:

$$J_{3,t} = \int_{\Gamma_G}\log\frac{n_t(x)}{\mu_{\beta_t}}\partial_t n_t(x) = \int_{\Gamma_G}\widetilde{\mathcal{L}}_{2,t}\left[\log\frac{n_t(x)}{\mu_{\beta_t}(x)}\right]n_t(\mathrm{d}x).$$

Since $\mu_{\beta_t}$ is the invariant distribution of $\widehat{\mathcal{L}}_{2,t}$ (see Equation (2.16)), it is natural to insert $\widehat{\mathcal{L}}_{2,t}$:

$$\begin{aligned}J_{3,t} &= \int_{\Gamma_G}\widetilde{\mathcal{L}}_{2,t}\left[\log\frac{n_t(x)}{\mu_{\beta_t}(x)}\right]n_t(\mathrm{d}x)\\ &= \int_{\Gamma_G}\widehat{\mathcal{L}}_{2,t}\left[\log\frac{n_t(x)}{\mu_{\beta_t}(x)}\right]n_t(\mathrm{d}x) - \underbrace{\int_{\Gamma_G}\left(\widehat{\mathcal{L}}_{2,t}-\widetilde{\mathcal{L}}_{2,t}\right)\left[\log\frac{n_t(x)}{\mu_{\beta_t}(x)}\right]n_t(\mathrm{d}x)}_{:=\kappa_t}.\end{aligned} \tag{2.29}$$

Since $\widehat{\mathcal{L}}_{2,t}$ is a diffusion operator and $\mu_{\beta_t}$ is its invariant measure, it is well known (see, *e.g.*, [21]) that the action of $\widehat{\mathcal{L}}_{2,t}$ on the entropy is closely linked to the Dirichlet form in the following way:

$$\int_{\Gamma_G} \widehat{\mathcal{L}}_{2,t}\left[\log \frac{n_t(x)}{\mu_{\beta_t}(x)}\right] n_t(\mathrm{d}x) = -2\int_{\Gamma_G}\left(\nabla_x\left\{\sqrt{\frac{n_t(x)}{\mu_{\beta_t}(x)}}\right\}\right)^2 \mu_{\beta_t}(\mathrm{d}x), \tag{2.30}$$

and therefore translates a mean reversion towards $\mu_{\beta_t}$ in the first term of Equation (2.29).

We now study the size of the difference between $\widehat{\mathcal{L}}_{2,t}$ and $\widetilde{\mathcal{L}}_{2,t}$: this difference is suspected to be small since we expect to show that $m_t(.|x) \longrightarrow \nu(.)$. Therefore, we introduce the “approximation" term of $\nabla_x \mathbf{U}_{\nu,t}(x)$ at time $t$:

$$\widetilde{R}_t(x) = \int_V \nabla_x(\mathbf{U}_{y,t})(x) m_t(y|x)\mathrm{d}y.$$

The relationship $\nabla_x \log(f) = 2\frac{\nabla_x\sqrt{f}}{\sqrt{f}}$, the Cauchy-Schwarz inequality and $2ab \leq a^2+b^2$ yield:

$$\begin{aligned}
|\kappa_t| &= \beta_t\left|\int_{\Gamma_G}\left(\widetilde{R}_t(x) - \nabla_x\mathbf{U}_{\nu,t}(x)\right)\nabla_x\left\{\log\frac{n_t(x)}{\mu_{\beta_t}(x)}\right\}n_t(\mathrm{d}x)\right| \\
&= 2\beta_t\left|\int_{\Gamma_G}\left(\widetilde{R}_t(x) - \nabla_x\mathbf{U}_{\nu,t}(x)\right)\nabla_x\left\{\sqrt{\frac{n_t(x)}{\mu_{\beta_t}(x)}}\right\}\sqrt{\frac{\mu_{\beta_t}(x)}{n_t(x)}}n_t(\mathrm{d}x)\right| \\
&\leq 2\beta_t\sqrt{\int_{\Gamma_G}\left(\widetilde{R}_t(x) - \nabla_x\mathbf{U}_{\nu,t}(x)\right)^2 n_t(\mathrm{d}x)}\cdot\sqrt{\int_{\Gamma_G}\nabla_x\left(\sqrt{\frac{n_t(x)}{\mu_{\beta_t}(x)}}\right)^2\mu_{\beta_t}(x)\mathrm{d}x} \\
&\leq \beta_t^2\int\left(\widetilde{R}_t(x) - \nabla_x\mathbf{U}_{\nu,t}(x)\right)^2 n_t(x)\mathrm{d}x + \int_{\Gamma_G}\left(\nabla_x\left\{\sqrt{\frac{n_t(x)}{\mu_{\beta_t}(x)}}\right\}\right)^2\mu_{\beta_t}(\mathrm{d}x).
\end{aligned}$$

If $d_{TV}$ denotes the total variation distance, the first term of the right hand side leads to:

$$\begin{aligned}
\left|\widetilde{R}_t(x) - \nabla_x\mathbf{U}_{\nu,t}(x)\right| &= \left|\int_V \nabla_x\mathbf{d}_t^2(x,y)[m_t(y|x) - \nu(y)]\mathrm{d}y\right| \\
&\leq \|\nabla_x\mathbf{d}_t^2(x,y)\|_\infty\left|\int_V[m_t(y|x) - \nu(y)]\mathrm{d}y\right| \\
&\leq 2\|\nabla_x\mathbf{d}_t^2(x,y)\|_\infty d_{TV}(m_t(.|x),\nu) \\
&\leq \sqrt{2}\|\nabla_x\mathbf{d}_t^2(x,y)\|_\infty\sqrt{\int_V\log\frac{m_t(y|x)}{\nu(y)}m_t(y|x)\mathrm{d}y},
\end{aligned}$$

where the last line comes from the Csiszár-Kullback inequality. Since $\mathbf{d}_t^2(\cdot,y)$ is differentiable and its derivative is bounded for all $y \in V$ by $12\mathcal{D}_{G,t}$, we can use $I_t$ defined in Equation (2.25) to obtain:

$$\int_{\Gamma_G}\left(\widetilde{R}_t - \nabla_x\mathbf{U}_{\nu,t}(x)\right)^2(x)n_t(x)\mathrm{d}x \leq 2(12\mathcal{D}_{G,t})^2 I_t.$$

Consequently, we obtain:

$$|\kappa_t| \leq 2(12\mathcal{D}_{G,t}\beta_t)^2 I_t + \int_{\Gamma_G}\left(\nabla_x\left\{\sqrt{\frac{n_t(x)}{\mu_{\beta_t}(x)}}\right\}\right)^2\mu_{\beta_t}(\mathrm{d}x). \tag{2.31}$$

Taking the inequalities Equation (2.30) and Equation (2.31), we now obtain in Equation (2.29):

$$J_{3,t} \leq 2(12\mathcal{D}_{G,t}\beta_t)^2 I_t - \int_{\Gamma_G} \left( \nabla_x \left\{ \sqrt{\frac{n_t(x)}{\mu_{\beta_t}(x)}} \right\} \right)^2 \mu_{\beta_t}(\mathrm{d}x).$$

We denote $f_t = \sqrt{\frac{n_t}{\mu_{\beta_t}}}$. Since $\|f\|_{2,\mu_{\beta_t}}^2 = 1$, one can easily see that $\mathrm{Ent}_{\mu_{\beta_t}}(f_t^2) = J_t$. Now, the Logarithmic Sobolev inequality on the (quantum) graph $\Gamma_G$ for the measure $\mu_{\beta_t}$ stated in Proposition 6 shows that:

$$\int_{\Gamma_G} \left( \nabla_x \left\{ \sqrt{\frac{n_t(x)}{\mu_{\beta_t}(x)}} \right\} \right)^2 \mu_{\beta_t}(\mathrm{d}x) \geq \frac{e^{-c^\star(\mathbf{U}_{\nu,t})\beta_t}}{C_{\Gamma_G}(1+\beta_t)} J_t,$$

where $c^\star(\mathbf{U}_{\nu,t})$ is defined in Equation (2.6) and is related to the maximal depth of a well containing a local but not global minimum. We thus obtain:

$$J_{3,t} = \int \widetilde{\mathcal{L}}_{2,t} \left[ \log \frac{n_t(x)}{\mu_{\beta_t}(x)} \right] n_t(\mathrm{d}x) \leq 2(12\mathcal{D}_{G,t}\beta_t)^2 I_t - \frac{e^{-c^\star(\mathbf{U}_{\nu,t})\beta_t}}{C_{\Gamma_G}(1+\beta_t)} J_t.$$

The proof is concluded by regrouping the three terms. □

### 2.7.2 Study of $\partial_t I_t$

**Proposition 4.** *Assume that $\mathcal{D}_{G,t} \geq 1$ and $\beta_t$ is an increasing inverse temperature with $\beta_t \geq 1$, then:*

$$\partial_t I_t \leq -\alpha_t I_t - \partial_t J_t + \mathcal{D}_{G,t}^2[\beta_t' + 6^3\beta_t^2] + \beta_t(\delta_t + \varphi_t). \tag{2.32}$$

*Proof:* We compute the derivative of $I_t$. Observing that $m_t(x,y) = n_t(x)m_t(y|x)$, we have:

$$\begin{aligned}
\partial_t I_t &= \partial_t \left\{ \int_{\Gamma_G} n_t(x) \left( \int_V \log \frac{m_t(y|x)}{\nu(y)} m_t(y|x)\mathrm{d}y \right) \mathrm{d}x \right\} \\
&= \partial_t \left\{ \int_{\Gamma_G} \int_V \log \frac{m_t(y|x)}{\nu(y)} m_t(x,y)\mathrm{d}x\mathrm{d}y \right\} \\
&= \underbrace{\int_{\Gamma_G} \int_V \partial_t\{\log m_t(y|x)\} m_t(x,y)\mathrm{d}x\mathrm{d}y}_{:=I_{1,t}} - \underbrace{\int_{\Gamma_G} \int_V \partial_t\{\log \nu(y)\} m_t(x,y)\mathrm{d}x\mathrm{d}y}_{:=I_{2,t}} \\
&+ \underbrace{\int \log \frac{m_t(y|x)}{\nu(y)} \partial_t m_t(x,y)\mathrm{d}x\mathrm{d}y}_{:=I_{3,t}}.
\end{aligned}$$

The computation of the first two terms is straightforward. For the first one we have:

$$\begin{aligned}
I_{1,t} &= \int_{\Gamma_G} \int_V \partial_t\{\log m_t(y|x)\} m_t(x,y)\mathrm{d}x\mathrm{d}y = \int_{\Gamma_G} \int_V \frac{\partial_t m_t(y|x)}{m_t(y|x)} m_t(x,y)\mathrm{d}x\mathrm{d}y \\
&= \int_{\Gamma_G} \int_V \partial_t m_t(y|x) n_t(x)\mathrm{d}x\mathrm{d}y = \int_{\Gamma_G} n_t(x) \left( \int_V \partial_t m_t(y|x)\mathrm{d}y \right) \mathrm{d}x \\
&= \int_{\Gamma_G} n_t(x) \partial_t \left\{ \underbrace{\int_V m_t(y|x)\mathrm{d}y}_{=1} \right\} \mathrm{d}x = 0.
\end{aligned}$$

The computation of $I_{2,t}$ is easy since $\nu$ does not depend on $t$, implying that $I_{2,t} = 0$.

For the third term, we use the backward Kolmogorov equation and obtain:

$$\begin{aligned}I_{3,t} &= \int_{\Gamma_G}\int_V \log\frac{m_t(y|x)}{\nu(y)}\partial_t m_t(x,y)\mathrm{d}x\mathrm{d}y = \int_{\Gamma_G}\int_V \mathcal{L}_t\left(\log\frac{m_t(y|x)}{\nu(y)}\right)m_t(x,y)\mathrm{d}x\mathrm{d}y\\ &= \underbrace{\int_{\Gamma_G}\int_V \mathcal{L}_{1,t}\left(\log\frac{m_t(y|x)}{\nu(y)}\right)m_t(x,y)\mathrm{d}x\mathrm{d}y}_{:=I^1_{3,t}} + \underbrace{\int_{\Gamma_G}\int_V \mathcal{L}_{2,t}\left(\log\frac{m_t(y|x)}{\nu(y)}\right)m_t(x,y)\mathrm{d}x\mathrm{d}y}_{:=I^2_{3,t}}.\end{aligned}$$

The jump part $\mathcal{L}_{1,t}$ exhibits a mean reversion on the entropy of the conditional law: applying the Jensen inequality for the logarithmic function and the measure $\nu$, we obtain:

$$\begin{aligned}I^1_{3,t} &= \alpha_t\int_{\Gamma_G}\int_V\left[\int_V\left(\log\frac{m_t(y'|x)}{\nu(y')}-\log\frac{m_t(y|x)}{\nu(y)}\right)\nu(y')\mathrm{d}y'\right]m_t(x,y)\mathrm{d}x\mathrm{d}y\\ &= \alpha_t\int_{\Gamma_G}\int_V\left[\int_V\left(\log\frac{m_t(y'|x)}{\nu(y')}\right)\nu(y')\mathrm{d}y'\right]\ m_t(x,y)\mathrm{d}x\mathrm{d}y-\alpha_t I_t\\ &\le \alpha_t\int_{\Gamma_G}\int_V\log\left(\int_V\frac{m_t(y'|x)}{\nu(y')}\nu(y')\mathrm{d}y'\right)\ m_t(x,y)\mathrm{d}x\mathrm{d}y-\alpha_t I_t\\ &\le \alpha_t\int_{\Gamma_G}\int_V\log\left(\int_V m_t(y'|x)\mathrm{d}y'\right)\ m_t(x,y)\mathrm{d}x\mathrm{d}y-\alpha_t I_t\\ &\le \alpha_t\int_{\Gamma_G}\int_V\log 1\cdot m_t(x,y)\mathrm{d}x\mathrm{d}y-\alpha_t I_t\le -\alpha_t I_t. \qquad (2.33)\end{aligned}$$

If we consider the action of $\mathcal{L}_{2,t}$ on the entropy of the conditional law, using $m_t(x,y) = m_t(y|x)n_t(x)$ yields:

$$\begin{aligned}I^2_{3,t} &= \int_{\Gamma_G}\int_V\mathcal{L}_{2,t}\left(\log\frac{m_t(y|x)}{\nu(y)}\right)m_t(x,y)\mathrm{d}x\mathrm{d}y\\ &= \int_{\Gamma_G}\int_V\mathcal{L}_{2,t}\left(\log\frac{m_t(x,y)}{\nu(y)\cdot n_t(x)}\right)m_t(x,y)\mathrm{d}x\mathrm{d}y\\ &= \int_{\Gamma_G}\int_V\mathcal{L}_{2,t}\log\left(m_t(x,y)\right)m_t(x,y)\mathrm{d}x\mathrm{d}y-\int_{\Gamma_G}\int_V\mathcal{L}_{2,t}\log\left(n_t(x,y)\right)m_t(x,y)\mathrm{d}x\mathrm{d}y,\end{aligned} \qquad (2.34)$$

because $\mathcal{L}_{2,t}(\nu)(y)=0$ ($\mathcal{L}_{2,t}$ only involves the $x$ component). We now study the first term of Equation (2.34):

$$\begin{aligned}\int_{\Gamma_G}\int_V\mathcal{L}_{2,t}\left(\log m_t(x,y)\right)\mathrm{d}m_t(x,y) &= \frac{1}{2}\int_{\Gamma_G}\int_V\Delta_x\left[\log m_t(x,y)\right]\mathrm{d}m_t(x,y)\\ &\quad-\beta_t\int_{\Gamma_G}\int_V<\nabla_x\mathbf{U}_{y,t},\nabla_x\log m_t(x,y)>\mathrm{d}m_t(x,y)\\ &= \frac{1}{2}\int_{\Gamma_G}\int_V\partial^2_x\left\{\log m_t(x,y)\right\}\mathrm{d}m_t(x,y)\\ &\quad-\beta_t\int\partial_x\mathbf{U}_{y,t}\ \partial_x\log m_t(x,y)\mathrm{d}m_t(x,y).\end{aligned}$$

The first term of the right-hand side $\int_{\Gamma_G}\int_V\partial^2_x x\left[\log m_t(x,y)\right]\mathrm{d}m_t(x,y)$ deserves special attention:

$$\begin{aligned}\int_{\Gamma_G}\int_V \partial_x^2\{\log m_t(x,y)\}\mathrm{d}m_t(x,y) &= \int_{\Gamma_G}\int_V \partial_x\left\{\frac{\partial_x\{m_t(x,y)\}}{m_t(x,y)}\right\}\mathrm{d}m_t(x,y)\\ &= \int_{\Gamma_G}\int_V \frac{\partial_x^2\{m_t(x,y)\}}{m_t(x,y)}\mathrm{d}m_t(x,y) - \int_{\Gamma_G}\int_V \frac{[\partial_x\{m_t(x,y)\}]^2}{m_t(x,y)}\mathrm{d}x\mathrm{d}y\\ &= \int_{\Gamma_G}\int_V \partial_x^2\{m_t(x,y)\}\mathrm{d}x\mathrm{d}y - 4\int_{\Gamma_G}\int_V\left(\partial_x\sqrt{m_t(x,y)}\right)^2\mathrm{d}x\mathrm{d}y\\ &= \sum_{e\in E}\int_V\left[\partial_x m_t(e(L_e),y) - \partial_x m_t(e(0),y)\right]\mathrm{d}y\\ &- 4\int_{\Gamma_G}\int_V\left(\partial_x\sqrt{m_t(x,y)}\right)^2\mathrm{d}x\mathrm{d}y,\end{aligned}$$

where $\{e(s), 0\le s\le L_e\}$ is the parametrization of the edge $e\in E$ introduced in Section 2.3.1. The gluing conditions (see Definition 4 and Equation (2.9)) yield:

$$\sum_{e\in E}\int_V\left[\partial_x m_t(e(L_e),y) - \partial_x m_t(e(0),y)\right]\mathrm{d}y = \int_V\sum_{v\in V}\left[\sum_{e\sim v} -d_e m_t(v,y)\right]\mathrm{d}y = 0.$$

As a consequence, we obtain:

$$\begin{aligned}\int_{\Gamma_G}\int_V \mathcal{L}_{2,t}\left(\log m_t(x,y)\right)\mathrm{d}m_t(x,y) &= -2\int_{\Gamma_G}\int_V\left(\partial_x\sqrt{m_t(x,y)}\right)^2\mathrm{d}x\mathrm{d}y\\ &- \beta_t\int_{\Gamma_G}\int_V \partial_x\mathbf{U}_{y,t}\partial_x\log m_t(x,y)\mathrm{d}m_t(x,y).\end{aligned}\tag{2.35}$$

The Cauchy-Schwarz inequality applied on the second term leads to:

$$\begin{aligned}\left|\beta_t\int_{\Gamma_G}\int_V \partial_x\mathbf{U}_{y,t}\partial_x\log m_t(x,y)\mathrm{d}m_t(x,y)\right| &\le \beta_t\|\partial_x\mathbf{d}_t^2\|_{2,m_t}\sqrt{\int_{\Gamma_G}\int_V\left(\partial_x\log m_t(x,y)\right)^2\mathrm{d}m_t(x,y)}\\ &\le \beta_t\|\partial_x\mathbf{d}_t^2\|_\infty\sqrt{\int_{\Gamma_G}\int_V\left(\partial_x\log m_t(x,y)\right)^2 m_t(x,y)\mathrm{d}x\mathrm{d}y}\\ &\le 24\beta_t\mathcal{D}_{G,t}\sqrt{\int_{\Gamma_G}\int_V\left(\partial_x\sqrt{m_t(x,y)}\right)^2\mathrm{d}x\mathrm{d}y}\\ &\le 2(6\mathcal{D}_{G,t}\beta_t)^2 + 2\int_{\Gamma_G}\int_V\left(\partial_x\sqrt{m_t(x,y)}\right)^2\mathrm{d}x\mathrm{d}y.\end{aligned}$$

Inserting this in Equation (2.35) leads to:

$$\int_{\Gamma_G}\int_V \mathcal{L}_{2,t}\log m_t(x,y)\mathrm{d}m_t(x,y) \le 2(6\beta_t\mathcal{D}_{G,t})^2.\tag{2.36}$$

We study the second term of (2.34) and use the proof of Proposition 3: the decomposition (2.26) with Equations (2.27) and (2.28) yield:

$$\partial_t J_t \le \left[\beta_t'\mathcal{D}_{G_t}^2 + \beta_t\delta_t\right] + \int\mathcal{L}_{2,t}\left(\log\frac{n_t(x)}{\mu_{\beta_t}(x)}\right)m_t(\mathrm{d}x,\mathrm{d}y).$$

This implies:

$$-\int_{\Gamma_G}\int_V \mathcal{L}_{2,t}\left(\log n_t(x)\right)m_t(\mathrm{d}x,\mathrm{d}y) \le -\partial_t J_t + \left[\beta_t'\mathcal{D}_{G_t}^2 + \beta_t\delta_t\right] - \int\mathcal{L}_{2,t}\left(\log\mu_{\beta_t}(x)\right)m_t(\mathrm{d}x,\mathrm{d}y).\tag{2.37}$$

Using the definition of $\mu_{\beta_t}$, and the upper bounds given by Proposition 2 and Equation (2.23) we obtain:

$$\begin{aligned}
-\int_{\Gamma_G}\int_V \mathcal{L}_{2,t} \log \frac{e^{-\beta_t \mathbf{U}_{\nu,t}(x)}}{Z_{\beta_t}} &= \int_{\Gamma_G}\int_V \left[\mathcal{L}_{2,t}\left(\beta_t \mathbf{U}_{\nu,t}(x)\right) + \mathcal{L}_{2,t}\left(\log Z_{\beta_t}\right)\right] m_t(\mathrm{d}x, \mathrm{d}y) \\
&= \beta_t \int_{\Gamma_G}\int_V \mathcal{L}_{2,t}\left(\mathbf{U}_{\nu,t}(x)\right) m_t(\mathrm{d}x, \mathrm{d}y) \\
&= \beta_t \int_{\Gamma_G}\int_V \frac{1}{2}\Delta_x \mathbf{U}_{\nu,t}(x) - \beta_t \nabla_x \mathbf{U}_{y,t}(x) \nabla_x \mathbf{U}_{\nu,t}(x) m_t(\mathrm{d}x, \mathrm{d}y) \\
&\leq \frac{\beta_t}{2}\|\Delta \mathbf{U}_{\nu,t}(x)\|_\infty + \beta_t^2 \|\nabla_x \mathbf{U}_{y,t}\|_\infty \qquad (2.38) \\
&\leq \frac{\beta_t}{2}\varphi_t + (12\beta_t \mathcal{D}_{G,t})^2.
\end{aligned}$$

This inequality used in Equation (2.37) yields:

$$-\int \mathcal{L}_{2,t}\left(\log n_t(x)\right) m_t(x,y) \leq -\partial_t J_t + \left[\beta_t' \mathcal{D}_{G_t}^2 + \beta_t \delta_t\right] + \frac{\beta_t}{2}\varphi_t + (12\beta_t)^2 \mathcal{D}_{G,t}^2. \tag{2.39}$$

We now use (2.36) and (2.39) in (2.34) to obtain:

$$I_{3,t}^2 \leq -\partial_t J_t + \mathcal{D}_{G,t}^2[\beta_t' + 6^3 \beta_t^2] + \beta_t(\delta_t + \varphi_t). \tag{2.40}$$

Combining (2.33) and (2.40) leads to the desired inequality given by Equation (2.32). □

### 2.7.3 Convergence of the entropy

The use of Propositions 3 and 4 makes it possible to obtain a system of coupled differential inequalities.

*Proof of Theorem 1*: If we denote $a = 12(\mathcal{D}_G^2 + 1)$, such that for $t$ large enough $(12\mathcal{D}_{G,t})^2 \leq a$, we can write:

$$\begin{cases} J_t' \leq -\frac{e^{-c^\star(\mathbf{U}_{\nu,t})\beta_t}}{C_{\Gamma_G}(1+\beta_t)} J_t + 2a\beta_t^2 I_t + a\beta_t' + \beta_t \delta_t \\ I_t' \leq -J_t' + a[\beta_t' + 2\beta_t^2] + \beta_t(\delta_t + \varphi_t). \end{cases}$$

We introduce an auxiliary function $K_t = J_t + k_t I_t$, where $k_t$ is a $\mathcal{C}^1(\mathbb{R}_+, \mathbb{R})$ positive decreasing function for $t$ large enough, so that $\lim\limits_{t\to\infty} k_t = 0$. We use the system above to deduce that:

$$\begin{aligned}
K_t' &= J_t' + k_t' I_t + k_t I_t' \\
&\leq J_t' + k_t I_t' \\
&\leq J_t' + k_t(-\alpha_t I_t - J_t') + a k_t(\beta_t' + 2\beta_t^2) + k_t \beta_t(\delta_t + \varphi_t) \\
&\leq (1 - k_t) J_t' - k_t \alpha_t I_t + a k_t(\beta_t' + 2\beta_t^2) + k_t \beta_t(\delta_t + \varphi_t).
\end{aligned}$$

In the first inequality, we use the fact that $k_t' \leq 0$ and $I_t$ is positive. The second inequality is given by Equation (2.32). The upper bound of Proposition 3 now leads to:

$$\begin{aligned}
K_t' \leq &-\epsilon_t(1 - k_t) J_t + I_t\left[-k_t \alpha_t + 2a\beta_t^2(1 - k_t)\right] \\
&+ k_t\left(3a\beta_t^2 + a\beta_t' + \beta_t(\delta_t + \varphi_t)\right) + a\beta_t' + \beta_t \delta_t,
\end{aligned}$$

where we denoted $\epsilon_t = \frac{e^{-c^\star(\mathbf{U}_{\nu,t})\beta_t}}{C_{\Gamma_G}(1+\beta_t)}$. We choose the function $k_t$ to obtain a mean reversion on $K_t$:

$$k_t := \frac{2a\beta_t^2}{\alpha_t + 2a\beta_t^2 - \epsilon_t/2}. \tag{2.41}$$

Note that this function is decreasing for sufficiently large $t$ as soon as $\beta_t = o(\alpha_t)$, which is the case according to the choices described in Theorem 1. Moreover, a straightforward consequence is $\lim_{t \longrightarrow +\infty} k_t = 0$. This ensures that a positive $T_0$ exists such that:

$$\forall t \geq T_0 \qquad 0 \leq k_t \leq \frac{1}{2}.$$

Consequently, we deduce that:

$$\begin{aligned}
\forall t \geq T_0 \qquad K_t' &\leq -\epsilon_t(1-k_t)J_t - k_t\frac{\epsilon_t}{2}I_t + k_t\left(3a\beta_t^2 + a\beta_t' + \beta_t(\delta_t + \varphi_t)\right) + a\beta_t' + \beta_t\delta_t \\
&\leq -\frac{\epsilon_t}{2}J_t - k_t\frac{\epsilon_t}{2}I_t + k_t\left(3a\beta_t^2 + a\beta_t' + \beta_t(\delta_t + \varphi_t)\right) + a\beta_t' + \beta_t\delta_t \\
&\leq -\frac{\epsilon_t}{2}K_t + \underbrace{k_t\left(3a\beta_t^2 + a\beta_t' + \beta_t(\delta_t + \varphi_t)\right) + a\beta_t' + \beta_t\delta_t}_{:=\eta_t}.
\end{aligned}$$

The next bound is an easy consequence of the Grönwall Lemma:

$$\forall t \geq T_0 \qquad K_t \leq K_{T_0} e^{-\displaystyle\int_{T_0}^t \frac{\epsilon_s}{2}\mathrm{d}s} + \int_{T_0}^T \eta_s e^{-\displaystyle\int_s^t \frac{\epsilon_u}{2}\mathrm{d}u}\mathrm{d}s,$$

which in turn implies that $K_t \longrightarrow 0$ as $t \longrightarrow +\infty$ as soon as $\eta_t = o_{t\sim+\infty}(\epsilon_t)$ with $\int_0^\infty \epsilon_u du = +\infty$.

We are now looking for a suitable choice for $\alpha_t$ , $\beta_t$ and $\delta_t$. Let us assume that $\alpha_t \sim \lambda t^\gamma$ for any $\gamma > 0$ and $\delta_t \sim \log^4 t / t^\rho$, with $\rho > 0$ and $\varphi_t \sim \log t$. This choice leads to:

$$k_t \sim \frac{2ab^2 \log(t+1)^2}{\lambda t^\gamma},$$

so that:

$$\eta_t \sim \frac{ab}{t} + \frac{b\log^5 t}{t^\rho} + \frac{2ab^2\log^2 t}{\lambda t^\gamma}\left((3ab^2+1)\log^2 t + \frac{a}{t} + \frac{b\log^5 t}{t^\rho}\right) = O(t^{-(1\wedge\gamma\wedge\rho)}\log^5 t).$$

At the same time, we can check that $\epsilon_t \sim \frac{t^{-c^\star(\mathbf{U}_{\nu,t})b}}{bC_{\Gamma_G}\log t}$. Now, our conditions on $\eta_t$ and $\epsilon_t$ imply that:

$$\int_0^\infty \epsilon_u \mathrm{d}u = +\infty \qquad \Longleftrightarrow \qquad b\,c^\star(U_\nu) \leq 1.$$

Using the convergence of $c^\star(\mathbf{U}_{\nu,t})$ towards $c^\star(U_\nu)$:

$$\eta_t = o(\epsilon_t) \qquad \Longleftrightarrow \qquad 1 \wedge \gamma \wedge \rho > b\,c^\star(U_\nu).$$

The optimal calibration of our parameters $\gamma$ and $b$ (minimal value of $\gamma$, maximal value of $b$) induces the choice $\gamma = 1$ and $b < c^\star(U_\nu)^{-1}$. As soon as $\rho > b\,c^\star(U_\nu)$ (and this is given by Proposition 2), up to the choices $\alpha_t \sim \lambda t$ and $\beta_t \sim b\log(t)$, we deduce that $K_t \longrightarrow 0$ when $t$ goes to infinity. Since $I_t$, $k_t$ and $J_t$ are positive, this also implies that $J_t \longrightarrow 0$ when $t$ goes to infinity. □

## 2.8 Functional inequalities

This section is devoted to the proof of the Log-Sobolev inequality for the measure $(\mu_\beta)_{\beta>0}$. The specific feature of this functional inequality deals with the quantum graph settings and deserves careful adaptation of the pioneering work of [77] (that only dealt with smooth compact manifolds). To derive a nice Log-Sobolev inequality at low temperature for $\mu_\beta$ in our more irregular setting, namely on the quantum graph, we start with a preliminary control of the Poincaré inequality and then obtain a good estimate with the help of standard functional analysis inequalities. We refer to [21] for a more in-depth description.

We prove the functional inequalities for the Gibbs measure corresponding to $U_\nu$, but they actually hold for any $\mathbf{U}_{\nu,t}$, up to a constant change. Namely one only needs to replace the graph's diameter $\mathcal{D}_G$ (every time it is used as an upper bound of $U_\nu$) by the maximum value of $\mathbf{d}_t$, denoted $\mathcal{D}_{G,t}$, and the energy barrier $c^\star(U_\nu)$ by the one corresponding to $\mathbf{U}_{\nu,t}$, denoted $c^\star(\mathbf{U}_{\nu,t})$.

The Sobolev inequality was used to deal with a Dirichlet form in the proof of proposition 3 (see Equation (2.30)). In particular, we need to obtain an accurate estimate when $\beta \longrightarrow +\infty$. For this purpose, we introduce the generic notation for Dirichlet forms :

$$\forall \beta \in \mathbb{R}_+^* \quad \forall f \in W^{1,2}(\mu_\beta) \qquad \mathcal{E}_\beta(f,f) = \|\nabla f\|_{\mu_\beta}^2 = \int_{\Gamma_G} |\nabla f(x)|^2 \mathrm{d}\mu_\beta(x).$$

If we denote $\langle f \rangle_{\mu_\beta} = \mu_\beta(f) = \int_{\Gamma_G} f \mathrm{d}\mu_\beta$, we are interested in showing the Poincaré inequality:

$$\|f - \langle f \rangle_{\mu_\beta}\|_{2,\mu_\beta}^2 = \int_{\Gamma_G} (f - \langle f \rangle_{\mu_\beta})^2 \mathrm{d}\mu_\beta \leq \lambda(\beta)\mathcal{E}_\beta(f,f), \tag{2.42}$$

and the Log-Sobolev Inequality (referred to as LSI below):

$$\int_{\Gamma_G} f^2 \log\left(\frac{f}{\|f\|_{2,\mu_\beta}}\right)^2 \mathrm{d}\mu_\beta \leq C(\beta)\mathcal{E}_\beta(f,f). \tag{2.43}$$

This technical section is split into two parts. The first one establishes a preliminary estimate when $\beta = 0$ (*i.e.*, when dealing with the uniform measure on $\Gamma_G$). The second one then uses this estimate to derive the asymptotic behavior of the LSI when $\beta \longrightarrow +\infty$.

### 2.8.1 Preliminary control for $\mu_0$ on $\Gamma_G$

We consider $\mu_0$ the normalized Lebesgue measure and use the standard notation for any measure $\mu$ on $\Gamma_G$:

$$\|f\|_{W^{1,p}(\mu)} := \|f\|_{p,\mu} + \|\nabla f\|_{p,\mu}.$$

Let us establish the next elementary result:

**Lemma 1.** *The ordinary Sobolev inequality holds on $\Gamma_G$, i.e., for all measurable functions $f$ we have:*

$$\|f - \langle f \rangle_0\|_{p,\mu_0}^2 \leq e^{2/e} L^2 \left[\frac{\mathcal{D}_G L}{2} + 1\right] \mathcal{E}_0(f,f), \tag{2.44}$$

*where $\mathcal{D}_G$ is the diameter of the graph and $L$ its perimeter defined by:*

$$L = \sum_{e \in E} L_e.$$

*Proof:* First, let us remind the reader that for a given interval $I$ in dimension 1 equipped with the Lebesgue measure $\lambda_I$, the Sobolev space $W^{1,p}(\lambda_I)$ is continuously embedded in $L^\infty(I)$ (compact injection when $I$ is bounded). In particular, it can be shown (see, *e.g.*, [39]) that while integrating w.r.t. the unnormalized Lebesgue measure:

$$\forall p \geq 1 \qquad \|f\|_{L^\infty(I)} \leq e^{1/e}\|f\|_{W^{1,p}(\lambda_I)}.$$

We now consider $f \in W^{1,p}(\mu_0)$. Since $G$ has a finite number of edges we can write: $\Gamma_G = \bigcup_{i=1}^n e_i$, where each $e_i$ can be seen as an interval of length $\ell_i$. We have seen that for each edge $e_i$:

$$\|f\|_{L^\infty(e_i)} \leq e^{1/e}\|f\|_{W^{1,p}(\lambda_{e_i})}.$$

We can use a union bound since $\Gamma_G$ represents the union of edges, and deduce that:

$$\|f\|_{L^\infty(\Gamma_G)} = \max_{1\leq i\leq n} \|f\|_{L^\infty(e_i)} \leq e^{1/e} \max_{1\leq i\leq n} \|f\|_{W^{1,p}(\lambda_{e_i})} \leq e^{1/e}\|f\|_{W^{1,p}(\lambda_{\Gamma_G})},$$

where the inequality above holds w.r.t. the unnormalized Lebesgue measure. If $L$ denotes the sum of the lengths of all edges in $\Gamma_G$, we obtain:

$$\|f\|_{L^\infty(\Gamma_G)} \leq e^{1/e}L\|f\|_{W^{1,p}(\mu_0)}. \tag{2.45}$$

Second, we establish a simple Poincaré inequality for $\mu_0$ on $\Gamma_G$. For any function $f \in W^{1,2}(\mu_0)$, we use the equality:

$$\begin{aligned}\|f - \langle f\rangle_0\|^2_{2,\mu_0} &= \frac{1}{2}\int_{\Gamma_G}\int_{\Gamma_G}[f(x) - f(y)]^2 \mathrm{d}\mu_0(x)\mathrm{d}\mu_0(y)\\ &= \frac{1}{2}\int_{\Gamma_G}\int_{\Gamma_G}\left[\int_{\gamma_{x,y}} f'(s)ds\right]^2 \mathrm{d}\mu_0(x)\mathrm{d}\mu_0(y).\end{aligned}$$

where $\gamma_{x,y}$ is the shortest path that connects $x$ to $y$, parametrized with speed 1 and $f'(s)$ refers to the derivative of $f$ w.r.t. this parametrization at time $s$. It should be noted that such a path exists because the graph $\Gamma_G$ is connected. The Cauchy-Schwarz inequality yields:

$$\|f - \langle f\rangle_0\|^2_{2,\mu_0} \leq \frac{1}{2}\int_{\Gamma_G}\int_{\Gamma_G}\left[\int_{\gamma_{x,y}} |\nabla f(s)|^2 ds\right]|\gamma_{x,y}|\mathrm{d}\mu_0(x)\mathrm{d}\mu_0(y) \leq \frac{\mathcal{D}_G L}{2}\|\nabla f\|^2_{2,\mu_0}. \tag{2.46}$$

The Sobolev inequality is now an obvious consequence of the previous inequality: consider $f \in W^{1,p}(\mu_0)$ and note that since $\mu_0(\Gamma_G) = 1$, then (2.45) applied with $p = 2$ leads to:

$$\begin{aligned}\forall q \geq 1 \qquad \|f - \langle f\rangle_0\|^2_{q,\mu_0} &= \left(\int_{\Gamma_G} |f - \langle f\rangle_0|^q \mathrm{d}\mu_0\right)^{2/q}\\ &\leq \|f - \langle f\rangle_0\|^2_{L^\infty(\Gamma_G)}\mu_0(\Gamma_G)\\ &\leq e^{2/e}L^2\|f - \langle f\rangle_0\|^2_{W^{1,2}(\mu_0)}.\end{aligned}$$

We can use the Poincaré inequality established for $\mu_0$ on $\Gamma_G$ in Equation (2.46) and obtain:

$$\forall q \geq 1 \qquad \|f - \langle f\rangle_0\|^2_{q,\mu_0} \leq e^{2/e}L^2\left[\frac{\mathcal{D}_G L}{2} + 1\right]\|\nabla f\|^2_{2,\mu_0} = e^{2/e}L^2\left[\frac{\mathcal{D}_G L}{2} + 1\right]\mathcal{E}_0(f, f),$$

which concludes the proof. □

**Remark 1.** *The constants obtained in the proof of Lemma 1 above could certainly be improved. Nevertheless, such an improvement would have little importance for the final estimates obtained in Proposition 6.*

### 2.8.2 Poincaré Inequality on $\mu_\beta$

In the following, we show a Poincaré Inequality for the measure $\mu_\beta$ for large values of $\beta$. This preliminary estimate will be useful for deriving LSI on $\mu_\beta$. This functional inequality is strongly related to the classical minimal elevation of the energy function $U_\nu$ for joining any state $x$ to any state $y$. We refer to Section 11 for the definition of the set of paths $\gamma_{x,y}$, the smallest elevation to join $x$ and $y$ denoted by $H(x,y)$ and the definition of $c^\star(U_\nu)$ given by Equation (2.6). In the following, every time we write $\min U_\nu$ we refer to $\min\limits_{x\in\Gamma_G} U_\nu(x)$.

**Theorem 4** (Poincaré inequality for $\mu_\beta$)**.** *For all measurable functions $g$ defined on $\Gamma_G$:*

$$Var_{\mu_\beta}(g) \leq \frac{|E| \max_{e\in E} L_e e^{2\mathcal{D}_G}}{2} e^{\beta c^\star(U_\nu)} \mathcal{E}_{\mu_\beta}(g,g).$$

*Proof:* Since the graph $\Gamma_G$ is connected, for any two points $x$ and $y$, we can find a minimal path $\gamma_{x,y}$ that links them and that minimizes $H$. We denote $x_0 = x, x_1, \cdots x_l, x_{l+1} = y$ where the sequence $x_1, \cdots, x_l$ refers to the nodes included in the path $\gamma_{x,y}$ and we compute:

$$\begin{aligned}
Var_{\mu_\beta}(g) &= \frac{1}{2}\int_{\Gamma_G}\int_{\Gamma_G} (g(y)-g(x))^2 \mathrm{d}\mu_\beta(x)\mathrm{d}\mu_\beta(y) \\
&= \frac{1}{2}\int_{\Gamma_G}\int_{\Gamma_G} \left(\sum_{i=0}^{l}(g(x_{i+1})-g(x_i)\right)^2 \mathrm{d}\mu_\beta(x)\mathrm{d}\mu_\beta(y) \\
&= \frac{1}{2}\int_{\Gamma_G}\int_{\Gamma_G} \left(\sum_{i=0}^{l}\int_{x_i}^{x_{i+1}} g'(s)\mathrm{d}s\right)^2 \mathrm{d}\mu_\beta(x)\mathrm{d}\mu_\beta(y) \\
&= \frac{1}{2}\int_{\Gamma_G}\int_{\Gamma_G} \left(\int_{\gamma_{x,y}} g'(s)\mathrm{d}s\right)^2 \mathrm{d}\mu_\beta(x)\mathrm{d}\mu_\beta(y) \\
&\leq \frac{1}{2}\int_{\Gamma_G}\int_{\Gamma_G}\int_{\gamma_{x,y}} |\nabla g(s)|^2 \mathrm{d}s |\gamma_{x,y}| \mathrm{d}\mu_\beta(x)\mathrm{d}\mu_\beta(y).
\end{aligned}$$

The last line is implied by the Cauchy-Schwarz inequality.

We can organize the terms involved in the above upper bound in the following way: for any edge of the graph $e \in E$, we denote $V_e$ the set of points $(x,y)$ such that $\gamma_{x,y} \cap e \neq \emptyset$. We therefore have:

$$\begin{aligned}
Var_{\mu_\beta}(g) &\leq \frac{1}{2}\sum_{e\in E}\int_e |\nabla g(s)|^2 \mathrm{d}s \int_{V_e} |\gamma_{x,y}| \mathrm{d}\mu_\beta(x)\mathrm{d}\mu_\beta(y) \\
&\leq \frac{1}{2}\sum_{e\in E}\int_e |\nabla g(s)|^2 \mu_\beta(s) \int_{V_e} |\gamma_{x,y}| \frac{1}{\mu_\beta(s)} \mathrm{d}\mu_\beta(x)\mathrm{d}\mu_\beta(y)\mathrm{d}s \\
&\leq \frac{1}{2}\sum_{e\in E}\left[\int_e |\nabla g(s)|^2 \mu_\beta(s)\mathrm{d}s\right]\left[\sup_{\bar{s}\in e}\frac{1}{\mu_\beta(\bar{s})}\int_{V_e} |\gamma_{x,y}| \mathrm{d}\mu_\beta(x)\mathrm{d}\mu_\beta(y)\right] \\
&\leq \frac{1}{2}\int_{\Gamma_G} |\nabla g(s)|^2 \mu_\beta(s)\mathrm{d}s \sup_{\bar{s}\in\Gamma_G}\frac{1}{\mu_\beta(\bar{s})}\int_{V_{e_{\bar{s}}}} |\gamma_{x,y}| \mathrm{d}\mu_\beta(x)\mathrm{d}\mu_\beta(y).
\end{aligned}$$

In this case, $e_{\bar{s}}$ denotes the edge of the graph that contains the point $\bar{s}$ and the set $V_{e_{\bar{s}}}$ is still the set of couples $(x,y)$ defined above, associated with each edge $e_{\bar{s}}$. We introduce the quantity $\mathcal{A}$ defined as:

$$\mathcal{A} = \sup_{\bar{s}\in\Gamma_G}\frac{1}{\mu_\beta(\bar{s})}\int_{V_{e_{\bar{s}}}} |\gamma_{x,y}| \mathrm{d}\mu_\beta(x)\mathrm{d}\mu_\beta(y).$$

Using this notation, we have obtained that for all functions $g$, we have the Poincaré inequality:

$$Var_{\mu_\beta}(g) \leq \frac{\mathcal{A}}{2}\mathcal{E}_{\mu_\beta}(g,g). \tag{2.47}$$

All that remains to be done is to obtain an upper bound of $\mathcal{A}$.

$$\begin{aligned}
\mathcal{A} &= \sup_{\bar{s}\in\Gamma_G} \frac{1}{\mu_\beta(s)} \int_{V_{e_{\bar{s}}}} |\gamma_{x,y}| \mathrm{d}\mu_\beta(x)\mathrm{d}\mu_\beta(y) \\
&= \sup_{\bar{s}\in\Gamma_G} Z_\beta e^{\beta U_\nu(\bar{s})} \int_{V_{e_{\bar{s}}}} \frac{e^{-\beta(U_\nu(x)+U_\nu(y))}}{Z_\beta^2} |\gamma_{x,y}| \mathrm{d}x\mathrm{d}y \\
&= \frac{1}{Z_\beta} \sup_{\bar{s}\in\Gamma_G} \int_{V_{e_{\bar{s}}}} e^{\beta(U_\nu(\bar{s})-U_\nu(x)-U_\nu(y))} |\gamma_{x,y}| \mathrm{d}x\mathrm{d}y.
\end{aligned}$$

Since $\gamma_{x,y}$ is the minimal path for $H(x,y)$, we have $H(x,y) = \max\limits_{s\in\gamma_{x,y}} U_\nu(s)$. Therefore:

$$\begin{aligned}
\mathcal{A} &\leq \frac{1}{Z_\beta} \sup_{\bar{s}\in\Gamma_G} \int_{V_{e_{\bar{s}}}} e^{\beta((H(x,y)-U_\nu(x)-U_\nu(y))} |\gamma_{x,y}| \mathrm{d}x\mathrm{d}y \\
&\leq \frac{1}{Z_\beta} \sup_{\bar{s}\in\Gamma_G} \int_{V_{e_{\bar{s}}}} e^{\beta(c^\star(U_\nu)-\min(U_\nu))} |\gamma_{x,y}| \mathrm{d}x\mathrm{d}y \\
&\leq \frac{e^{\beta(c^\star(U_\nu)-\min(U_\nu))}}{Z_\beta} \sup_{\bar{s}\in\Gamma_G} \int_{V_{e_{\bar{s}}}} |\gamma_{x,y}| \mathrm{d}x\mathrm{d}y \\
&\leq \frac{e^{\beta(c^\star(U_\nu)-\min(U_\nu))}}{Z_\beta} |E| \max_{e\in E} L_e.
\end{aligned}$$

Using the definition of $Z_\beta$, we have:

$$\mathcal{A} \leq \frac{e^{\beta c^\star(U_\nu)} |E| \max_{e\in E} L_e}{\displaystyle\int_{\Gamma_G} e^{-\beta(U_\nu(x)-\min(U_\nu))} \mathrm{d}x}. \tag{2.48}$$

If $x^\star \in M_\nu$ is a Fréchet mean that minimizes $U_\nu$, we designate $\mathbb{B}\left(x^\star, \dfrac{1}{\beta}\right)$, as the ball of center $x^\star$ and radius $\dfrac{1}{\beta}$, for the geodesic distance $d$ on the graph $\Gamma_G$. It is easy to check that:

$$\begin{aligned}
|U_\nu(x) - U_\nu(x^\star)| &= \left| E_{Y\sim\nu}\left[d^2(x,Y) - d^2(x^\star,Y)\right]\right| \\
&\leq E_{Y\sim\nu}\left|d^2(x,Y) - d^2(x^\star,Y)\right| \leq 2\mathcal{D}_G \times d(x,x^\star).
\end{aligned}$$

We can then deduce a lower bound on the denominator involved in Equation (2.48):

$$\begin{aligned}
\int_{\Gamma_G} e^{-\beta(U_\nu(x)-\min(U_\nu))}\mathrm{d}x &\geq \int_{\mathbb{B}\left(x^\star,\frac{1}{\beta}\right)} e^{-\beta(U_\nu(x)-\min(U_\nu))}\mathrm{d}x \\
&\geq \int_{\mathbb{B}\left(x^\star,\frac{1}{\beta}\right)} e^{-2\mathcal{D}_G\beta d(x,x^\star)}\mathrm{d}x \\
&\geq e^{-2\mathcal{D}_G} \int_{\mathbb{B}\left(x^\star,\frac{1}{\beta}\right)} \mathrm{d}x.
\end{aligned}$$

The Lebesgue measure of $\mathbb{B}\left(x^\star, \frac{1}{\beta}\right)$ may be lower bounded by $\beta^{-1}$ since there is, at the least, one path in $\Gamma_G$ passing by the point $x^\star$. Inserting this inequality in (2.48) gives:

$$\mathcal{A} \leq |E| \max_{e \in E} L_e e^{2\mathcal{D}_G} \beta e^{\beta c^\star(U_\nu)}.$$

Using this upper bound in (2.47) leads to the desired Poincaré inequality. □

### 2.8.3 Sobolev Inequalities on $\mu_\beta$

**Preliminary control on Dirichlet forms**

The next result will be useful to derive a LSI for $\mu_\beta$ from a Poincaré inequality on $\mu_\beta$ (given in Theorem 4). It generalizes the Poincaré inequality for $p$ norms with $p > 2$ while using the Sobolev inequality given in Lemma 1.

First, we introduce the maximal elevation of $U_\nu$ as:

$$M = \sup_{x \in \Gamma_G} U_\nu(x) - \inf_{x \in \Gamma_G} U_\nu(x). \tag{2.49}$$

Note that in our case, $M$ may be upper bounded by $\mathcal{D}_G^2$.

**Proposition 5.** *For any $p > 2$ and all measurable functions $f$, we have:*

$$\forall \beta \geq 0 \qquad \|f - \langle f \rangle_\beta\|^2_{p,\mu_\beta} \leq 4L^2 \left(\frac{\mathcal{D}_G L}{2} + 1\right) e^{2/e + M\beta} \mathcal{E}_\beta(f, f). \tag{2.50}$$

*Proof:* The Jensen inequality applied to the convex function $x \longmapsto x^p$ yields:

$$\begin{aligned}
\|f - \langle f \rangle_\beta\|^p_{p,\mu_\beta} &= \int_{\Gamma_G} |f - \langle f \rangle|^p_\beta \mathrm{d}\mu_\beta \\
&= \int_{\Gamma_G} \left| f(x) - \int_{\Gamma_G} f(y) \mathrm{d}\mu_\beta(y) \right|^p \mathrm{d}\mu_\beta(x) \\
&= \int_{\Gamma_G} \left| \int_{\Gamma_G} f(x) - f(y) \mathrm{d}\mu_\beta(y) \right|^p \mathrm{d}\mu_\beta(x) \\
&\leq \int_{\Gamma_G} \int_{\Gamma_G} |f(x) - f(y)|^p \, \mathrm{d}\mu_\beta(y) \mathrm{d}\mu_\beta(x) \\
&\leq \int_{\Gamma_G} \left(|f(x) - \langle f \rangle_0| + |f(y) - \langle f \rangle_0|\right)^p \mathrm{d}\mu_\beta(y) \mathrm{d}\mu_\beta(x).
\end{aligned}$$

Again, the Jensen inequality $|a + b|^p \leq 2^{p-1}[|a|^p + |b|^p]$ implies that:

$$\begin{aligned}
\|f - \langle f \rangle_\beta\|^p_{p,\beta} &\leq 2^{p-1} \int_{\Gamma_G} |f(x) - \langle f \rangle_0|^p + |f(y) - \langle f \rangle_0|^p \mathrm{d}\mu_\beta(y) \mathrm{d}\mu_\beta(x) \\
&\leq 2^p \int_{\Gamma_G} |f(x) - \langle f \rangle_0|^p \mathrm{d}\mu_\beta(x) = 2^p \|f - \langle f \rangle_0\|^p_{p,\mu_\beta}.
\end{aligned}$$

We conclude that:

$$\|f - \langle f \rangle_\beta\|^2_{p,\mu_\beta} \leq 4 \|f - \langle f \rangle_0\|^2_{p,\mu_\beta}. \tag{2.51}$$

Using the fact that:

$$\mu_\beta(x) = \frac{e^{-\beta \min U_\nu}}{Z_\beta} \leq \frac{e^{-\beta \min U_\nu}}{Z_\beta} \times Z_0 \mu_0(x),$$

we also have: $\|f - \langle f \rangle_0\|_{p,\mu_\beta}^p \leq \dfrac{Le^{-\beta \min U_\nu}}{Z_\beta} \|f - \langle f \rangle_0\|_{p,\mu_0}^p$, since $Z_0$ is the perimeter $L$ of the graph $\Gamma_G$. Consequently, we obtain:

$$\|f - \langle f \rangle_0\|_{p,\mu_\beta}^2 = \left(\|f - \langle f \rangle_0\|_{p,\mu_\beta}^p\right)^{2/p} \leq \left(\frac{Le^{-\beta \min U_\nu}}{Z_\beta}\right)^{2/p} \|f - \langle f \rangle_0\|_{p,\mu_0}^2.$$

Using inequality (2.51), the fact that $Z_\beta \leq Le^{-\beta \min U_\nu}$ and the assumption $2/p < 1$, we conclude that:

$$\|f - \langle f \rangle_\beta\|_{p,\mu_\beta}^2 \leq \frac{4Le^{-\beta \min U_\nu}}{Z_\beta} \|f - \langle f \rangle_0\|_{p,\mu_0}^2.$$

The Sobolev inequality given by Lemma 1 implies that:

$$\|f - \langle f \rangle_\beta\|_{p,\mu_\beta}^2 \leq \frac{4L^3 \left(\mathcal{D}_G L/2 + 1\right) e^{2/e - \beta \min U_\nu}}{Z_\beta} \mathcal{E}_0(f,f). \tag{2.52}$$

We now find an upper bound for the Dirichlet form $\mathcal{E}_0(f,f)$ that involves $\mathcal{E}_\beta(f,f)$:

$$\begin{aligned}
\mathcal{E}_0(f,f) &= \int_{\Gamma_G} < \nabla f, \nabla f > \mathrm{d}\mu_0 \\
&= Z_\beta \int_{\Gamma_G} < \nabla f, \nabla f > e^{\beta U_\nu(x)} \frac{e^{-\beta U_\nu(x)}}{Z_\beta} \mathrm{d}\mu_0 \\
&\leq \frac{Z_\beta \exp\left(\beta \sup\limits_{x \in \Gamma_G} U_\nu(x)\right)}{Z_0} \int_{\Gamma_G} < \nabla f, \nabla f > \mathrm{d}\mu_\beta \\
&\leq \frac{Z_\beta \exp\left(\beta \sup\limits_{x \in \Gamma_G} U_\nu(x)\right)}{L} \mathcal{E}_\beta(f,f). 
\end{aligned} \tag{2.53}$$

Putting (2.52) and (2.53) together concludes the proof. □

### Log-Sobolev Inequality on $\mu_\beta$

For all probability measures $\mu$ and all measurable functions $f$, we denote:

$$\mathrm{Ent}_\mu(f^2) = \int_{\Gamma_G} f^2 \log\left(\frac{f^2}{\|f\|_{2,\mu}^2}\right) \mathrm{d}\mu.$$

**Proposition 6.** *The Log-Sobolev Inequality holds on $\Gamma_G$. A constant $C_{\Gamma_G}$ exists such that for all $\beta \geq 0$ and all $\mu_\beta$-measurable functions $f$, we have:*

$$\mathrm{Ent}_{\mu_\beta}(f^2) \leq C_{\Gamma_G}[1+\beta] e^{c^\star(U_\nu)\beta} \mathcal{E}_\beta(f,f).$$

Proof: We consider $\beta \geq 0$ and a $\mu_\beta$ measurable function $f$. We apply the Jensen inequality for the logarithmic function and the measure $f^2/\|f\|_{2,\mu_\beta}^2 \mathrm{d}\mu_\beta$ to obtain:

$$\begin{aligned}
\int_{\Gamma_G} \left(\frac{f}{\|f\|_{2,\mu_\beta}}\right)^2 \log\left(\frac{f^2}{\|f\|_{2,\mu_\beta}^2}\right) \mathrm{d}\mu_\beta &= \frac{2}{p-2} \int_{\Gamma_G} \left(\frac{f}{\|f\|_{2,\mu_\beta}}\right)^2 \log\left(\frac{|f|^{p-2}}{\|f\|_{2,\mu_\beta}^{p-2}}\right) \mathrm{d}\mu_\beta \\
&\leq \frac{2}{p-2} \log\left(\int_{\Gamma_G} \frac{|f|^{p-2}}{\|f\|_{2,,\mu_\beta}^{p-2}} \frac{f^2}{\|f\|_{2,,\mu_\beta}^2} \mathrm{d}\mu_\beta\right) \\
&\leq \frac{2}{p-2} \log\left(\frac{\|f\|_{p,\mu_\beta}^p}{\|f\|_{2,\mu_\beta}^p}\right) = \frac{p}{p-2} \log\left(\frac{\|f\|_{p,\mu_\beta}^2}{\|f\|_{2,\mu_\beta}^2}\right).
\end{aligned}$$

Observing that for all $x, \delta > 0$ we have $\log(x\delta) \leq \delta x$, we obtain $\log(x) \leq x\delta + \log(\delta^{-1})$. Therefore:

$$\begin{aligned}
\forall \delta > 0 \qquad \int_{\Gamma_G} f^2 \log\left(\frac{f}{\|f\|_{2,\mu_\beta}}\right)^2 \mathrm{d}\mu_\beta &= \|f\|^2_{2,\mu_\beta} \int_{\Gamma_G} \left(\frac{f}{\|f\|_{2,\mu_\beta}}\right)^2 \log\left(\frac{f^2}{\|f\|^2_{2,\mu_\beta}}\right) \mathrm{d}\mu_\beta \\
&\leq \frac{p\|f\|^2_{2,\mu_\beta}}{p-2} \log\left(\frac{\|f\|^2_{p,\mu_\beta}}{\|f\|^2_{2,\mu_\beta}}\right) \\
&\leq \frac{p\|f\|^2_{2,\mu_\beta}}{p-2} \left[\delta \frac{\|f\|^2_{p,\mu_\beta}}{\|f\|^2_{2,\mu_\beta}} + \log \frac{1}{\delta}\right].
\end{aligned}$$

Replacing $f$ with $f - \langle f \rangle_\beta$ and choosing $\delta = e^{-M\beta}$ leads to:

$$\begin{aligned}
&\int_{\Gamma_G} (f - \langle f \rangle_\beta)^2 \log\left(\frac{f - \langle f \rangle_\beta}{\|f - \langle f \rangle_\beta\|_{2,\beta}}\right)^2 \mathrm{d}\mu_\beta \\
&\qquad \leq \frac{p}{p-2} \left[e^{-\beta M} \|f - \langle f \rangle_\beta\|^2_{p,\mu_\beta} + M\beta \|f - \langle f \rangle_\beta\|^2_{2,\mu_\beta}\right].
\end{aligned}$$

Proposition 5 and the Poincaré inequality established in Theorem 4 yield:

$$\begin{aligned}
\mathrm{Ent}_{\mu_\beta}\left[(f - \langle f \rangle_\beta)^2\right] &= \int_{\Gamma_G} (f - \langle f \rangle_\beta)^2 \log\left(\frac{f - \langle f \rangle_\beta}{\|f - \langle f \rangle_\beta\|_{2,\mu_\beta}}\right)^2 \mathrm{d}\mu_\beta \\
&\leq \frac{p}{p-2}\left(4L^2 \frac{\mathcal{D}_G L + 2}{2} e^{2/e} + M\beta |E| \max_{e \in E} L_e \frac{e^{2\mathcal{D}_G}}{2} \beta e^{c^\star(U_\nu)\beta}\right) \mathcal{E}_\beta(f, f).
\end{aligned} \tag{2.54}$$

It remains to use Rothau's Lemma (see Lemma 5.1.4 of [21]) that states that for any measure $\mu$ and any constant $a$:

$$\mathrm{Ent}_\mu (g + a)^2 \leq \mathrm{Ent}_\mu g^2 + 2 \int_{\Gamma_G} g^2 \mathrm{d}\mu.$$

Let $p = 4$ in Equation (2.54). Putting this together with Rothau's Lemma and Theorem 4, we obtain the following:

$$\begin{aligned}
\mathrm{Ent}_{\mu_\beta}(f^2) &= \int_{\Gamma_G} f^2 \log\left(\frac{f^2}{\|f\|^2_{2,\mu_\beta}}\right) \mathrm{d}\mu_\beta \\
&\leq \mathrm{Ent}_{\mu_\beta}\left[(f - \langle f \rangle_\beta)^2\right] + 2Var_{\mu_\beta}(f) \\
&\leq \mathcal{E}_\beta(f, f) \left[\beta |E| \max_{e \in E} L_e e^{2\mathcal{D}_G} (1 + M\beta) e^{c^\star(U_\nu)\beta} + 4L^2 (2 + \mathcal{D}_G L) e^{2/e}\right] \\
&\leq C_{\Gamma_G} [1 + \beta] e^{c^\star(U_\nu)\beta} \mathcal{E}_\beta(f, f),
\end{aligned}$$

where $C_{\Gamma_G}$ is a large enough constant (independent of $\beta$) that could be made explicit in terms of constants $\mathcal{D}_G$, $|E|$ and $L$ since we trivially have $M \leq \mathcal{D}_G^2$. □

**Remark 2.** *Considering the fact that $\mathcal{D}_{G,t}$ converges to the graph's diameter $\mathcal{D}_G$, we can define $\mathbf{d}_t$ such that*

$$\sup_{t \in \mathbb{R}_+} \mathcal{D}_{G,t} < +\infty,$$

*and thus that there exists a constant $C_{\Gamma_G}$ independent of $\beta$ and of $t$ such that the above Sobolev inequality can be transferred to the Gibbs measure $\mu_{\beta_t}$ and the Dirichlet form $\mathcal{E}_{\beta_t}$ corresponding to $\mathbf{U}_{\nu,t}$:*

$$\mathrm{Ent}_{\mu_{\beta_t}}(f^2) \leq C_{\Gamma_G} [1 + \beta_t] e^{c^\star(\mathbf{U}_{\nu,t})\beta_t} \mathcal{E}_{\beta_t}(f, f)$$

## 2.9 Appendix: proof of Proposition 2

In what follows we denote $d$ the restriction of the distance to $\Gamma_G \times V$. For all $y$ in $V$, $d(\cdot, y)$ is continuous and $\mathcal{C}^\infty$ almost everywhere. The eventual points where $d(\cdot, y)$ is not differentiable are the cut locus points $x$, such that there exist multiple shortest paths connecting $x$ to $y$. Since we have a finite number of vertices and edges, there are only a finite number of such points.

Let $e \in E$ be a fixed edge, denote $u$ and $v$ its extremities (the adjacent vertices) and let $y \in V$ be a fixed vertex. Seeing that the graph has a finite number of edges and vertices, to prove Proposition 2, it is enough to show that there exists a function $\mathbf{d}_{e,y} : \mathbb{R}_+ \times e \to \mathbb{R}$, that respects the initial conditions restricted to $e$ and $y$.

**Reg** For all $t \geq 0$, $\mathbf{d}_{e,y,t}$ is in $\mathcal{C}^2(e)$.

**Neu** For all $t \geq 0$, $\mathbf{d}_{e,y,t}$ respects the Neumann conditions, namely:

$$\mathbf{d}'_{e,y,t}(u) = \mathbf{d}'_{e,y,t}(v) = 0.$$

**Uni** The function $\mathbf{d}_{e,y}$ converges uniformly to the restriction of $d(\cdot, y)$ to $e$.

**Der** There exists a constant $C_{e,y}$ such that for all $t \geq 0$:

$$\sup_{x \in e} |\partial_t \, \mathbf{d}_{e,y,t}(x)| \leq C \log^4(t)/t$$

A first remark is that there exists at most one point on $e$ such that there are multiple shortest paths going from it to a fixed $y$. Furthermore, we remind the reader that the quantum graph $\Gamma_G$ can be seen as a collection of real segments. Thus, using the parametrization $x_e$ our problem can be translated to approaching a continuous function $f : [0, L_e] \to \mathbb{R}$ that is piecewise $\mathcal{C}^\infty$, with derivatives that have at most one discontinuity point, by a function $g : \mathbb{R}_+ \times [0, L_e] \to \mathbb{R}$ that is $\mathcal{C}^1$ in time (its first variable) and $\mathcal{C}^2$ in $x$ (second variable) and whose partial derivative in $x$ is null at 0 and $L_e$ and the partial derivative in time is properly controlled. Let us first deal with the conditions **Reg**, **Neu** and **Uni**.

We follow a classical approach: for all $t$ we set $g_t(x) = f(x)$, except in a neighborhood of a problematic point and decrease the size of this neighborhood with time. To properly 'glue' $g_t$ we need to control its value and those of its first two derivatives at gluing points. In what follows we give an example of a polynomial function defined on an interval $[0, \epsilon]$, whose values, and those of its first two derivatives, at the end of the interval are fixed in advance.

For a fixed $\epsilon > 0$, we want to define on $[0, \epsilon]$ a polynomial function $p_\epsilon$ such that:

$$\begin{cases} p_\epsilon(0) = a \\ p_\epsilon(\epsilon) = b \end{cases} \qquad \begin{cases} p'_\epsilon(0) = c \\ p'_\epsilon(\epsilon) = d \end{cases} \qquad \begin{cases} p''_\epsilon(0) = e \\ p''_\epsilon(\epsilon) = h \end{cases} \tag{2.55}$$

An example of such function is:

$$\begin{aligned} p_\epsilon(x) = {} & \frac{1}{\epsilon}\left(a(\epsilon - x) + bx\right) + \frac{x(\epsilon - x)}{\epsilon^3}\left[(c\epsilon + a - b)(\epsilon - x) - x(d\epsilon + a - b)\right] \\ & + \frac{x^2(\epsilon - x)^2}{\epsilon^5}\left[(\epsilon - x)\left[\frac{e\epsilon^2}{2} + (2c + d)\epsilon + 3(a - b)\right] + x\left[\frac{h\epsilon^2}{2} - (c + 2d)\epsilon - 3(a - b)\right]\right] \end{aligned} \tag{2.56}$$

We can use such a polynomial function to define $g$ in the neighborhood of any of the problematic points, by eventually using a translation and adjusting the constants. For simplicity, we deal with a neighborhood of 0. Setting the proper boundary conditions, (2.55) becomes:

$$\begin{cases} p_\epsilon(0) = a = f(0) \\ p_\epsilon(\epsilon) = b = f(\epsilon) \end{cases} \qquad \begin{cases} p'_\epsilon(0) = c = 0 \\ p'_\epsilon(\epsilon) = d = f'(\epsilon) \end{cases} \qquad \begin{cases} p_\epsilon(0) = e = 0 \\ p''_\epsilon(\epsilon) = h = f''(\epsilon) \end{cases}$$

And thus we obtain:

$$\begin{aligned} p_\epsilon(x) = f(0) &\left( \frac{\epsilon - x}{\epsilon} + \frac{x(\epsilon - x)(\epsilon - 2x)}{\epsilon^3} + \frac{3x^2(\epsilon - 2x)(\epsilon - x)^2}{2\epsilon^5} \right) \\ &+ f(\epsilon) \left( \frac{x}{\epsilon} + \frac{x(\epsilon - x)\,(2x - \epsilon)}{\epsilon^3} + \frac{3x^2(2x - \epsilon)(\epsilon - x)^2}{2\epsilon^5} \right) \\ &+ f'(\epsilon) \left( -\frac{x^2(\epsilon - x)}{\epsilon^2} + \frac{x^2(\epsilon - 2x)(\epsilon - x)^2}{\epsilon^4} \right) + f''(\epsilon) \frac{x^3(\epsilon - x)^2}{2\epsilon^3}. \end{aligned}$$

**Time derivative** Setting for all $t \geq 0$ and all $x \in [0, \epsilon_t]$,

$$g_t(x) = p_{\epsilon_t}(x) \quad \text{with} \quad \epsilon_t = c_{e,y}/\log(t+1), \tag{2.57}$$

and observing that $f$ is a linear function (before $\bar{x}$, the discontinuity point of the derivative), we obtain that there exists $C_{y,e} >$ such that for all $x \in [0, \epsilon]$ and for all $\in R_+$:

$$|\partial_t g_t(x)| \leq C_{e,y} \frac{\log^4(t+1)}{t+1}.$$

The constant $c_{e,y}$ in the definition of $\epsilon_t$ is chosen such that for all $t$, $\epsilon_t$ is small enough (for example $\epsilon_t < (\bar{x} \wedge L_e \wedge 1)/2$.

Repeating this procedure for all problematic points, we can bound in the same manner the derivative of $g$ on $[0, L_e]$ and then define $\mathbf{d}_{e,y,t}$ according to $g$ (using the parametrization $x_e$). To conclude about the time derivatives, one can remark that $\mathcal{D}_{G,t}$, defined by:

$$\mathcal{D}_{G,t} = \max\{|\mathbf{d}_t(x,y)|, (x,y) \in \Gamma_G \times V\},$$

is bounded in time and:

$$\forall (t,x,y) \in \mathbb{R} \times \Gamma_G \times V, \quad |\partial_t \mathbf{U}_{\nu,t}(x,y)| \leq 2\mathcal{D}_{G,t}\ |\partial_t \mathbf{d}_t(x,y)|.$$

**First order derivative in $x$** The first derivative in $x$ is bounded as soon as $f$ is continuous and piecewise $\mathcal{C}^2$. To see that, one ca observe that, in general, $p_\epsilon$ can also be written as:

$$\begin{aligned} p_\epsilon(x) = f(0) + \frac{f(\epsilon) - f(0)}{\epsilon} &\left( x + \frac{x(\epsilon - x)\,(2x - \epsilon)}{\epsilon^2} + \frac{3x^2(2x - \epsilon)(\epsilon - x)^2}{2\epsilon^4} \right) \\ + c + (f'(\epsilon) - c) &\left( -\frac{x^2(\epsilon - x)}{\epsilon^2} + \frac{x^2(\epsilon - 2x)(\epsilon - x)^2}{\epsilon^4} \right) + \\ + e + (f''(\epsilon) - e) &\left( \frac{x^3(\epsilon - x)^2}{2\epsilon^3} \right). \end{aligned}$$

Using the fact that $f$ is the image through a parametrization of the distance $d$, we have that in our case:

$$p'_\epsilon(x) = \frac{5x^4 - 10\epsilon x^3 + 3\epsilon^2 x^2 + 3\epsilon^3 x}{\epsilon^4}$$

Thus, as soon as $\epsilon < 1$, $|p'_\epsilon(x)| \leq 6$ for all $x \in [0, \epsilon]$. Since the same property translates to $\mathbf{d}_{e,y}$, independently of $e$ and $y$ and taking into account that $|\partial_x d(\cdot, y)| = 1$ for all points on which it is well defined, we obtain:

$$|\partial_x \mathbf{d}_t(x,y)| \leq 6 \quad \text{and} \quad |\partial_x \mathbf{U}_{\nu,t}(x,y)| \leq 12\mathcal{D}_{G,t}, \quad \forall (t,x,y) \in \mathbb{R}_+ \times \Gamma_G \times V. \tag{2.58}$$

**Second order derivative in $x$** The second derivative in $x$ is however not uniformly bounded by a constant, but can be controlled by $1/\epsilon$ since:

$$p''_\epsilon(x) = \frac{1}{\epsilon} \cdot \frac{20x^3 + 30\epsilon x^2 + 6\epsilon^2 x + 3\epsilon^3}{\epsilon^3}$$

and thus for all $x \in [0, \epsilon]$, $p''_\epsilon(x) \leq 60/\epsilon$. This property can be transferred to $\mathbf{d}_{e,y}$. Using the definition (2.57) of $\epsilon_t$, one can see that there exists a constant $r = \max\{60/c_{e,y} \mid e \in E, y \in V\}$ such that for all $(t,x,y) \in \mathbb{R}_+ \times \Gamma_G \times V$ :

$$\partial_x^2 \mathbf{d}_t(x,y) \leq r \log(t+1) \tag{2.59}$$

Using the bound (2.58), for all $(t,x,y) \in \mathbb{R}_+ \times \Gamma_G \times V$, we have:

$$|\partial_x^2 \mathbf{U}_{\nu,t}| \leq 2\left(r\mathcal{D}_{G,t} \log(t+1) + 36\mathcal{D}_{G,t}^2\right), \tag{2.60}$$

which ends the proof of Proposition 2.

□

## 2.10 Further developments

In what follows, we present some results that are strongly related to the first part of this chapter, but are not part of the article [66].

In the previous sections, we have developed an algorithm that estimates the Fréchet mean on quantum graphs. Seeing that this is achieved by optimizing a function over $\Gamma_G$, a first natural question is: can this method be applied to other functions? And if yes, what assumptions on these functions are sufficient? In the first part we show that the method can indeed be used to optimize other cost functions, then we present some possible applications and finally we show that the algorithm can be adapted to sample from certain probability distributions.

### 2.10.1 Generalization of the method

Given a function $f : \Gamma \times V \to \mathbb{R}$ and a probability measure $\nu$ on the vertex set $V$, we define the following cost function $\mathcal{U}_\nu : \Gamma \to \mathbb{R}$:

$$\forall x \in \Gamma, \quad \mathcal{U}_\nu = \mathbb{E}_{Y \sim \nu}[f(x,Y)]. \tag{2.61}$$

Suppose that for all $y \in V$, $f(\cdot, y)$ is continuous and piecewise $\mathcal{C}^3(\Gamma_G)$, having a finite number of eventual points of discontinuity for its derivatives. Using the procedure described in Appendix 2.9, we approach $f$ uniformly by a smooth function $\mathbf{f} : \mathbb{R}_+ \times \Gamma_G \times V \to \mathbb{R}$ and associate to it the following time depending cost function:

$$\mathcal{U}_{\nu,t}(x) = \sum_{y \in V} \mathbf{f}_t(x,y)\nu(y). \tag{2.62}$$

**Remark 3.** *The regularity assumptions on $f$, might be loosened. Indeed, it is well known that any continuous function can be uniformly approached by smooth ones using a convolution procedure. For example, in [14], the authors use a regularization with a heat kernel in order to construct a Markov process that converges in probability towards the generalized Frechet mean, of a continuous function $k$, on a compact manifold that has no borders. A generalized Fréchet mean, has the same form as the classical one, only that the Riemannian distance $d$ was replaced by a more general continuous function $f$. However, for us, such a procedure would not suffice, since we also have to deal with the border conditions. Thus the regularization should be done in two steps: first the convolution to obtain a $\mathcal{C}^\infty(\Gamma_G)$ function, and then the polynomial approximation at the neighborhood of the vertices to comply with the Neumann conditions. However, this would no doubt render the proof more technical and it would be of no use to the applications presented in Section 2.10.2. Thus, we prefer to restrain ourselves the proposed framework.*

We introduce the same objects as in Section 2.2 and Section 2.3.1, by replacing at each time $U_\nu$ and $\mathbf{U}_{\nu,t}$ by $\mathcal{U}_\nu$ and $\mathcal{U}_{\nu,t}$. The new homegenized simulated annealing is given by:

$$\begin{cases} X_0 \in \Gamma_G \\ dX_t = -\beta_t \nabla \mathbf{f}_t(X_t, Y_{N_t^\alpha})dt + dB_t. \end{cases} \tag{2.63}$$

Since the new drift is still smooth, as soon as the corresponding diffusion operator is defined on the same domain (2.9) as the previous one, using the gluing conditions established in Definition 4, all the results with respect to the existence and regularity of the Markov process $(X_t, Y_t)_{t \geq 0}$ and its corresponding distributions presented in Section 2.3.1 still hold.

We keep the same notations as in the previous sections, for the distributions of the processes involved in (2.63) $(m_t, n_t, m_t(\cdot|y))$, the associated Gibbs measure $\mu_{\beta_t}$:

$$\mu_{\beta_t} = \frac{1}{Z_{\beta_t}} \exp(-\beta_t \mathcal{U}_{\nu,t}(x)), \tag{2.64}$$

where $Z_{\beta_t}$ is the normalization factor, and the relative entropy $J_t = KL(n_t || \mu_{\beta_t})$, but we emphasize that in this section all these objects no longer correspond to $U_\nu$ but to a general function $\mathcal{U}_\nu$.

We can now generalize Theorem 3 to this new framework.

**Theorem 5.** *For any constant $\lambda > 0$ such that $\alpha_t = \lambda(t+1)$ and $\beta_t = b\log(1+t)$ with $b < \{c^\star(\mathcal{U}_\nu)\}^{-1}$, we have:*

$$\lim J_t = 0 \qquad \mathit{as} \qquad t \longrightarrow +\infty.$$

The constant $c^\star(\mathcal{U}_\nu)$ represents the maximum depth of a well not containing a fixed global minimum, its analog for $U_\nu$ being defined in (2.6). Since the theorem's proof is essentially the same as the one of Theorem 3, we present only the corresponding key results, without demonstrating them. In a first part, we shall assume that the Sobolev inequality holds for each $\mu_{\beta_t}$, and we concentrate on the evolution of the relative entropy $J_t$.

### Study of the relative entropy

We remind the reader that the relative entropy $J_t$ is defined by:

$$J_t = \int_{\Gamma_G} \log\left[\frac{n_t(x)}{\mu_{\beta_t}(x)}\right] dn_t(x), \tag{2.65}$$

and that when dealing with its derivative we also need to measure the closeness between the conditional law of $y$ given $x$ at time $t$ to $\nu$, defined as:

$$I_t := \int_{\Gamma_G} \left[\int_V \log\frac{m_t(y|x)}{\nu(y)} m_t(y|x)dy\right] .dn_t(x). \tag{2.66}$$

The proof consist in obtaining functional inequalities for $J_t$ and $I_t$, and then using Grönwall's Lemma to conclude.

**Study of $\partial_t J_t$** Looking at the proof of Proposition 3 we can see that the derivative of $J_t$ can be separated in three terms that are dealt with separately.

- The first term $J_{1,t} = \int_{\Gamma_G} \partial_t \log(n_t(x))\mathrm{d}n_t(x)$ is always null since $n_t$ is a probability distribution.
- The second term, $J_{2,t}$, includes the time derivative of $\mu_{\beta_t}$, and thus naturally involves a uniform bound on the time derivative $\mathcal{U}_{\nu,t}$, and thus the regularization is established such that:

  $$\forall t \in \mathbb{R}_+, \forall x \in \Gamma_G, \quad |\partial_t \mathcal{U}_{\nu,t}(x)| \leq \delta_t,$$

  where $\delta_t$ is a function that converges (fast enough) to zero when $t$ goes to infinity. This term also involves an upper bound on the $x$ component, namely:

  $$\mathbf{M}(\mathcal{U}_{\nu,t}) = \max_{x\in\Gamma_G} \mathcal{U}_{\nu,t}(x).$$

  Since for each $t$ $\mathcal{U}_{\nu,t}$ is continuous in $x$, and $\Gamma_G$ is compact, the upper bound is always finite. Putting these two elements together we have:

  $$|J_{2,t}| = \left|\int_{\Gamma_G} \partial_t\{\log(\mu_{\beta_t}(x))\}\mathrm{d}n_t(x)\right| \leq \mathbf{M}(\mathcal{U}_{\nu,t})\beta_t' + \delta_t\beta_t.$$

- The third term, $J_3 = \int_{\Gamma_G} \log\left[\frac{n_t(x)}{\mu_{\beta_t}}\right] \partial_t n_t(x)\mathrm{d}x$, implies the time derivative of the distribution $n_t$. Since this distribution is not explicit, we deal with this term using the Kolmogorov equations. This is possible as soon as $\mathcal{U}_{\nu,t}$ are $C^2(\Gamma_G)$ and their directional derivatives are null at each vertex (which is ensured by the regularization procedure).

  The Kolmogorov equation implies the use of the diffusion operator and thus the derivatives (in $x$) of $\mathbf{f}_t$. Since the functions $\mathbf{f}_t(\cdot, y)$ are $\mathcal{C}^2(\Gamma_G)$, their derivatives are bounded and we denote:

  $$\|\nabla_x \mathbf{f}_t\|_\infty = \max\{|\nabla_x \mathbf{f}_t(x,y)| \mid x \in \Gamma_G, y \in V\}.$$

  Another important component, is the use of the Sobolev inequality (we explain why this holds later on). We can now state that:

  $$J_{3,t} \leq 2\|\nabla_x \mathbf{f}_t\|_\infty^2 \beta_t^2 I_t - \frac{e^{-c^\star(\mathcal{U}_{\nu,t})\beta_t}}{C(\Gamma_G, \mathcal{U}_\nu)(1+\beta_t)} J_t.$$

All these arguments allow us to conclude that:

$$\partial_t J_t \leq \mathbf{M}(\mathcal{U}_{\nu,t})\beta_t' + \delta_t\beta_t + 2\|\nabla_x \mathbf{f}_t\|_\infty^2 \beta_t^2 I_t - \frac{e^{-c^\star(\mathcal{U}_{\nu,t})\beta_t}}{C(\Gamma_G, \mathcal{U}_\nu)(1+\beta_t)} J_t. \tag{2.67}$$

**Study of $\partial_t I_t$** The derivative of $I_t$ also implies three terms. As shown in the proof of Proposition 4, dealing with the first two is trivial, since they are both null. This is obtained with no special assumptions on the involved distributions, except of course, for their derivability in time. Thus in our framework also:

$$I_{1,t} = \int_{\Gamma_G}\int_V \partial_t\{\log m_t(y|x)\}\mathrm{d}m_t(x,y) = 0 \quad I_{2,t} = \int_{\Gamma_G}\int_V \partial_t\{\log \nu(y)\}\mathrm{d}m_t(x,y) = 0.$$

The third term, $I_t$ involves once more the backward Kolmogorov equation and thus the Markov generator. Separating the part of the generator that acts on $Y$ form the part that acts on $X$, we obtain two terms:

- The part of the generator that acts on $Y$, is independent of the considered cost function, and thus the corresponding term can be dealt with exactly in the same way to obtain:
$$I^1_{3,t} \leq -\alpha_t I_t.$$
- The remaining term is the only one (in the study of $\partial_t I_t$) that is directly influenced by the fact that we changed the drift. It involves the first and second derivative of $\mathbf{f}_t(\cdot, y)$ and the inequality already obtained on $\partial_t J_t$. We denote:
$$\|\Delta_x \mathbf{f}_t\|_\infty = \max\{|\Delta_x \mathbf{f}_t(x,y)| \mid x \in \Gamma_G, y \in V\}.$$
The other elements of proof involve items that we kept unchanged like the gluing conditions (4) or the ellipticity of the diffusion operator. We can thus state the analog of (2.40):
$$I^2_{3,t} \leq -\partial_t J_t + \frac{3}{2}\beta_t^2\|\nabla_x \mathbf{f}_t\|^2_\infty + \frac{\beta_t}{2}\|\Delta_x \mathbf{f}_t\|_\infty + \beta_t\delta_t + \mathbf{M}(\mathcal{U}_{\nu,t})\beta'_t.$$

Putting all these elements together, we conclude that:

$$\partial_t I_t \leq -\partial_t J_t - \alpha_t I_t + \frac{3}{2}\beta_t^2\|\nabla_x \mathbf{f}_t\|^2_\infty + \frac{\beta_t}{2}\|\Delta_x \mathbf{f}_t\|_\infty + \beta_t\delta_t + \mathbf{M}(\mathcal{U}_{\nu,t})\beta'_t. \tag{2.68}$$

**Convergence of the relative entropy** This part demands an upper bound on the partial derivatives of $\mathbf{f}$. Denote $\varepsilon_t$ the bound from the Sobolev inequality:

$$\varepsilon_t = \frac{e^{-c^\star(\mathcal{U}_{\nu,t})\beta_t}}{C(\Gamma_G, \mathcal{U}_\nu)(1+\beta_t)}.$$

Let $K_t = J_t + k_t I_t$, where $k_t$ is chosen in way that allows us to obtain a mean reversion on $K_t$ and depends on the coefficients obtained for the differential inequalities of $J_t$ and $I_t$:

$$k_t = \frac{2\|\nabla_x \mathbf{f}_t\|^2_\infty \beta_t^2}{\alpha_t + 2\|\nabla_x \mathbf{f}_t\|^2_\infty \beta_t^2 + \varepsilon_t/2} \tag{2.69}$$

Using the inequalities (2.67) and (2.68) we can obtain, for $t$ large enough:

$$\partial_t K_t \leq -\frac{\epsilon_t}{2}K_t + \eta_t, \tag{2.70}$$

where:

$$\eta_t = k_t\left(\frac{3}{2}\beta_t^2\|\nabla_x \mathbf{f}_t\|^2_\infty + \frac{\beta_t}{2}\|\Delta_x \mathbf{f}_t\|_\infty + \beta_t\delta_t + \mathbf{M}(\mathcal{U}_{\nu,t})\beta'_t\right) + \mathbf{M}(\mathcal{U}_{\nu,t})\beta'_t + \beta_t\delta_t \tag{2.71}$$

We can conclude that $K_t \to 0$, when $t$ goes to infinity, using Grönwall's lemma, as soon as $\eta_t = o_{t\sim+\infty}(\epsilon_t)$ with $\int_0^\infty \epsilon_u du = +\infty$. The second condition is implied by the definition of the temperature schedule in Theorem 5 and the fact that $c^\star(\mathcal{U}_{\nu,t}) \to c^\star(\mathcal{U}_\nu)$.

The first condition deserves more attention since it requires bounds on the partial derivatives of the regularization function $\mathbf{f}$ and the time derivative of $\mathcal{U}_{\nu,t}$.

- We can immediately see that the time derivative of $\mathcal{U}_{\nu,t}$ needs to go fast enough to zero as $t$ goes to infinity:

  $$\beta_t \delta_t = o(\epsilon_t).$$

  As soon as $f$, the function approached by $\mathbf{f}$, is piecewise $\mathcal{C}^3(\Gamma_G)$, we can always build, using the same procedure as in Appendix 2.9, a function $\mathbf{f}$ that complies with this condition. This is the only point that involves the third order derivative of $f$.

- The regularization procedure presented in Appendix 2.9, allows us to obtain a finite bound on $\|\nabla_x \mathbf{f}_t\|_\infty$, and thus let $\mathfrak{c} > 0$ be such that:

  $$\|\nabla_x \mathbf{f}_t\|_\infty \leq \mathfrak{c}.$$

  This facilitates the proof since it implies that $k_t \sim \beta_t^2/\alpha_t$, but if $\|\nabla_x \mathbf{f}_t\|_\infty$ would go to infinity, it would not be a problem, as long as it would do so slow enough (for example $\|\nabla_x \mathbf{f}_t\|_\infty \sim \beta_t^k$, would have no influence on our result). Since $\alpha_t \sim t$, we obtain that the first term of $\eta_t$, is equivalent to $k_t \beta_t^2$ and thus equal to $o(\varepsilon_t)$, when $t$ goes to infinity.

- The last term we need to deal with is the one involving $\|\Delta_x \mathbf{f}_t\|_\infty$. In general, we cannot obtain a finite bound for this partial derivative. Furthermore, without imposing additional conditions on $f$, we cannot obtain the same bound as the one we had for the regularization of the distance (at least not with the same method). However, this in not necessary, since we would only like to have:

  $$\|\Delta_x \mathbf{f}_t\|_\infty \beta_t^3/\alpha_t = o(\varepsilon_t).$$

  One can check that the coefficients of the polynomial function $p_\epsilon$, defined in (2.56) Appendix 2.9, are of the form $c/\epsilon^k$, where $c$ is an arbitrary constant and $0 \leq k \leq 5$. Since the second derivative in $x$, of this polynomial function, has the same form, as soon as the size of the neighborhoods on which we approximate $f$ decreases as $1/\log t$, we have that there exists $\mathfrak{r} > 0$ such that:

  $$\|\Delta_x \mathbf{f}_t\|_\infty \leq \mathfrak{r} \log^5 t. \tag{2.72}$$

Putting all these bounds together we obtain that $\eta_t = o(\varepsilon)$, and thus that $K_t \to 0$. As before, using the fact that the $I_t$ and $J_t$ are positive, we have that the same result holds for $J_t$, thus ending the proof. □

In what follows, we present the Poincaré and Sobolev inequalities for the Gibbs measure associated to our cost function $\mathcal{U}_\nu$ and explain why the proofs provided in Section 2.8 still hold.

**Functional inequalities**

In the proof of the Poincaré inequality stated in Theorem 4, besides some inherent properties of the graph $\Gamma_G$, the only assumption we need to make on the function that generates the Gibbs measure $\mu_\beta$ is for it to be Lipschitz continuous. This is obviously the case for $\mathcal{U}_\nu$ since it is continuous and piecewise $\mathcal{C}^3$ (and of course for its regularized versions $\mathcal{U}_{\nu,t}$). Let $L(\mathcal{U}_\nu)$ be a Lipschitz constant for $\mathcal{U}_\nu$, namely such that:

$$\forall x, y \in \Gamma_G \quad |\mathcal{U}_\nu(x) - \mathcal{U}_\nu(y)| \leq L(\mathcal{U}_\nu) d(x, y).$$

Then the Poincaré inequality can be stated as:

$$Var_{\mu_\beta}(g) \leq \frac{|E| \max_{e \in E} L_e e^{L(\mathcal{U}_\nu)}}{2} e^{\beta c^\star(\mathcal{U}_\nu)} \mathcal{E}_{\mu_\beta}(g, g),$$

where we remind the reader that $|E|$ represents the number of edges and $\mathcal{E}_{\mu_\beta}(g, g)$ is the squared $L^2(\mu_\beta)$ norm of the gradient of $g$.

At a closer look, we see that with no additional assumptions on $\mu_\beta$, we can also deduce the Sobolev inequality on $\Gamma_G$. The key elements of its proof are the following:

- The Poincaré inequality.
- The preliminary control on the normalized Lebesgue measure obtained in Lemma 1, that relies entirely upon the graph's structure, namely its perimeter $L$, the graph's diameter $\mathcal{D}_G$ and the fact that it can be seen as a collection of real segments.
- The generalization of the Poincaré inequality for $L^p(\mu_\beta)$ norms, that besides the graph's structure, implies the maximal elevation of the function $\mathcal{U}_\nu$:

$$M(\mathcal{U}_\nu) = \max_{x \in \Gamma_G} \mathcal{U}_\nu(x) - \min_{x \in \Gamma_G} \mathcal{U}_\nu(x).$$

Putting all these elements together gives us the following inequality:

$$\mathrm{Ent}_{\mu_\beta}(f) \leq \mathcal{E}_\beta(f, f) \left[ \beta |E| \max_{e \in E} L_e e^{L(\mathcal{U}_\nu)} (1 + M(\mathcal{U}_\nu)\beta) e^{c^\star(\mathcal{U}_\nu)\beta} + 4L^2(2 + \mathcal{D}_G L) e^{2/e} \right]$$

This in particular implies that there exists a constant $C(\Gamma_G, \mathcal{U}_\nu)$ (independent of $\beta$) such that:

$$\mathrm{Ent}_{\mu_\beta}(f) \leq C(\Gamma_G, \mathcal{U}_\nu)[1 + \beta] e^{c^\star(\mathcal{U}_\nu)\beta} \mathcal{E}_{\mu_\beta}(f, f)$$

For each $\mathcal{U}_{\nu,t}$, there exists an equivalent constant $C(\Gamma_G, \mathcal{U}_{\nu,t})$, such that the corresponding Sobolev inequality holds and such that:

$$C(\Gamma_G, \mathcal{U}_{\nu,t}) \longrightarrow C(\Gamma_G, \mathcal{U}_\nu), \qquad \text{when } t \to +\infty.$$

. Supposing that $\mathcal{U}_{\nu,t}$ is continuous in time and converges uniformly towards $\mathcal{U}_\nu$, there exists a constant $\mathbf{C}(\Gamma_G, \mathcal{U}_\nu)$, such that the Sobolev inequalities hold for all $\mathcal{U}_{\nu,t}$.

### 2.10.2 Applications

Theorem 5 implies that the algorithm used to find the barycenter of a graph, described in Algorithm 3, Section 2.2.3, can be adapted in order to minimize other functions $\mathcal{U}_\nu$, as soon as they are the expectancy of continuous and piecewise $\mathcal{C}^3(\Gamma_G)$ functions $f$. In order to adapt the algorithm, we only need to change the way the coordinate $X_t$ jumps each time a new variable $Y$ is sampled according to the probability measure $\nu$. Suppose

$T$ is a jumping time. Then at time $T$, $X_t$ will advance from $X_{T^-}$ according to the drift $\nabla_x f(X_{T^-}, Y_T)$, following the graph's parametrization. If $X_{T^-} \in e$, an edge parametrized according to $x_e : e \to [0, L_e]$, then supposing that $x_e(T) - \frac{\beta_t}{\alpha_t}\nabla_x f(X_{T^-}, Y_T \in [0, L_e])$, the movement can be described by:

$$X_T = x_e^{-1}\left(x_e(T) - \frac{\beta_t}{\alpha_t}\nabla_x f(X_{T^-}, Y_T)\right) \tag{2.73}$$

**Fréchet $p$-means** A class of functions that fall under the assumptions of Theorem 5, are the Fréchet $p$-means, for $p \geq 1$. In that case the adaptation of the algorithm is straight forward, at each jumping time $T$, the coordinate $X$ advances towards $Y_T$, a distance of $p\beta_T/\alpha_T d^{p-1}(X_{T^-}, Y_T)$:

$$X_T = X_{T^-} + \frac{\beta_T}{\alpha_T} p d^{p-1}(X_{T^-}, Y_T)\overrightarrow{X_{T^-}Y_T}, \tag{2.74}$$

where $\overrightarrow{X_{T^-}Y_T}$ represents a geodesic path from $X_T$ to $Y_T$ in $\Gamma_G$.

**Median** Besides the barycenter, another Fréchet $p$-mean that could be of special interest is the one corresponding to $p = 1$, also called a median (since it is the generalization of the Euclidean median to metric spaces). An advantage of the median is that it provides a notion of centrality, less influenced by extreme values. Figure 2.9 illustrates an example of a graph were the barycenter is different from the median. For example, in the case of a citation network, the median would favor people that have a lot of coauthors, even though there might be very far from other individuals.

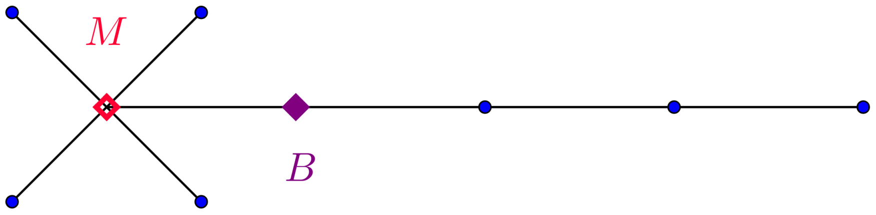


Figure 2.9: Example of a graph for which the median $M$ and barycenter $B$ are different. We consider the uniform probability on the vertex set and all edges of length 1.

### Barycenter on discrete and quantum graphs

The question that motivated the works developed in the first part of this chapter, was the one of finding a barycenter of a discrete graph $G$. Algorithm 3, developed in Section 2.2, estimates the Fréchet mean of a quantum graph.

In the examples provided in Section 2.4, the Fréchet mean of the discrete and quantum graph coincide. However, there is no reason to believe that this holds in general, especially since there are trivial counter examples: for a segment of length one, with uniform probability on the vertices, both vertices are barycenters for the discrete graph, but the corresponding quantum graph has a unique Fréchet mean: the center of the segment.

Since a tight relation between the two can be observed, one might wonder if for every q-min (minimizerss of $U_\nu$ on $\Gamma_G$ and thus a barycenter of $\Gamma_G$) there always exists a d-min (minimizer of $U_\nu$ over $G$) on the same edge. We can show that this is always the case for some very simple graphs, like a triangle or a square with edges of length one, independently of the of the probability measure $\nu$, but for example for a hexagon there exist probability measures $\nu$, that have an isolated q-min (with no adjacent d-min).

**How to estimate the barycenter of a discrete graph** The difference between the two Fréchet means, is not necessarily a problem for our method. The generalization provided in Theorem 5, allows us to estimate the barycenter of the underlying graph $G$. This can be done in a straightforward way by replacing the cost function $U_\nu$ by a new one $\widetilde{U_\nu}$, that on each edge $e$ of $\Gamma_G$, having $u, v$ as extremities, is defined as an linear interpolation of $U_\nu(v)$ and $U_\nu(u)$:

$$\widetilde{U_\nu}(x) = \frac{1}{L_e}\left(U_\nu(v)x_e(x) + U_\nu(u)(L_e - x_e(v))\right), \quad x \in e, \tag{2.75}$$

where $L_e$ is the length of the edge $e$, and $x_e$ is the associated parametrization such that $x_e(u) = 0$ and $x_e(v) = L_e$.

It is obvious that any minimizer of $U_\nu$ will also be a minimizer of $\widetilde{U_\nu}$. Furthermore, $\widetilde{U_\nu}$ can have a minimal value inside an edge, only if it is constant on $e$, namely if $U_\nu(u) = U_\nu(v)$. In this case, choosing randomly one adjacent vertex of such a minimzer, gives a barycenter of the discrete graph $G$.

### 2.10.3 Sampling from a law

Using the same notations and assumption as in sub-section 2.10.1, suppose that instead of being interested in finding the minimum of the function $\mathcal{U}_\nu$ (defined in (2.61)), we rather sample from a probability measure $\mu_\beta$, whose density is of the form:

$$\mu_\beta(x) = \frac{e^{-\beta \mathcal{U}_\nu(x)}}{Z_\beta}, \qquad \text{for some } \beta > 0. \tag{2.76}$$

In order to do that, we define the a homogenized simulated annealing at a fixed temperature $\beta$:

$$\begin{cases} X_0 \in \Gamma_G \\ dX_t = -\beta \nabla \mathbf{f}_t(X_t, Y_{N_t^\alpha})dt + dB_t, \end{cases} \tag{2.77}$$

where $B_t$ is a Brownian motion and the process $(N_t)_{t\geq 0}$ is a Poisson process of intensity $\alpha$ that dictates the rate at which we use the sequence $(Y_n)_{n\geq 0}$ of independent random variables distributed according to $\nu$.

Denote $J_t$ the relative entropy of $n_t$, the distribution of $X_t$, with respect to the target law $\mu_\beta$:

$$J_t := KL(n_t || \mu_\beta) = \int_{\Gamma_G} \log\left[\frac{n_t(x)}{\mu_\beta(x)}\right] dn_t(x). \tag{2.78}$$

**Theorem 6.** *If $\alpha_t$ is such that $\alpha_t / \log^5 t$ goes to $0$ as $t$ goes to infinity, the relative entropy $J_t$ converges to zero:*

$$\lim J_t = 0 \qquad \textit{as} \qquad t \longrightarrow +\infty.$$

We do not include the proof since it follows the same steps as the ones described in Section 2.10.1. We only show why we have different restrictions on the rate at which $\alpha_t$ goes to infinity. Let $I_t$ be defined as before:

$$I_t := \int_{\Gamma_G} \left[ \int_V \log \frac{m_t(y|x)}{\nu(y)} m_t(y|x)dy \right] .dn_t(x),$$

where $m_t(y|x)$ is the conditional law of $y$ given $x$ at time $t$. Then the differential inequalities corresponding to (2.67) and (2.68) lead to the following system of differential equations:

$$\begin{cases} \partial_t J_t \leq -\varepsilon J_t + 2\|\nabla_x \mathbf{f}_t\|_\infty \beta^2 I_t \\ \partial_t I_t \leq -\partial_t J_t - \alpha_t I_t + \frac{3}{2}\beta^2 \|\nabla_x \mathbf{f}_t\|_\infty^2 + \frac{\beta}{2}\|\Delta_x \mathbf{f}_t\|_\infty . \end{cases} \tag{2.79}$$

We introduce once more a function $K_t$, of the form $K_t = J_t + k_t I_t$, where this time $k_t$ becomes:

$$k_t = \frac{2\|\nabla_x \mathbf{f}_t\|_\infty^2 \beta^2}{\alpha_t + 2\|\nabla_x \mathbf{f}_t\|_\infty^2 \beta^2 + \varepsilon/2}$$

The inequality system (2.79) implies that, for $t$ large enough:

$$\partial_t K_t \leq -\frac{\epsilon}{2} K_t + \eta_t, \qquad \text{with} \qquad \eta_t = 2k_t \left( \beta^2 \|\nabla_x \mathbf{f}_t\|_\infty^2 + \beta \|\Delta_x \mathbf{f}_t\|_\infty \right) \qquad \text{and } \varepsilon > 0.$$

Thus $J_t$ goes to zero, as soon as $\eta_t$ does so. Since we have seen that $\|\mathbf{f}_t\|_\infty$ can be bounded by a finite constant, a sufficient condition is:

$$\frac{\|\Delta_x \mathbf{f}_t\|_\infty}{\alpha_t} \longrightarrow 0 \quad \text{when} \quad t \to +\infty. \tag{2.80}$$

We have seen that under our assumptions, $\|\Delta_x \mathbf{f}_t\|_\infty$ is at most of order $\log^5 t$, and this concludes the proof. □

**Remark 4.** *If the probability distribution we want to sample from admits a density with respect to the normalized Lebesgue measure that is in the generator's domain, a sufficient condition for the convergence in law of the process towards $\mu_\beta$ is to have $\alpha_t \longrightarrow +\infty$ (at any rate).*

# Chapter 3

# Barycenter estimation on very large graphs: a heuristic approach

One important difficulty in dealing with graphs is that many of the existing networks have an extremely large number of vertices and edges, making most algorithmic methods inefficient or even unsuitable due to the computational cost. This problem is very common in most areas of interest regarding graphs, as representations, community detection, *etc.* When trying to estimate a graph's barycenter, due to memory cost, the algorithm proposed in Chapter 2 cannot be used on current desktop computers for very large graphs (number of vertices $N > 10^5$). In this chapter we present a heuristic method that aims to overcome this issue and further more to reduce the computational time.

The memory issue in estimating the Fréchet mean for very large graphs is due to the distance matrix whose size increases as $N^2$. To overcome this, our strategy was inspired by the property of geometrical decomposition of the barycenter in a Euclidean space. The main idea is to partition the graph into subgraphs on which the estimation of the barycenter becomes far cheaper and then to create a new graph using these estimations. Obviously the barycenter defined on graphs does not have the same properties as in the Euclidean case and thus our method is a little more involved. A detailed description and an intuitive explanation of our method are given in Section 3.1. The focus in the present work is not on how we partition the graph, but rather on the strategy of estimating the barycenter, supposing that a proper partition is given. However, in Section 3.2 we briefly present some of the existing methods for graph partitioning and clustering and the methods we chose. Section 3.3 is dedicated to simulation results and numerical insights. Finally we present a short conclusion and some potential developments.

**Note To Reader** This chapter consist in a joint work with Laurent Risser, currently under finalization.

## 3.1 Multiscale Graph Center Estimation

### 3.1.1 Strategy: *Divide et Impera*

As mentioned before, our method is motivated by the property of geometrical decomposition of the barycenter in a Euclidian space and is based on a *Divide and conquer* strategy. It is well known that for $n$ points, $(A_i)_{i=1\ldots n}$ of an affine space and an associated sequence of scalars $(a_i)_{i=1\ldots n}$ of non-null sum, the barycenter is defined as the only point $G$ such that:

$$\sum_{i=1}^{n} a_i \overrightarrow{GA_i} = \vec{0}, \quad G =: \operatorname{bar}\left((A_i, a_i)\right)_{i=1\ldots n}.$$

Suppose $(a_i)_{i=1\ldots n}$ is partitioned into two sets of scalars, each of non-null sum. Let $I, J$ denote the corresponding set of indexes and $G_I = \operatorname{bar}\left((A_i, a_i)\right)_{i\in I}$, $G_J = \operatorname{bar}\left((A_i, a_i)\right)_{i\in J}$. The decomposition property states that the barycenter of the $n$ points is the barycenter of the two sub-barycenters, meaning:

$$\operatorname{bar}\left((A_i, a_i)\right)_{i=1\ldots n} = \operatorname{bar}\left((G_I, \sum_{i\in I} a_i), (G_J, \sum_{i\in J} a_i)\right). \tag{3.1}$$

This property can be iterated multiple times and still holds for $k$ partitions of this type, $k \leq n$.

If one might want to use a similar strategy for a graph, *a valid partition* $(C_i)_{i=1\ldots k}$ of the vertex set $V$ needs to satisfy the following conditions:

- The subsets $C_i$ are disjoint and their union contains all the vertices, $V = \bigsqcup_{i=1}^{k} C_i$;
- The weight associated to each subset is non-null: for all $0 \leq i \leq k$, $\nu(C_i) \neq 0$. In our framework this is implicit since $\nu$ charges each vertex, $\nu(y) > 0$, $\forall y \in V$ ;
- For each part $C_i$ the associated sub-graph $G_i$ is connected. This is the most important condition since the barycenter is not defined for non-connected graphs.

**Definition 5.** *For $C_i \subset V$, a subset of the vertex set, we call associated sub-graph $G_i$ a graph $G_i = (C_i, E_i)$ formed by all edges of the initial graph $G$, connecting two points of $C_i$. In other words, the set of edges of $G_i$ is:*

$$E_i = \{e = (e_-, e_+) \in E \;| e_-, e_+ \in C_i\}. \tag{3.2}$$

We say that two disjoint subsets $C_i, C_j \subset V$ are neighboring clusters of the graph $G = (V, E)$, and denote it $C_i \sim C_j$ if there exists a pair of vertices $v_i \in C_i$ and $v_j \in C_j$ that are neighbors in $G$:

$$C_i \sim C_j \iff \exists\; v_i \in C_i, v_j \in C_j \text{ such that } (v_i, v_j) \in E. \tag{3.3}$$

A first remark is that, given a valid partition of the vertex set, the union of all associated edges does not contain all the edges of the initial graph, $E \neq \bigsqcup_{i=1}^{k} E_i$. For us, this implies a loss of information that will be taken into account when constructing the new graph from the barycenter estimations obtained on the partition. There exist partitioning algorithms that minimize the number of excluded edges, such a partition being generally referred to as minimal cut. We will give more details about different methods of clustering in Section 3.2. For now we content ourselves of considering a valid partition $(C_i)_{i\leq k}$, this being enough to present our method.

Let us establish a few notations for the sequel:

- $G = (V, E)$ is an undirected metric graph. Each edge $e \in E$ has a length $l_e > 0$ and $d$ is the associated distance.
- $\nu$ is a probability measure on $V$ that charges all points: $\forall v \in V$, $\nu(v) > 0$.
- $(Y_n)_{n\geq 0}$ is a sequence of independent random variables distributed according to $\nu$.
- $\mathsf{P} = (C_i)_{i\leq k}$ is a valid partition of the vertex set $V$.

### 3.1.2 Auxiliary procedures

In order to emphasize the main steps of the algorithm we define a few helpful functions (procedures), that will be used in this section.

GCE EstimateGraphCenter is a function that takes as input $G = (V, E)$ and $(Y_n)_{n\geq 0}$ and returns a vertex $x^*$, the estimated barycenter of $G$, using Algorithm 3 presented in Chapter 2, with the parameters proposed in Section 2.4.

CSG CreateSubGraph is a function that takes as input a subset of vertices $C_i$, the complete set of edges $E$ and a sequence $(Y_n)_{n\geq 0}$.
It returns the associated subgraph $G_i = (C_i, E_i)$, defined in Definition 5 and $(Y_n^i)_{n\geq 0}$ a sequence of independent random variables distributed according to $\nu_i$, a probability measure on $C_i$ defined by:

$$\forall v \in V \quad \nu_i(v) = \frac{\nu(v)}{\nu(C_i)}. \tag{3.4}$$

CUG CreateUpscaleGraph is a function that takes as input $G = (V, E)$, $(Y_n)_{n\geq 0}$, $(C_i)_{1\leq i\leq k}$ and a list of $k$ nodes $\mathbf{V} \subset V$ such that for all $v \in \mathbf{V}$ there exits only one subset $C_i$ that contains it.
The function returns a new undirected weighted graph $\mathbf{G}$ and a sequence $(\mathbf{Y}_n)_{n\geq 0}$ distributed according to $\nu_{\mathbf{G}}$, a probability measure on its vertex set $\mathbf{V}$.

The probability associated to each node is the total probability of the cluster that contains it:

$$\forall v \in \mathbf{V}, \quad \nu_{\mathbf{G}}(v) = \nu(C_i), \qquad \text{where } C_i \text{ is such that } v \in C_i. \tag{3.5}$$

The edge set of the new graph is defined by:

$$\mathbf{E} = \{(v_i, v_j) \text{ with } v_i, v_j \in \mathbf{V}, v_i \in C_i, v_j \in C_i \text{ and } C_i \sim C_j\}, \tag{3.6}$$

there exists an edge between two vertices if and only if their respective clusters are neighboring clusters and the length of each new edge is defined as the distance between its extremities in the initial graph $G$.

One should observe that while EstimGraphCenter represents a stochastic approximation procedure, the other two, CreateSubGraph and CreateUpscaleGraph, are completely deterministic. We illustrate the effect of the latter on a simple graph in Figure 3.1.

#### Estimate Graph Center

We need to make several remarks regarding the item GCE. Algorithm 3, introduced in Section 2.2.3, is defined on a quantum graph. However, in this section, most of the time, this continuous structure is not important and thus we consider that the function EstimateGraphCenter takes as input a discrete graph and creates by itself the quantum structure on which the generated process evolves. This is rather natural, seeing that this is what we actually do in practice. Furthermore, Algorithm 3 (Section 2.2.3) determines a point $\widetilde{x}^*$ on a quantum graph (uniformly among all Fréchet means of the graph), but EstimateGraphCenter returns the closest vertex to $\widetilde{x}^*$ in the sense defined in Chapter 2, subsection 2.4.

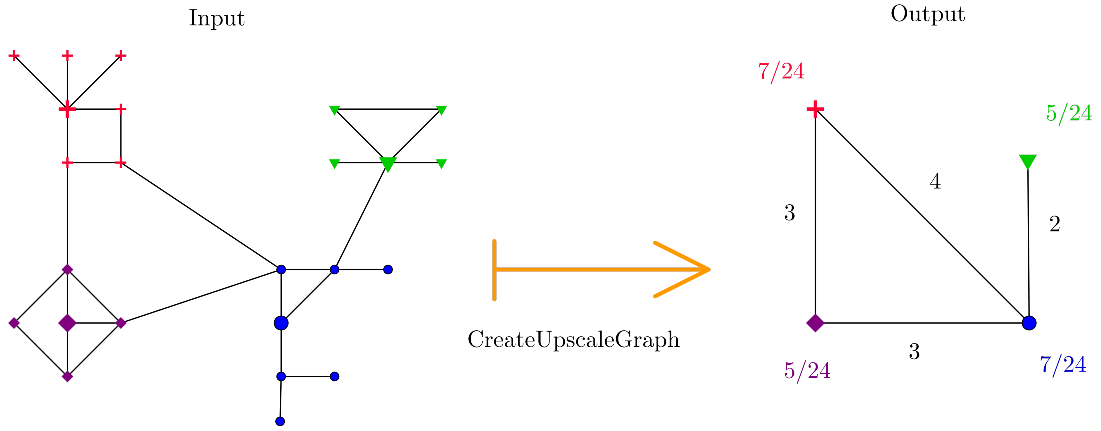


Figure 3.1: On the left hand side we have an initial graph $G$ partitioned in 4 communities (the vertices of each community are represented by different symbols and colors). We consider $\nu$ as the uniform probability on its vertex set and all edges have length 1. We take a subset $\mathbf{V}$ formed of vertices represented in a larger size. The upscale graph created by the Cug procedure is represented on the right hand side, along with the length of the new edges and the new probability corresponding to each vertex. We have chosen to represent the distribution $\nu_{\mathbf{G}}$, instead of a sequence $(\mathbf{Y}_n)_{n\geq 0}$, because it is easier to visualize.

### Create sub-graph

Regarding Csg, it is worth mentioning that the probability $\nu_i$ defined in Equation (3.4) corresponds to the initial probability $\nu$ conditioned upon being in $C_i$: $\nu_i(v) = \nu(v|v \in C_i)$. In other words, if $Y^i \sim \nu_i$ and $Y \sim \nu$ are two independent variables, we have:

$$\forall v \in C_i, \quad \mathbb{P}(Y^i = v) = \mathbb{P}(Y = v|Y \in C_i).$$

This tells us how to define the sequence $(Y^i)_{n\geq 0}$. Let $\varphi_i$ be a random sequence such that:

$$\varphi_i(0) = \inf\{m \geq 0|Y_m \in C_i\} \text{ and } \varphi_i(n) = \inf\{m > \varphi(n-1)|Y_m \in C_i\}.$$

We now simply define: $(Y^i_n)_{n\geq 0} = (Y_{\varphi(n)})_{n\geq 0}$. From a practical point of view this does not pose any problem. It only means that we reject any realization $Y_n$ that is not in the wanted subset $C_i$.

### Create upscale graph

We would also like to make some remarks regarding the item Cug. The definition of the associated probability measure for the upscale graph in (3.5) is the analog of summing the scalars in the affine case in (3.1). Accessing independent random variables $(Y_n)_{n\geq 0}$ distributed according to $\nu$, we can easily define another sequence $(\mathbf{Y}_n)_{n\geq 0}$ of i.i.d. random variable of law $\nu_{\mathbf{G}}$:

$$\mathbf{Y}_n = c_i \in \mathbf{V} \cap C_i \quad \text{if and only if} \quad Y_n \in C_i. \tag{3.7}$$

From the simulation point of view, when we have access to $(Y_n)_{n\geq 0}$, Equation (3.7) means that every time a node in a cluster $C_i$ is given by the sequence, we see it as the unique node $c_i$ that represents the cluster in the upscale graph.

### 3.1.3 Upper Scale Approximation

Computing the barycenter of each community, creating a new graph as in CUG and then computing its barycenter is a natural extension of the decomposition of the barycenter in the Euclidean case. This idea is at the core of Upper Scale Approximation strategy described in Algorithm 4.

**Algorithm 4:** Upper Scale Approximation (USA)

```
  Data: A graph $G = (V, E)$, a valid partition $(C_i)_{i \leq k}$, a sequence of nodes $(Y_n)_{n \geq 0}$.
1 ListOfCenters=[]
2 for $i = 1 \ldots k$ do
3     Create the associated subgraph $G_i$ from the vertex set $C_i$:
        $(G_i, \nu_i) = \text{CreateSubGraph}(C_i, E)$
4     Estimate the barycenter of this subgraph: $c_i = \text{EstimateGraphCenter}(G_i, \nu_i)$
5     Add $c_i$ to ListOfCenters
6 end
7 Create a new graph $\mathbf{G}$ and a corresponding probability measure using the centers:
    $(\mathbf{G}, \nu_{\mathbf{G}}) = \text{CreateUpscaleGraph}(G, \nu, (C_i)_{1 \leq i \leq k},\ \text{ListOfCenters})$
8 Output: EstimateGraphCenter$(\mathbf{G}, \nu_{\mathbf{G}})$.
```

If the chosen partition has a specific meaning, this procedure has an interest on its own, allowing us to study some larger scale properties of the graph. For example if each cluster $C_i$ represents a community, this is a natural way of defining a central community.

**Definition 6.** *Let $\mathsf{P} = (C_i)_{i \leq k}$ be a valid partition of a graph $G = (V, E)$, $\mathbf{V}$ a set such that $\forall i \leq k$ there exists an unique $c_i \in \mathbf{V} \cap C_i$ and each $c_i$ is a Fréchet mean of the associated subgraph $G_i$ with respect to $\nu_i$, the corresponding probability defined in Equation (3.4). We call $\mathsf{P}$-central community a set $C_i$ that corresponds to a vertex $c_i$, a Fréchet mean of the graph $\mathbf{G}$ with respect to the probability $\nu_{\mathbf{G}}$, defined in Equation (3.5).*

The Upper Scale Approximation presented in Algorithm 4 determines the representative point of a central community for a given partition. Since the graph's barycenter does not have the same properties as the euclidean one, the result of this procedure is not necessarily a Fréchet mean of the initial graph. However, for reasonable partitions, one might expect the barycenter to be not far from the central community obtained on the upscale graph.

#### Illustration on the Parisian Subway Graph

We illustrate this strategy on the Parisian subway graph introduced in Section 2.4.3. Obviously, this graph being very small, if our purpose is to estimate its barycenter, from a computational point of view, this method has no interest. However its advantage is that it can be easily visualized, giving thus a better intuition on the principles, advantages and disadvantages of the method. In Figure 3.2, one can see the complete Parisian subway network, the initial partition we used and also the true barycenter. Each edge represents a connection between two stations, the length of an edge is the time needed to go from one station to the other, and the weight of each vertex is given by the number of people using the station. In Figure 3.3 we represent the graph created on line 7, of the up-scaling Algorithm 4, where the estimation in line 4 has been replaced by the computation of the true barycenter of each cluster. One can see that the barycenter of the initial graph is not included in this new graph, and thus has no chance at being its center. However, the Fréchet mean on this coarsened version of the network, is the station *Opera*, that represents the cluster of *Chatelet*, the barycenter of the complete graph.

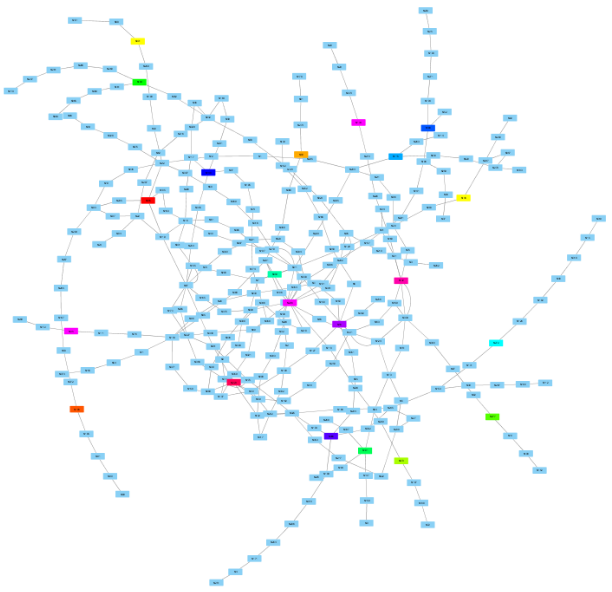


Figure 3.2: Complete graph of Parisian Metro. The colored nodes represent centers of their clusters.

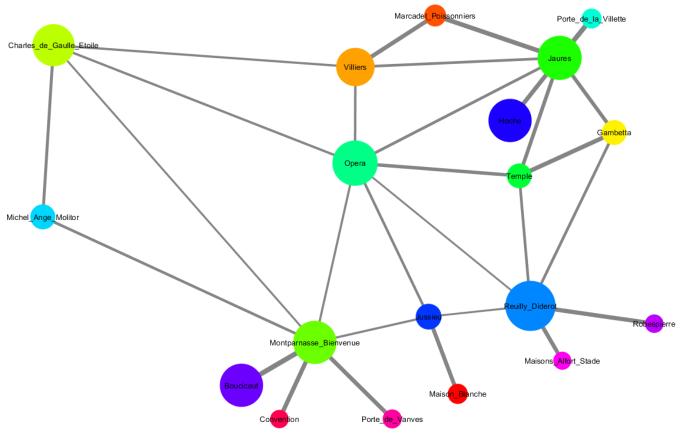


Figure 3.3: Up Scale Graph Parisian Metro. The size of the nodes represents their weight and the width of the edges is inversely proportional to the time needed to go from one station to the other.

### 3.1.4 Multiscale Graph

If all we are interested in is a rough notion of centrality, finding the central cluster might be enough. To gain precision, we decided to add another step to our method, reinserting the central cluster in the upscaled graph.

Consider a graph $G = (V, E)$ and a probability measure $\nu$ on $V$. For a valid partition of the vertex set $\mathsf{P} = (C_i)_{i \le k}$, let $(G_i, \nu_i)_{i \le k}$ denote the associate sub-graphs with their respective probabilities measures, defined in Csg. With the notations introduced in Cug, let $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ be an up-scale version of $G$ corresponding to the partition $\mathsf{P}$. In what follows we define $\mathbf{G}_M = (V_M, E_M)$ the *multi-scale version* of $G$ with respect to $(\mathbf{G}, C)$, where $C$ is an element of $\mathsf{P}$, and $\nu_M$ the corresponding probability measure. The definition of the vertex set and the associated probability are straightforward. $V_M$ contains the nodes of $\mathbf{G}$ and $C$:

$$V_M = \mathbf{V} \cup C, \tag{3.8}$$

and $\nu_M$ redistributes the mass of $C$ to its nodes, while leaving the others values of $\nu_{\mathbf{G}}$ unchanged:

$$\nu_M(v) = \begin{cases} \nu(v) & \text{if } v \in C \\ \nu_{\mathbf{G}}(v) & \text{if } v \in \mathbf{V} \end{cases} \tag{3.9}$$

The edge set $E_M$ contains the edges of $\mathbf{G}$, except those that were added to $c$ , the node that represents the cluster $C$ in the up-scale graph, and all internal edges of $C$. On top of that we add new edges going from boundary of $C$ to the nodes corresponding to its neighboring clusters:

$$\text{BorderEdges}(C, \mathrm{P}) = \{(v, c_i)|\ v \in C, \exists v_i \in C_i \text{ with } (v, c_i) \in E\} \tag{3.10}$$

The length of such an edge is defined as the initial distance between its extremities in the complete graph. Now the set of edges $E_M$ can be written as:

$$E_M = E_C \cup (\mathbf{E} \setminus \{e|e \sim c\}) \cup \text{BorderEdges}(C, \mathrm{P}), \tag{3.11}$$

where $e \sim c$ means that $c$ is a vertex of $e$. Again, from a simulation point of view, constructing a sequence of random variables distributed according to $\nu_M$ is easy once we have access to $(Y_n)_{n\geq 0}$ of law $\nu$. We simply set:

$$Y_n^M = \begin{cases} Y_n, & \text{if } Y_n = v, \text{ with } v \in C \\ c_i, & \text{if } Y_n = v, \text{ with } v \in C_i \end{cases} \tag{3.12}$$

Before presenting the new algorithm in Algorithm 5, we introduce one more function:

CMG CreateMultiscaleGraph is a procedure that takes as input a graph $G = (V, E)$, an up-scale version of it $\mathbf{G} = (\mathbf{V}, \mathbf{E})$, an element $c \in \mathbf{V}$, a subset $C \subset V$ such that $c \in C$, and returns a multi-scale graph $G_M = (V_M, E_M)$, where $V_M$ and $E_M$ are defined in Equation (3.8) and Equation (3.11) respectively.

**Algorithm 5:** Multi-Scale Approximation

```
Data: A graph $G = (V, E)$, a valid partition $(C_i)_{i\leq k}$, a sequence of nodes $(Y_n)_{n\geq 0}$.
Create ListOfCenters, lines 1-6 of Algorithm 4.
Create a new graph $\mathbf{G}$ and a sequence $(\mathbf{Y})_{n\geq 0}$, using the centers:
    $(\mathbf{G}, (\mathbf{Y})_{n\geq 0}) = \text{CreateUpscaleGraph}(G, (Y_n)_{n\geq 0}, (C_i)_{1\leq i\leq k},\ \text{ListOfCenters})$
Estimate the center of this new graph:
    $c^* = \text{EstimateGraphCenter}(\mathbf{G}, (\mathbf{Y})_{n\geq 0})$
Identify $C^*$, the cluster of $c^*$.
Create $G_M$ a multiscale version of the graph $G$, using $\mathbf{G}$ and $C^*$:
    $G_M = \text{CreateMultiscaleGraph}(G, \mathbf{G}, c^*, C^*)$
  Create a new sequence of nodes $(Y_n^M)_{n\geq 0}$ as described in Equation (3.12) using $(Y_n)_{n\geq 0}$.
Output: EstimateGraphCenter$(G_M, (Y_n^M)_{n\geq 0})$.
```

We used the CreateMultiscaleGraph procedure, defined in CMG, on the Parisian metro network, merging the up-scale graph obtained using the centers of each cluster (represented in Figure 3.3), with its central cluster. Figure 3.4 presents a visualization of the result. We also computed the barycenter of this new multi-scale graph, with respect to the probability measure $\nu_M$ defined in Equation (3.9). This *zoom in* procedure applied to the central cluster is enough to reestablish the station *Chatelet* as barycenter. Of course we do not claim that in general a Fréchet mean of the graph ,obtained following the multi-scale procedure, is a Fréchet mean for the initial graph. However, this *zoom out/ zoom in* procedure is somehow intuitive and seems to work well in practice. For details about our experiments see Section 3.3.

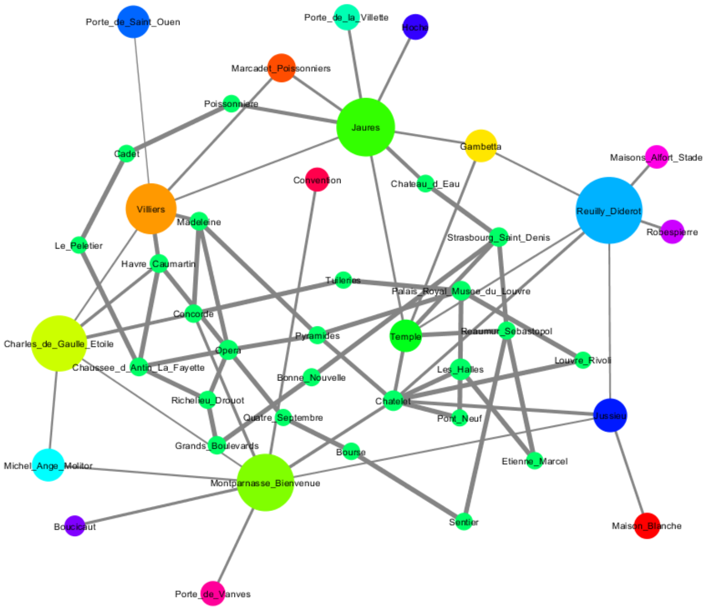


Figure 3.4: Multi-Scale Graph Parisian Metro. The color of the nodes represents their cluster (corresponding to the colors used in Figure 3.3) and the width of the edges is inversely proportional to the time needed to go from one station to the other.

## 3.2 Graph Partitioning and Clustering

Considering the size of the networks we usually deal with nowdays, understanding the structure of a graph (local and global) becomes more and more challenging. This issue generated an important field in graph theory: clustering (graph partitioning). This problematic is extremely vast and here we will only mention some of the most commonly used techniques and give a brief presentation of the methods that are of interest in our present work. A quite detailed overview of clustering methods is provided in [33].

There are two different approaches when it comes to dividing a network in subsets: block modeling, also called community detection (associated more with sociology, biology or physics), and graph partitioning (studied mostly in computer science and related fields), for details see [105]. Seen from afar they have a common goal, but a closer look reveals that their purpose is quite different.

Networks often model interactions between individuals or objects. Whether or not these interactions are driven or encouraged by certain characteristics of the individuals is a question we might ask ourselves when studying the network. Also, if the answer is yes, then one might wonder what properties of the vertices encourage interactions. A first step in understanding all this is community detection: given a graph we want to know if there exists a natural partition of its vertex set, considering that we have no restrictions on the number of subsets or their size (apart the fact that they form a partition). In graph theory, communities represent groups of vertices that have a higher density of edges within each group than between them. To be more precise, for a division to be considered representative, it is not enough to have few edges between groups, but to have less than the expected number of edges between them. The same holds for the inter-connectivity of clusters, the vertices inside a group need to be more connected then expected, in order for it to be considered as a community. Thus, it is important to quantify the difference between existing edges and edges put there at random. Using this criteria as a way to measure the quality of a clustering, Newman and Girvan [107] introduced the notion of modularity.

Modularity is proportional to the difference between the number of edges falling within groups minus the expected number in an equivalent random network. What an equivalent random network is, depends on the graph model. The most commonly used model is the random graph with prescribed expected degrees (also called the Chung-Lu model [45]). In what follows, this is the reference model we consider.
Consider a finite undirected graph, $G = (V, E)$. For each vertex $i$ we denote $d_i$ its degree. Let $m$ be the number of edges of the graph, $m = |E|$, and $A$ its adjacency matrix ($A_{ij}$ represents the number of edges between $i$ and $j$). The expected number of edges between two vertices $i$ and $j$ is:

$$\frac{d_i d_j}{2m}.$$

The mathematical formulation of the modularity for a given partition $(C_i)_{1 \leq i \leq k}$ is then:

$$Q = \frac{1}{4m} \sum_{i,j} \left( A_{ij} - \frac{d_i d_j}{2m} \right) \delta(C_i, C_j), \tag{3.13}$$

where $\delta(C_i, C_j)$ is the classical Dirac measure. The factor $1/4m$ in Equation (3.13) is only conventional, but it is included in the definition introduced by Newman and Girvan [107]. When modularity is close to zero, it means that the graph is close to what we expect from a random network. High values of $Q$ are associated with a good partition and looking for partitions with high modularity seems to be an effective way of detecting communities [71].
Maximizing the modularity over the whole set of possible partition is a NP-complete problem [17]. Its complexity is intuitive since the number of such partitions increases extremely quickly with $n$. A possible approach to this problem is the use of spectral methods [106], another one is hierarchical clustering [105]. Here we briefly present the second one. Hierarchical clustering is of special interest in our present work because it enables a greedy optimization technique and it is widely used for huge graphs. This method can also be used considering other similarity features on the vertex set (a distance for example), that imply the optimization of other functions on the set of all possible

partition of $V$, denoted $\mathcal{P}$. We present the main ideas of this method in Algorithm 6.

**Algorithm 6:** Hierarchical Clustering

```
  Data: A graph $G$ and a function $F : \mathcal{P} \to \mathbb{R}$
1 Each vertex is the sole member of a community: $\mathsf{P}_0 = \{\{v\}, v \in V\}$
2 for $i = 1 \ldots n-1$ do
3     Merge into one community $C_{\max}$ the communities $C_i$ and $C_j$ that maximize $F$
      over all the possible fusions of two groups of $\mathsf{P}_k$:
4         $$\mathsf{P}_{k+1} = (\mathsf{P}_k \setminus \{C_i, C_j\}) \cup C_{\max}$$
5     Update distances between communities.
6 end
7 Output: Clustering Dendrogram: $(\mathsf{P}_\mathsf{k})_{k \leq n-1}$
```

A first remark is that in practice, instead of creating the complete dendrogram, a very popular stopping criteria is: 'Stop when there is no merging that increases the function $F$: given $\mathsf{P}_k$, for all possible $\mathsf{P}_{k+1}$, $F(\mathsf{P}_{k+1}) - F(\mathsf{P}_k) < 0$'.

Going back to modularity, for a partition $\mathsf{P} = (C_i)_{1 \leq i \leq k}$, denote $\Delta Q_{i,j}$ the difference between the modularity of the initial partition $\mathsf{P}$ and the modularity of the partition obtained by merging $C_i$ and $C_j$. When applying Algorithm 6 for modularity, the instruction of line 4 becomes: join together the two communities such that the corresponding new partition produces the highest increase in modularity. Merge $C_i$ with $C_j$, where $(i, j) \in \arg\min \Delta Q_{i,j}$.

Although modularity is widely used as a way to measure the quality of a clustering, many other clustering heuristics have been developed using different approaches based for example on spectral methods, graph bisection, betweenness or Markov Chains and random walks [116]. In what follows, we briefly present a method that falls under the latter category, called Markov Cluster Algorithm. This is one of the methods we used in practice for our simulations.

The Markov Cluster Algorithm (MCL) uses random Markov walks on graphs in order to uncover a clustering structure. The clustering is obtained by performing, in a repeated manner, two types of operations on the probability transition matrix of a random walk on the graph and observing its evolution in large times. The intuitive justification of this approach is that a random walk on the graph will seldom pass from one natural cluster to another and when it enters a natural cluster, it will have a tendency to spend a large amount of time in it. This method was introduced by Stijn van Dongen and for a detailed description of it and related results we refer the reader to [121].
We include here only a brief description. The algorithm takes as initial entry a stochastic matrix $M$ associated to a graph $G$ (the $m_{i,j}$ entry represents the probability to go from node $i$ to node $j$) and an infinite sequence $(e_k, r_k)_{k \in \mathbb{N}}$, where $(e_k)_{k \geq 0}$ are strictly positive integers and $(r_k)_{k \geq 0}$ are strictly positive real numbers. The stochastic matrix $M$ is altered by two operations applied one after the other in a repetitive way. Thus the algorithm returns an infinite sequence of matrices $(M_k)_{k \geq 1}$ corresponding to the probability transitions of an inhomogenous Markov chain. The used operations are called expansion and inflation. Expansion of a matrix $M$ with parameter $e \in \mathbb{N}^*$ corresponds to raising $M$ to the power $e$. Inflation of $M$ with parameter $r > 0$ corresponds to raising each entry to the power $r$ and then re-normalizing the obtained matrix to make it stochastic again. This procedure is resumed in Algorithm 7, where, for simplicity, we consider the sequence of parameters to be constant, $e_k = 2$ and $r_k = r > 0$, $\forall i \in \mathbb{N}$.

**Algorithm 7:** Markov Cluster Algorithm

```
  Data: A graph $G$ and a parameter $r > 0$.
1 Add loops to $G$.
2 Construct an associated stochastic matrix $M$.
3 while change do
4     $M_1 = M \times M$
5     $M = \text{Inflation}(M_1, r)$
6     change= $M - M_1$
7 end
8 Output: Graph corresponding to $M$.
```

Line 1 of Algorithm 7 corresponds to setting a positive probability of returning in one step at each vertex and is motivated by spectral considerations on the matrix $M$. Line 4 corresponds to the extension procedure and for a more general case it can be replaced by $M_1 = M^k$. This represents the probability transition of the random walk in $k$ steps. The intuitive interpretation of its role is the following: in a cluster the vertices are highly connected, and so, inside it there are a lot of ways of going from one node to another, generating thus a relatively large probability transition. Inflation (line 5) boosts the probabilities inside the clusters and diminishes the others. Increasing its parameter increases the granularity. Equilibrium is reached when the algorithm reaches a matrix invariant under both operations. Such a matrix corresponds to a graph formed by several connected (directed) components that define thus a clustering.

Although the granularity of the clustering can sometimes be influenced by the user, the number and the size of communities are generally determined by the network itself and not by the user. The priority in graph partitioning is fast computational time and its aim is to split the graph into smaller ones that can be dealt with in parallel by different processors. To maximize efficiency, the number and sizes of communities are chosen in advance according to the number of available processors and their technical performances. As opposed to clustering, this technique does not preoccupy itself with whether there is an underling community structure or not.

Since the main goal of our present work is to improve computational features, we also tested our method on a preliminary division of the network obtained using a graph partitioning method based on a agglomerative heuristic whose principles are similar to the one presented in Algorithm 6. The aim here is not to optimize a function $F$ on the set of possible partitions of the vertex set, but to form a fixed number of groups of similar sizes. The similarity with Algorithm 6 lays in the 'bottom up' approach: we start with groups of size 1 and repeatedly merge them until a stopping criteria is met.

## 3.3 Numerical Results

We first tested our method on the Parisian subway graph represented in Figure 3.2, in order to measure its stability and its accuracy, since we have access to the true corresponding results. Then we tested its behavior on a large graph using two types of preliminary clustering. We have chosen a graph of 264.346 nodes and 733.846 edges of the crossroads in the New York urban area.

### New York Urban Area

As mention before, this graph has 264.346 nodes and 733.846 edges. Its nodes are the GPS coordinates of crossroads in a rather large New York urban area. We referred to it as such by convenience, but the area is not limited to the state of New York, see Figure 3.5. The graph can be found on the website of the Center for Discrete Mathematics and Theoretical Computer Science [1]. On the website it is mentioned that some gaps might exist and thus the graph does not necessarily contain all crossroads. The distance considered between two nodes is the physical one, and not the transit time. Furthermore, the graph is undirected, namely each street allows travel in both directions.

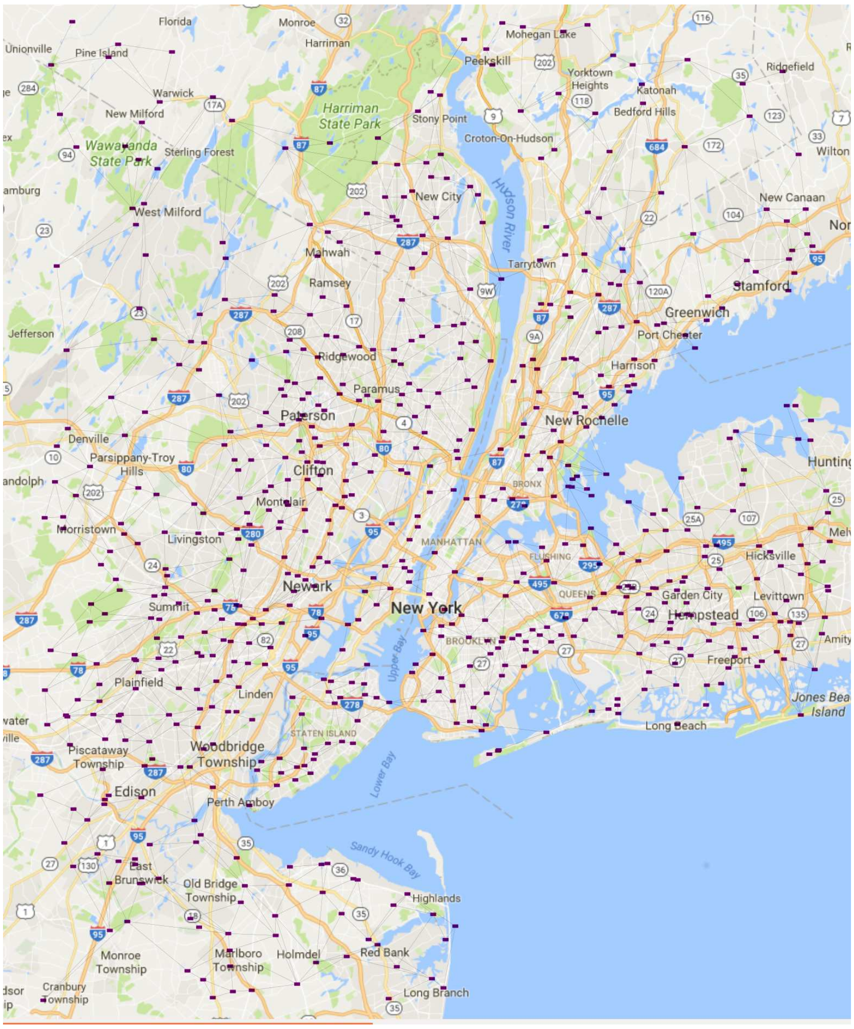


Figure 3.5: New York urban area. Image obtained using Cytoscape and ©Google Maps. Purple points represent estimated centers of the NG700 partition.

We have performed two types of preliminary clustering. One, based on a bottom up approach, meant to provide clusters of homogeneous size, and another, based on the MCL method described in Algorithm 7. The graph partition obtained with the first method has 700 clusters (from now on it will be referred to as GP700) and the second one has 1776 (we will referred to it as MCL12).

**Technical details** In terms of memory cost, this kind of graph can definitely not be handled with the method proposed in Chapter 2. A rough estimation suggests that the associated matrix distance would need around 360 GB of memory, whereas this new method employs far less resources, being of the order of 15GB (or less). The computational time is

[1] http://www.dis.uniroma1.it/challenge9/download.shtml

451 County Road 644
Hoboken
Empire State Building
Whitney Museum of American Art
Jersey City
New York
New York Stock Exchange
Parc Liberty State Park
Hoboken
Empire State Building
Whitney Museum of American Art
Jersey City
New York
New York Stock Exchange
Parc Liberty State Park

(a) GP700

(b) MCL12

Figure 3.6: Illustration of estimated centers before and after the multiscale analysis, using the itinerary option of ©Google maps. Centers estimated by USA are represented as a departing point (in white), while the centers provided by the MLT are represented as a destination, (pointed in red). In blue, we have an itinerary between the two.

also reasonable. Using the default parameters it takes, in average, 35 min for the GP700 and 1 h 30 min for the MCL12. The rapidity of the algorithm is, among others, due to the fact that the estimations of clusters' centers can be done in parallel. It is not surprising that the barycenter's estimation on GP700 is faster, since the clusters have a more homogeneous size.

Since the graph is too dense to visualize, we have chosen to use the GP700 partition in order to facilitate Figure 3.5. To be more precise, we have used the upper scale approximation procedure described in Algorithm 4 to form a new graph from the estimated centers of each cluster. A visualization using the GPS coordinates of the nodes was created with the aid of the Cytoscape software. This illustration was afterwards overlaid on a map of the area provided by ©Google Maps. The result is shown in Figure 3.5. The purpose of this figure is to give an idea of the area covered by the complete graph and not to show the exact position of each node in the upper scale graph.

In Figure 3.6 we represent the evolution between the center estimated using the USA (Algorithm 4) procedure and the center obtained using the multiscale analysis (MLT) described in Algorithm 5. We have chosen to do this using the itinerary option of Google Maps, in order to better emphasize a direction. Of course, the itineraries proposed by Google Maps do not necessarily correspond to a shortest path in the graph itself, since the application takes into account different restrictions like traffic, one way streets, *etc.*, that are not integrated into the graph's structure.

**Parameters** We tested our method using the following three sets of parameters. Let us first briefly recall the default parameters of Gce, presented in Section 2.4, since these parameters will be used as a reference.

$$T^\star_{\max} = 100 + 0.1N \quad S^\star = 1000 \quad \beta^\star_t = \frac{2\log t}{\mathcal{D}_G}, \tag{3.14}$$

where, $T^\star_{\max}$ represents the maximal time, $N$ the number of vertices and $S^\star$ defines how many observations $(Y_n)_{n\geq 0}$ are used between $T^\star_{\max}$ and $T^\star_{\max}-1$. $\mathcal{D}_G$ is the graph's diameter (the maximal distance between two points) and $\beta^\star_t$ is the inverse of the temperature schedule, namely the sizes of jumps that $X_t$ makes toward a new arrival of an observation $Y$, are proportional to it.

I The default parameters defined in 3.14.

II The second set is chosen as:

$$T^2_{\max} = 4T^\star_{\max} \quad S^2 = 2S^\star \quad \beta^2_t = 4\beta^\star_t$$

III The third set is defined as:

$$T^3_{\max} = 8T^\star_{\max} \quad S^3 = 4S^\star \quad \beta^3_t = 4\beta^\star_t$$

In Figure 3.7 and Figure 3.8 we illustrate the results obtained on the GP700 partition, respectively on the MCL12 partition, with the 3 stets of parameters described above (I,II,III). Experiments with the default parameters were repeated 5 times, whereas the other two only once. For each experiment, we represent the center of the central cluster obtained using the upscale method of Alogorithm 4 (these centers will be diamond shaped points) and the center obtained afterwards, with the multiscale procedure of Algorithm 5, when all nodes of the central cluster are taken into account. For both images, the violet diamond shaped points represent the centers obtained with the upscale approximation and the other points are centers obtained with the multiscale method. The blue points are obtained with the set of parameters I, the green ones with the set II and the red ones with III.

In both figures , on the right hand side we include a general view of the New York complete graph represented in Figure 3.5.

**Results on GP700.** Two clusters were chosen as central three times (each) and another one only once (this is the one in the north-east section). We can observe that the final centers present less variation, they are closer to each other and only two outliers are not concentrated on the same spot. Moreover, the appearance of a center in the Upper East Side, is due to the first approximation of the central cluster.

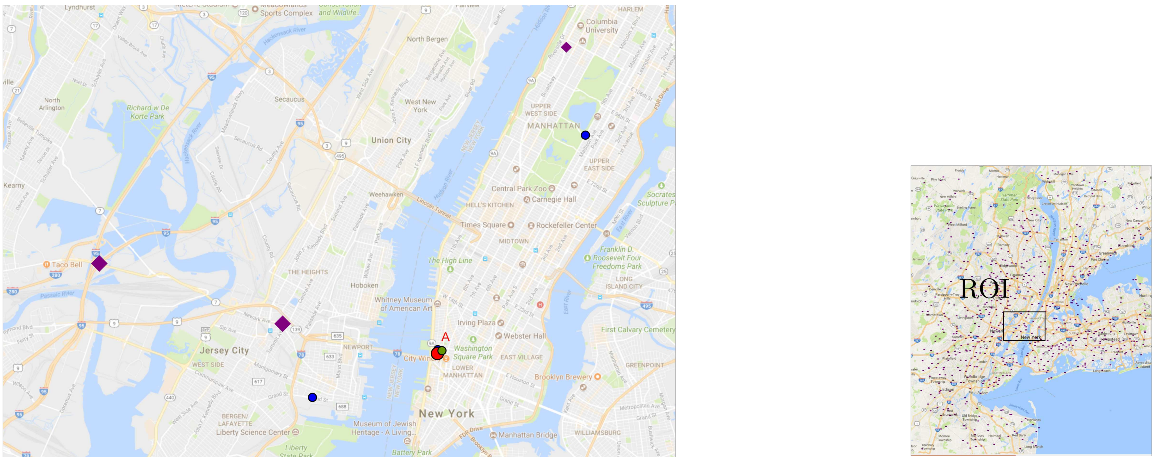


Figure 3.7: The left side image was obtained using Geogebra and ©Google Maps. On the left we have the region of interest (ROI) presenting the main results obtained on the GP700 partition. The red center, denoted $A$, was also obtained twice with the parameters set I.

**Results on the MCL12** The analog results obtained on the MCL12 partition are illustrated in Figure 3.8. We can observe the first step of the procedure was very robust, seeing that the result of the upscale algorithm was always the same. However, we observe a variation for the second part of the method, the multiscale analyses. Nevertheless, the results seem coherent since all nodes are on the north-east region of the upscale center and they seem attracted towards the Manhattan island.

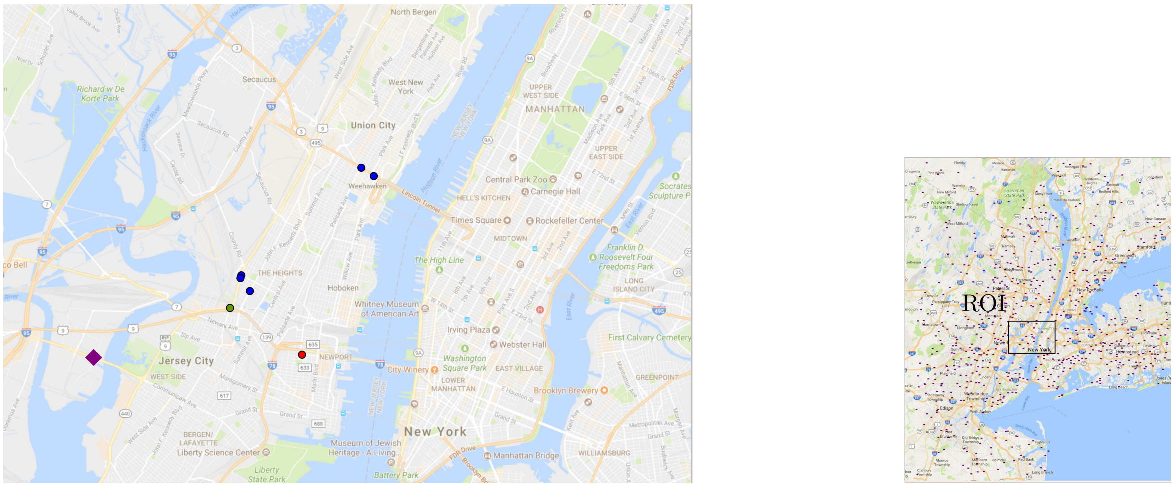


Figure 3.8: The left side image was obtained using Geogebra and ©Google Maps. On the right hand side we have a general view of the New York complete graph represented in Figure 3.5. On the left we have the region of interest (ROI) presenting the main results obtained on the MCL12 partition.

### Conclusions

When comparing Figure 3.7 with Figure 3.8, we can see that, as expected, the initial partition has an influence on the results, since the multiscale estimated centers tend to concentrate on different zones. However, we can also observe a coherence between the two sets of results. We can see that the centers of the upscale graphs are often quite close, especially if we consider that the approximation is made only on the centers of clusters. The multiscale centers of GP700 are almost always on the Manhattan island and we can remark that the estimated centers of the MCL12 partition, also tend to approach the island. This is summed up in Figure 3.9, where we have represented together the results for both partitions.

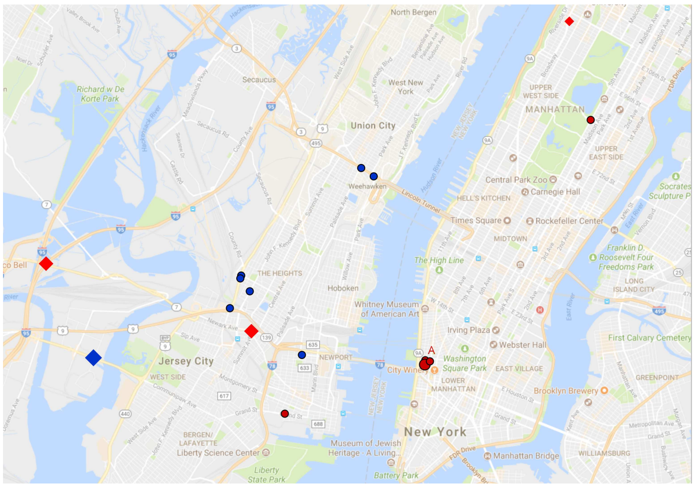


Figure 3.9: The image was obtained using Geogebra and ©Google Maps. The results obtained on the MCL12 are represented in blue and the results obtained on GP700 are represented in red.

The fact that we can observe a variability in the results is quite natural. Indeed, even though the GCE procedure performs very well with its default parameters, the probability of estimating all $k$ centers correctly decreases geometrically with respect to $k$. However, the influence of the small perturbation of size $\epsilon$ that might appear, can be explicitly computed and thus upper bounded, using the fact that the function $U_\nu$, used to define the barycenters, is Lipshitz continuous of constant $2\mathcal{D}_G$.

Although the method seems to behave rather well on the New York graph, in order to validate it, we need to test it on other graphs. It is expected that our method is well suited for graphs that have a structure similar to a grid, but it might need further adjustments for graphs that have a completely different structure (like social networks for example). In particular, a graph partitioning technique that does not take into account the graph's structure, is probably not well suited for the study of a graph that follows a stochastic block model.

# Chapter 4

# Principal Component Analysis on graphs

Principal component analysis (PCA) is a multivariate statistical technique. Its goal is to provide a simpler description of a large data set. This is generally done by defining uncorrelated variables, from a large set of correlated ones, in a way that preserves the variation as much as possible. This uncorrelated new variables are called principal components. More details about PCA and their applications can be found for example, in [85].

**Roadmap** Section 4.1 presents our general framework and a strategy of PCA based on a variational formulation: a principal component of order $d$ is defined as a minimizer of a function $U_\nu$ over the set of geodesics that have a relative degree smaller than $d+1$. Section 4.2 focuses on principal components of order one (also referred to as first principal components) and provides the existence of such an element. A first difficulty we encounter is that the space of geodesics of relative degree smaller than 2 is not connected. Thus, we show that the optimization can be done on a more convenient (connected) space $\mathcal{G}_1$. Section 4.3 is dedicated to the study of $\mathcal{G}_1$ from a topological point of view, namely it shows that $\mathcal{G}_1$ can be partitioned in cells that are homeomophic to subsets of $\mathbb{R}^2$ and that it can be seen as a CW-complex. These properties are then used in Section 4.4 to define differential operators and introduce an infinitesimal generator $\mathcal{L}$. We conjecture the existence and uniqueness (in law) of an associated Markov process and in a second time, in Section 4.5, its convergence in probability towards the wanted set. We explain why we believe this conjecture is true and present some perspectives in how to prove it. Finally, in Section 4.6 we propose an algorithm for estimating first principal components of a quantum graph and provide some numerical results of a simplified version of it.

## 4.1 $d$-Principal Components on metric graphs

Let us first establish the framework and some of the notations that will be used throughout this chapter.

We consider $\Gamma$ a finite, undirected, simple, connected and weighted metric graph. We denote $V$ its vertex set and $E$ its set of edges. Each edge is homeomorphic to a real segment. Thus we fix a parametrization $(x_e)_{e\in E}$, such that for each edge $e$:

$$x_e : e \to [0, L_e],$$

where $L_e$ represents its length. We denote $d : \Gamma \times \Gamma \to \mathbb{R}_+$ the associated distance.

Given two points $y, z$ on an edge $e$, we denote $[y, z]$ or $[z, y]$, the segment of $\Gamma$, formed by the two: $[z, y] = [y, z] = x_e^{-1}([x_e(y), x_e(z)])$ if $x_e(y) \leq x_e(z)$ and $[z, y] = [y, z] = x_e^{-1}([x_e(z), x_e(y)])$ otherwise. The length of $[y, z]$ is naturally defined as $|x_e(z) - x_e(y)|$.

For $x, y \in \Gamma$, denote $\mathcal{C}_{x,y}$ the set of paths connecting $x$ to $y$:

$$\mathcal{C}_{x,y} = \{p \subset \Gamma, p = \bigcup_{i=0}^{n-1} [p_i, p_{i+1}]\},$$

where $n \in \mathbb{N}$ and:

- The extremities correspond to $x$ and $y$: $p_0 = x, p_n = y$, and the interior segments are edges: $[p_i, p_{i+1}] \in E$ for $i \in \llbracket 1, n-2 \rrbracket$
- The path does not cross the same vertex twice in the sense that: $p_i \neq p_j$ for all $i, j \in \llbracket 1, n-1 \rrbracket, i \neq j$ .
- The end points, $x$ and $y$ are not necessarily vertices: there exist $e_0$ and $e_n \in E$, such that $[p_0, p_1] \subset e_0$ and $[p_{n-1}, p_n] \subset e_n$ .

Observe that $x$ might equal $y$ and $\mathcal{C}_{x,x}$ is formed of the set of cycles containing $x$ and the singleton itself.

We define the length of a path $p$ as the sum of the lengths of the segments composing it and denote it $|p|$. If $A$ is a countable set, we also maintain the classical notation, $|A|$, for the cardinal of $A$, since generally it can not create confusion.

Let $\widetilde{\mathcal{G}}$ be the space of closed geodesics of $\Gamma$. A subset $\gamma \subset \Gamma$ is called a geodesic if for all $x, y \in \gamma$ there exists a shortest path from $x$ to $y$ included in $\gamma$ :

$$\widetilde{\mathcal{G}} = \{\gamma \subset \Gamma \text{ s.t. } \bar{\gamma} = \gamma,\ \forall x, y \in \gamma, x \neq y,\ \exists p \in \mathcal{C}_{x,y}, p \subset \gamma \text{ with } |p| = d(x, y)\}.$$

Here, the closure of a set is considered with respect to the natural topology of the metric space $(\Gamma, d)$.

All singletons of $\Gamma$ are included in $\widetilde{\mathcal{G}}$. If $\gamma_{x,y}$ is a path connecting two points $x, y$ of $\Gamma$, $x \neq y$ such that $\gamma_{x,y} \in \widetilde{\mathcal{G}}$ we call it *a geodesic path.*

For $x$ in $\Gamma$ and $\epsilon > 0$ denote $B_\epsilon(x) = \{y \in \Gamma \text{ s.t. } d(x, y) < \epsilon\}$, the open ball of radius $\epsilon$ centered at $x$.

**Definition 7.** *If $A$ is a connected subset of $\Gamma$ and $x$ a fixed point of $\Gamma$, we call the degree of $x$ relative to $A$, the number of connected components in which the set $A$ is split, only into an arbitrarily small neighborhood of $x$, when we remove $x$ from it:*

$$\deg^A(x) := \inf_{\epsilon > 0} N_c((A \setminus \{x\}) \cap B_\epsilon(x))$$

*where $N_c$ represents the number of connected components.*

The degree of $x$ relative to $A$ represents in how many directions we can go from $x$ inside the subset $A$. For $d \geq 0$, denote:

$$\widetilde{\mathcal{G}}_d := \{\gamma \in \widetilde{\mathcal{G}} \text{ s.t. } \forall x \in \gamma, \deg^\gamma(x) \leq d + 1\}.$$

This represents the set of geodesics $\gamma$ that contain only points with a relative degree (to $\gamma$) smaller or equal to $d + 1$. For example $\widetilde{\mathcal{G}}_0 = \{\{x\}; x \in \Gamma\}$ is the set of all singletons of the quantum graph.

An important case for us is $\widetilde{\mathcal{G}}_1$, the set of geodesics $\gamma$ such that $\deg^\gamma(x) \leq 2$, for all $x$ in $\gamma$. An element $\gamma$ of $\widetilde{\mathcal{G}}_1$ can be described in the following manner: either there exist

$x, y \in \Gamma$, $x \neq y$ such that $\gamma \in \mathcal{C}_{x,y}$, and thus $\gamma$ is a geodesic path, $\gamma$ is a singleton or $\gamma$ is a geodesic cycle (one can imagine a path that has the same starting and ending point $(x = y)$ and at the same time is a geodesic).

Given a probability measure $\nu$ over the vertex set $V$, for all $C \subset \Gamma$ we denote:

$$U_\nu(C) = \mathbb{E}[d^2(X, C)] = \sum_{i=1}^{n} d^2(x_i, C)\nu(x_i), \tag{4.1}$$

where $d(x, C)$ is defined in a natural way as:

$$d(x, C) = \inf_{y \in C} d(x, y).$$

**Definition 8.** *For $d \geq 0$ we call principal component of order $d$, an element $g_d^* \in \widetilde{\mathcal{G}}_d$ such that:*

$$g_d^* \in \arg\min_{g \in \widetilde{\mathcal{G}}_d} U_\nu(g).$$

*We denote $\mathcal{M}_\nu^d$ the set of minimizers of $U_\nu$ over $\widetilde{\mathcal{G}}_d$.*

A first observation is that for $d = 0$, minimizing $U_\nu$ over $\widetilde{\mathcal{G}}_0$ one retrieves the set $\mathcal{M}_\nu$, representing the classical Fréchet mean.
In an Euclidean space, principal components can be described as a sequence of nested affine sub-spaces of increasing dimension that maximize the variance of the projections or minimize the sum of norms of projection residuals. For example, consider the case of $m$ points $x_1, \dots x_m$, in $\mathbb{R}^n$. For all $v \in \mathbb{R}^n$ denote:

$$S_v = \{\bar{x}_m + tv, t \in \mathbb{R}\}, \text{ where } \bar{x}_m = \frac{1}{m}\sum_{i=1}^{m} x_i \text{ is the Euclidean mean.}$$

A first principal component of the data can be described as $S_{v_1}$, where $v_1$ is such that:

$$v_1 \in \arg\min_{\|v\|=1} \sum_{i=1}^{m} d^2(x_i, S_v).$$

With this in mind, one might expect that a first principal component on the graph has an equivalent property, namely that it contains a Fréchet mean. This however is not generally true. We illustrate a counter example in Figure 4.1.

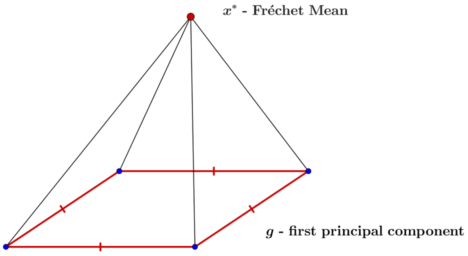


Figure 4.1: We consider the uniform probability on the vertex set and all edges of length 1. Then the top of the pyramid is the Fréchet mean and the edges forming the base are the first principal component.

Since this nesting property of principals modes in Euclidean spaces no longer holds for the notion of principal components of graphs introduced in Definition 8, we introduce

another notion, similar to the one introduced in [32] for the space of probability measures on the real line, of nested-principal components.
For $d = 1$ and $x \in \Gamma$, denote $\widetilde{\mathcal{G}}_1^x := \{g \in \widetilde{\mathcal{G}}_1, x \in g\}$, the subset of geodesics of $\widetilde{\mathcal{G}}_1$ containing $x$.

**Definition 9.** *We call $x$-principal component of order 1, a set $\mathfrak{g}_1^* \in \widetilde{\mathcal{G}}_1^x$ such that:*

$$\mathfrak{g}_1^* \in \underset{g \in \widetilde{\mathcal{G}}_1^x}{\arg\min}\, U_\nu(g).$$

*A quantity of interest is obviously a $x$-principal component, when $x$ is a Fréchet mean. We denote $\mathcal{N}_\nu^1$ the set of such principal components and also call them Fréchet principal components.*

This definition can now be extended in an iterative way. For $g_{d-1} \in \widetilde{\mathcal{G}}_{d-1}$, we call a $g_{d-1}$-principal component of order $d$, or nested principal component if $g_{d-1}$ is a principal component of order $d-1$, an element:

$$\mathfrak{g}_d^* \in \underset{g \in \widetilde{\mathcal{G}}_d, g_{d-1}^* \subset g}{\arg\min}\; U_\nu(g).$$

The two types of principal components defined above, have slightly different properties and could be of interest in different applications. In what follows we interest ourselves only in the study of the first principal component, namely of order one. We will mostly study the general principal components introduced in Definition 8. Only some of the results are explicitly presented for the Fréchet principal components but we believe that all included results can be extended.

## 4.2 First (Fréchet) Principal Component

Our goal is to construct an algorithm that converges towards an element of $\mathcal{M}_\nu^1$ (or $\mathcal{N}_\nu^1$). This implies optimizing a function over $\widetilde{\mathcal{G}}_1$ (respectively $\widetilde{\mathcal{G}}_1^x$). We start by introducing a metric on these spaces and present some of their basic topological properties.
Since $\widetilde{\mathcal{G}}_1$ is a subset of the power set of $\Gamma$, it is natural to consider the metric space $(\widetilde{\mathcal{G}}_1, h)$, where $h$ is the Hausdorff distance. We recall the definition of this distance.

**Definition 10.** *For two non-empty subsets $A, B \subset \Gamma$, define $d(A, B) = \sup_{x \in A} d(x, B)$. The Hausdorff distance $h$, between $A$ and $B$ is:*

$$h(A, B) = \max\{d(A, B), d(B, A)\}.$$

**Remark 5.** *Defined on the non-empty subsets of $\Gamma$, $h$ is only a pseudo-metric, but on the closed non-empty subsets it is a true distance (because $\Gamma$ is compact). Thus, one of the advantages of considering only the closed geodesics is that $h$ is indeed a distance on $\widetilde{\mathcal{G}}_1$.*

The space $\widetilde{\mathcal{G}}_1$ is not necessarily connected with respect to the topology induced by the Hausdorff distance, because the geodesic cycles (when they exist) are isolated points. We cannot approach a geodesic cycle by geodesic paths $g$, because a geodesic path cannot cover more than half of a cycle. A geodesic path, in particular, is a shortest path between its extremities. Thus the distance between the extremities cannot become less than the length of $g$.

Since our interest is to define a diffusion that is able to explore all the possible elements in a continuous manner, we introduce an auxiliary connected space, defined by:

$$\mathcal{G}_1 := \bigcup_{x,y\in\Gamma} \{g \in \mathcal{C}_{x,y}; \exists \gamma_{x,y} \text{ s.t. } g \cup \gamma_{x,y} \in \widetilde{\mathcal{G}}_1.\},$$

where for two points $x, y$, $\gamma_{x,y}$ is a geodesic path from $x$ to $y$ or the empty set. A first remark is that $\widetilde{\mathcal{G}}_1 \subset \mathcal{G}_1$, since all geodesic paths are included and all the cycles can be retrieved by taking $x = y$ and $\gamma_{x,y} = \{x\}$.

We show that optimizing $U_\nu$ over $\mathcal{G}_1$ instead of $\widetilde{\mathcal{G}}_1$ is not problematic in the sense that from any minimizing element of the first space, we can construct a minimizing element of the second. Moreover the minimum value on both spaces is the same.

**Lemma 2.** *From a given $g^* \in \arg\min_{g\in\mathcal{G}_1} U_\nu(g)$, we can easily construct a $\widetilde{g}^* \in \arg\min_{g\in\widetilde{\mathcal{G}}_1} U_\nu(g)$.*

*Proof.* To see that, first observe that if $g \in \mathcal{G}_1 \setminus \widetilde{\mathcal{G}}_1$, then $g$ is a path going from one point $x$ to a point $y$ and there exists a geodesic path $\gamma_{x,y}$ such that $g \cup \gamma_{x,y} \in \widetilde{\mathcal{G}}_1$.

Secondly, observe that $U_\nu$ is a decreasing function, in the following sense: $\forall A \subset B \subset \Gamma$, we have $U_\nu(A) \geq U_\nu(B)$. Thus, $U_\nu(g \cup \gamma_{x,y}) \leq U_\nu(g)$. Considering this and the fact that $\widetilde{\mathcal{G}}_1 \subset \mathcal{G}_1$, we obtain that the minimum value of $U_\nu$ over the two different sets is the same and:

$$\arg\min_{g\in\widetilde{\mathcal{G}}_1} U_\nu(g) \subset \arg\min_{g\in\mathcal{G}_1} U_\nu(g).$$

Since for all $g \in \mathcal{G}_1 \setminus \widetilde{\mathcal{G}}_1$ we can easily construct $g' \in \widetilde{\mathcal{G}}_1$ such that $U_\nu(g') \leq U_\nu(g)$, from any $g^* \in \arg\min_{g\in\mathcal{G}_1} U_\nu(g)$ we can immediately construct a $\widetilde{g}^* \in \arg\min_{g\in\widetilde{\mathcal{G}}_1} U_\nu(g)$. □

The arguments presented in the proof of Lemma 2 work in the same way on the subsets $\mathcal{G}_1^x = \{g \in \mathcal{G}_1, x \in g\}$ and $\widetilde{\mathcal{G}}_1^x = \{g \in \widetilde{\mathcal{G}}_1, x \in g\}$. Thus in order to find an element of $\mathcal{N}_\nu^1$ it is sufficient to minimize $U_\nu$ over $\mathcal{G}_1^x$, where $x$ is a Fréchet mean.

### 4.2.1 Existence of a (Fréchet) first principal component

This sub-section is dedicated to the proof of the existence of a first principal component and first Fréchet principal component.

**Lemma 3.** *$U_\nu$ is continuous on the power set of $\Gamma$, with respect to the Hausdorff distance $h$.*

*Proof.* First observe that for all $x \in \Gamma$ and $A, B \subset \Gamma$, $d(x, A) \leq d(x, B) + d(A, B) \leq d(x, B) + h(A, B)$. This implies that, for all $x \in \Gamma$, $|d(x, A) - d(x, B)| \leq h(A, B)$.

$$\begin{aligned}
|U_\nu(A) - U_\nu(B)| &\leq \sum_{x\in V} |d^2(x, A) - d^2(x, B)|\nu(x) \\
&\leq \sum_{x\in V} (d(x, A) + d(x, B))|d(x, A) - d(x, B)|\nu(x) \\
&\leq \sum_{x\in V} 2\mathcal{D}_G h(A, B)\nu(x) \\
&\leq 2\mathcal{D}_G h(A, B),
\end{aligned}$$

where $\mathcal{D}_G$ is the diameter of the graph. We have thus obtained that $U_\nu$ is Lipschitz continuous. □

**Definition 11.** *Let $g$ be an element of $\mathcal{G}_1 \backslash \{\{v\}, v \in V\}$. We call support of $g$ the minimal set of edges containing $g$:*

$$s_g := \{e \in E \textit{ such that } \mathring{e} \cap g \neq \emptyset\}, \tag{4.2}$$

*where $\mathring{e}$ represents the interior of the edge $e$. If $g \in \mathcal{G}_1$ is a vertex, in the sense that $\exists v \in V$ such that $g = \{v\}$, we define its support as itself, $s_g = g$.*

We denote $\mathcal{S}_{\mathcal{G}_1}$ the set of all supports of $\mathcal{G}_1$:

$$\mathcal{S}_{\mathcal{G}_1} = \{s_g \mid g \in \mathcal{G}_1\}. \tag{4.3}$$

A support is thus a subset of $E$ or a singleton formed by a vertex. We denote $\mathfrak{s}$ the subset of $\Gamma$ corresponding to $s$. In the first case, $s \subset E$ , this subset is formed by the union all the edges of $s$, $\mathfrak{s} = \cup_{e \in s} e = \{x \in \Gamma \mid \exists e \in s \text{ s.t. } x \in e\}$ and in the second case it is the support itself $\mathfrak{s} = s \subset \Gamma$.

**Definition 12.** *For a support $s$ we call $e \in s$ an extremal edge if and only if there exists $x \in e$ such that $\deg^{\mathfrak{s}}(x) = 1$. If $g$ has $s$ as support, we refer to the extremal edges of $s$ also as extremal edges of $g$. Moreover, we call $x \in g$ an extremity of $g$ if $deg^g(x) \leq 1$.*

**Remark 6.** *Any element of $\mathcal{G}_1$ is determined by its extremities (when they exist) and its support. When extremities do not exist, then the element is a cycle and thus entirely determined by its support.*

**Lemma 4.** *The metric spaces $(\widetilde{\mathcal{G}}_1, h)$ and $(\mathcal{G}_1, h)$ are compact ($h$ is the Hausdorff distance).*

*Proof.* Since we are dealing with metric spaces, it is enough to prove that they are sequentially compact.

Let $(C_n)_{n\geq 0}$ be a sequence of elements in $\mathcal{G}_1$. First we prove that it has a sub-sequence that converges. Then we prove that the limit is in $\mathcal{G}_1$, and furthermore if $C_n \in \widetilde{\mathcal{G}}_1$ for all $n$ then the limit is in $\widetilde{\mathcal{G}}_1$.

Any element $C_n$ can be written as $(x_n, s_n, y_n)$, with $x_n, y_n \in \Gamma$ and $s_n \in \mathcal{S}_{\mathcal{G}_1}$, where $\mathcal{S}_{\mathcal{G}_1}$ is defined in (4.3) and $x_n, y_n$ are the extremities of $C_n$ introduced in Definition 12. If $C_n$ is a cycle and has no extremities, for any $x \in C$ we can consider $x_n = y_n = x$.

We fix the following product metric, $d_1 : \Gamma \times \Gamma \to \mathbb{R}_+$ defined by:

$$d_1((x_1, y_1), (x_2, y_2)) = d(x_1, x_2) + d(y_1, y_2),$$

where $d$ is the usual distance defined on $\Gamma$. Since $(\Gamma \times \Gamma, d_1)$ is a compact metric space, there exists $((x'_n, y'_n))_{n\geq 0}$ a convergent sub-sequence of $((x_n, y_n))_{n\geq 0}$. Denote $(x, y)$ its limit.

Seeing that $\Gamma$ has a finite number of edges, $\mathcal{G}_1$ has a finite number of possible supports. Thus there exists a strictly increasing $\psi : \mathbb{N} \to \mathbb{N}$, and an element $s \in \mathcal{S}_{\mathcal{G}_1}$ such that $(x'_{\psi(n)}, y'_{\psi(n)})$ converges to $(x, y)$, and $(x'_{\psi(n)}, s, y'_{\psi(n)}) =: C_{\psi(n)}$. To simplify the notations from now on we refer to this sub-sequence as $((x_n, s, y_n))_{n\geq 0}$ and we prove that it converges with respect to the Hausdorff distance.

Denote $C = (x, s, y)$. Observe that $d(C_n, C) \leq \max\{d(x_n, x), d(y_n, y)\}$ and the same holds for $d(C, C_n)$. Thus:

$$h(C, C_n) \leq \max\{d(x_n, x), d(y_n, y)\} \leq d_1((x_n, y_n), (x, y)),$$

and so $C_n \to C$ when $n$ goes to infinity.

Now we need only to show that the limit is in the right space. First, suppose $(C_n)_{n\geq 0} \subset \widetilde{\mathcal{G}}_1$. Since $\widetilde{\mathcal{G}}_1 \subset \mathcal{G}_1$, this is only a particular case of the property proved above and thus the limit $C$ still exists.

If $x = y$ then either $C$ is a cycle and there exists $N$ such that $C_n = C$ for all $n \geq N$ (here the equality is understood in terms of subsets of $\Gamma$), or $C$ is reduced to a point. In both cases, we have $C \in \widetilde{\mathcal{G}}_1$. Suppose $x \neq y$. If $C_n$ are geodesic paths from $x_n$ to $y_n$, then

$C$ is a geodesical path from $x$ to $y$. This is easy to see considering the fact that the length of a path is continuous in its extremities with respect to the distance $h$. This shows that $(\widetilde{\mathcal{G}}_1, h)$ is indeed compact.

To conclude about $\mathcal{G}_1$ we use the following property: for all $A_1, B_1, A, B \subset \Gamma$, we have:

$$h(A_1 \cup B_1, A \cup B) \leq h(A_1, A) + h(B_1, B). \tag{4.4}$$

This property can be easily verified using the definition of $h$.

$$\begin{aligned} d(A_1 \cup B_1, A \cup B) &= \sup_{x \in A_1 \cup B_1} d(x, A \cup B) \\ &\leq \sup_{x \in A_1} d(x, A \cup B) + \sup_{x \in B_1} d(x, A \cup B) \\ &\leq \sup_{x \in A_1} d(x, A) + \sup_{x \in B_1} d(x, B) \\ &\leq d(A_1, A) + d(B_1, B) \end{aligned}$$

By symmetry we also have $d(A \cup B, A_1 \cup B_1) \leq d(A, A_1) + d(B, B_1)$, and so we have (4.4).

Let $(C_n)_{n \geq 0} \subset \mathcal{G}_1$. If for all $N \in \mathbb{N}$ there exists $n \geq N$ such that $C_n \in \widetilde{\mathcal{G}}_1$ then we can form a sub-sequence $(C'_n)_{n \geq 0} \subset \widetilde{\mathcal{G}}_1$ that converges to $C$. Since $\widetilde{\mathcal{G}}_1$ is compact this implies $C \in \widetilde{\mathcal{G}}_1 \subset \mathcal{G}_1$.

Suppose there exists $N \in \mathbb{N}$, such that for all $n \geq N$, $C_n \in \mathcal{G}_1 \setminus \widetilde{\mathcal{G}}_1$, and re-label this sub-sequence $(C_n)_{n \geq 0}$. For all $n$ there exists $\gamma_n$, a geodesic path from $x_n$ to $y_n$, such that $C_n \cup \gamma_n \in \widetilde{\mathcal{G}}_1$. Since $(x_n, y_n) \to (x, y)$, there exists a shortest path from $x$ to $y$, $\gamma_{x,y}$ such that $\gamma_n$ converges to $\gamma_{x,y}$ with respect to the Hausdorff distance (this path can eventually be reduced to a point when $x = y$).

Using (4.4), we have that $\lim\limits_{n \to \infty} C_n \cup \gamma_n = C \cup \gamma_{x,y}$. Since $C_n \cup \gamma_n \in \widetilde{\mathcal{G}}_1, \forall n$, we have that $C \cup \gamma \in \widetilde{\mathcal{G}}_1$, and thus $C \in \mathcal{G}_1$, ending the proof. □

**Proposition 7.** *Given $\Gamma$, a continuous compact graph with a finite number of vertices, for all probability measure $\nu$, $\mathcal{M}^1_\nu$ and $\mathcal{N}^1_\nu$ are non empty.*

*Proof.* The fact that the set of minimizers of $U_\nu$ on $\widetilde{\mathcal{G}}_1$ (or $\mathcal{G}_1$) is non-empty is an immediate consequence of Lemma 3 and Lemma 4. As for $\mathcal{N}^1_\nu$, let $x \in \mathcal{M}_\nu$ be fixed. $\widetilde{\mathcal{G}}^x_1$ is by definition a closed subset of $\widetilde{\mathcal{G}}_1$, and thus compact. Therefore, by Lemma 3, $\mathcal{N}^1_\nu$ is non-empty. □

In what follows, we focus our attention on $\mathcal{G}_1$, but the results can also be extended to $\mathcal{G}^x_1$.

## 4.3 Partition of $\mathcal{G}_1$

In order to define a stochastic process on $\mathcal{G}_1$ we start by partitioning the space into open sets that are homeomorphic to open convex sets of $\mathbb{R}^2$. This is the analog of first defining a diffusion inside the edges for the quantum graph.

A first step is to observe that $\mathcal{G}_1$ can be naturally partitioned using the set of supports:

$$\mathcal{G}_1 = \bigsqcup_{s \in \mathcal{S}_{\mathcal{G}_1}} \{g \in \mathcal{G}_1, \text{ with } s_g = s\}.$$

For $s \in \mathcal{S}_{\mathcal{G}_1}$, denote $\langle s \rangle$ the set of elements of $\mathcal{G}_1$ that have $s$ as support:

$$\langle s \rangle = \{g \in \mathcal{G}_1, \text{ with } s_g = s\}. \tag{4.5}$$

We denote $\bar{s}$ the closure of $\langle s \rangle$ in $\mathcal{G}_1$, that can also be described as:

$$\bar{s} = \{g \in \mathcal{G}_1,\ g \subset \mathfrak{s}\ |\forall e \in s; e \cap g \neq \emptyset\}. \tag{4.6}$$

In order to better understand this partition we treat separately the main types of support and the sets they generate. Also we introduce a collection of homeomorphisms that is crucial for defining differential operators.

First, if $s$ is a vertex then $\langle s \rangle = \bar{s} = s$.

Secondly, consider that $s$ is a path that is not a cycle. This case is a little more complex, so we consider two sub-cases.

Path I If $s$ has one extremal edge, then $s$ has only one edge $e$. For all $g$ in $\bar{s}$, define $f_s(g) = (x_e(g_-), x_e(g_+))$, where $g_-$ and $g_+$ are the extremities of $g$. Since an element of $\mathcal{G}_1$ is determined by its support and extremities, $f_s : \bar{s} \to \mathrm{Im}(f_s)$ is obviously a bijection. One can easily check that:

$$\mathrm{Im}(f_s) = \{(x, y) \in \mathbb{R}^2 | x \in [0, L_e], y \in [x, L_e]\}.$$

Path II If $s$ has two distinct extremal edges, call them $e_-$ and $e_+$. Suppose that the parametrizations $x_{e_\pm}$ are such that for all $g \in \bar{s}$, $x_{e_\pm}^{-1}(0) \in g$, in other words, both parametrisations go from the interior of each geodesic toward the exterior. This is without loss of generality, and is used only to simplify notations.

There are situations when for a given support $s$, we can have points on an extremal edge such that no elements in $\bar{s}$ contain them. See for example the point $y$ in Figure 4.2. However, if $g \in \bar{s}$, then for all $x \in g \cap e_\pm$ there exists an element $g_x \in \bar{s}$ for which $x$ is an extremal point.

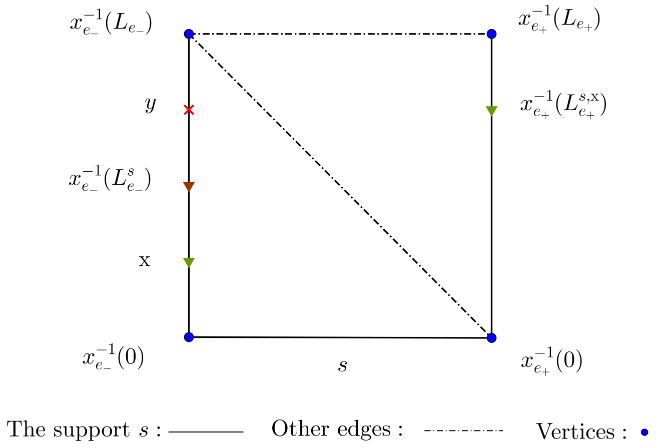


Figure 4.2: Consider the uniform probability on the vertex set and all edges of length 1. Here $L_{e_-}^s$ is the maximal starting point for an element of $\bar{s}$, defined in Equation (4.7), $y$ is an example of a point not contained by any element of $\bar{s}$ and $L_{e_+}^{s;\mathrm{x}}$ is the maximal ending point for a geodesic starting at x.

Even though there is no orientation, we refer to $e_-$ as the starting edge of the support and to $e_+$ as the ending edge, and for $g \in \bar{s}$, we denote $g_\pm$ the extremity that corresponds to $e_\pm$ and call it starting point (respectively the ending point). We define $L_{e_-}^s$ as the maximal possible starting point for an element of $\bar{s}$:

$$L_{e_-}^s = \max\{t \in (0, L_{e_-}] \mid \exists g \in \bar{s},\ \text{with } g_- = x_{e_-}^{-1}(t)\}. \tag{4.7}$$

Fixing a starting point can limit the choices of an ending point on the other extremal edge (see for example x in Figure 4.2), and so we denote $L_{e_+}^{s,x} \in [0, L_{e_+}]$ the maximal point that can be an ending point for an element of $\mathcal{G}_1$ starting at $x$ and having $s$ as support (or $s \setminus e_+$, since we look for elements of $\bar{s}$). Observe that $L_{e_+}^{s,x}$ is non-increasing with respect to $x$.

For all $g \in \bar{s}$ we define $f_s(g) = (x_{e_-}(g_-), x_{e_+}(g_+))$. Using the above notations we have:

$$\mathrm{Im}(f_s) = \{(x, y) \in \mathbb{R}^2 | x \in [0, L_{e_-}^s], y \in [0, L_{e_+}^{s,x}]\}, \tag{4.8}$$

and since $f$ is injective, we obtain once more a bijection from $\bar{s}$ to a convex compact of $\mathbb{R}^2$.

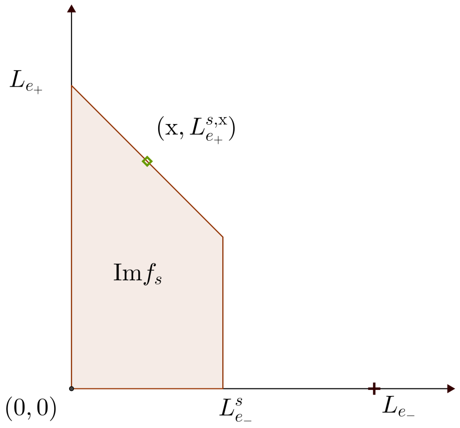


Figure 4.3: The image of $f_s$, defined in Equation (4.8), for the support $s$, represented in Figure 4.2

**Remark 7.** *For each $s$, of type* Path I *and* Path II*, the associated function $f_s$, is a local isometry, meaning that $\forall x \in \bar{s}$, $\exists\ \epsilon > 0$ such that $B_\epsilon^{\bar{s}}(x) := B_\epsilon(x) \cap \bar{s}$, the ball around $x$ of radius $\epsilon$ intersected with $\bar{s}$, is isometric to its image $f_s(B_\epsilon^{\bar{s}}(x))$ endowed with the supremum distance. To be more precise, $f_{s|_{B_\epsilon^{\bar{s}}(x)}}$ is an isometry between $(B_\epsilon^{\bar{s}}(x), h)$ and $(f_s(B_\epsilon^{\bar{s}}(x)), d_\infty)$, where $d_\infty(x, y) = max(|x_1 - y_1|, |x_2 - y_2|)$ is the distance associated to the infinity norm on $\mathbb{R}^2$.*
*At first glance, one might think that $f_s$ is an isometry on $\bar{s}$, and this is accurate for example, in the simple case of a graph with edges of same length. However, if we allow edges of arbitrary lengths it is not necessarily true. We illustrate a counter example in Figure 4.5.*

Finally, we consider the case when $s$ is a cycle. Suppose $s$ has $m$ edges, indexed from 1 to $m$. All elements of $\langle s \rangle$, except the cycle itself, either have both extremities on one edge or they have them on two distinct adjacent edges. In order to describe $\langle s \rangle$ it is convenient to consider separately these two cases.

Cycle I For each edge $e_i \in s$ denote $\langle s \rangle_i$ the set of elements of $\langle s \rangle$ whose extremities are on $e_i$ and $\bar{s}_i$ its closure. For all $1 \leq i \leq m$ denote:

$$D_i = \{(x, y) \in \mathbb{R}^2 | x \in [0, L_{e_i}]; y \in [x, L_{e_i}]\} \text{ and } \mathcal{R}_i = \{(x, x)\ | x \in [0, L_e]\} \tag{4.9}$$

We now consider $D_i/\mathcal{R}_i$ the quotient space obtained by identifying in $D_i$ all points of $\mathcal{R}_i$, endowed with its natural quotient topology. This space is homeomorphic to $\bar{s}_i$. To see this, we make the convention that the cycle $c$, (as an element of $\mathcal{G}_1$) can have any of its points $c_- = c_+ \in c$ as both starting and ending point and define the following homeomorphism:

$$f_{s_i} : \bar{s}_i \to D_i/\mathcal{R}_i \quad f_{s_i}(g) = (x_{e_i}(g_-), x_{e_i}(g_+)) \tag{4.10}$$

We give an illustration of such a function and its image in Figure 4.4. From a topological point of view, $D_i/\mathcal{R}_i$ can also be seen as an ellipse.

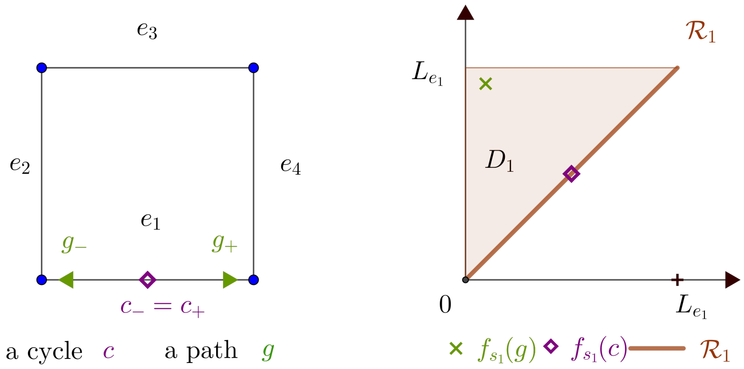


Figure 4.4: Right side: a geodesic cycle and two elements of $\bar{s}_1$. Left side: The sets $D_1$ and $\mathcal{R}_1$ defined in Equation (4.9). The image of $f_{s_1}$ (defined in Equation (4.10)) corresponds to the triangle if we identify all points on $\mathcal{R}_1$.

Cycle II For all $1 \leq i < j \leq m$ such that $e_i$ and $e_j$ have a common vertex, we denote $\langle s \rangle_{ij}$ the subset of elements of $\langle s \rangle$ that have one extremity on $e_i$ (denoted $g_-$) and the other on $e_j$ (denoted $g_+$), and $\bar{s}_{ij}$ its closure. Then, for each such pair $(i, j)$, there exists a bijection:

$$f_{s_{ij}} : \bar{s}_{ij} \to [0, L_{e_i}] \times [0, L_{e_j}], \text{ defined by } f_{s_{ij}}(g) = (x_{e_i}(g_-), x_{e_j}(g_+)) \tag{4.11}$$

with the convention that the complete cycle is considered to have as both starting and ending point the common vertex of the two edges.

**Remark 8.** *The functions $f_{i,j}$ defined in Cycle II, are local isometries (in the sense described in Remark 7) between $(\bar{s}_{ij}, h)$ and $(\mathrm{Im} f_{ij}, d_\infty)$. This property also holds for $f_i$ defined in Cycle I, but only on the interior of $\bar{s}_i$. Actually $d_\infty$ is not defined on $D_i/\mathcal{R}_i$, but we can introduce a corresponding distance. For all $\bar{x}, \bar{y}$, equivalence classes in the quotient space $D_i/\mathcal{R}_i$, we set:*

$$\bar{d}_i(\bar{x}, \bar{y}) = \min(d_\infty(x, \mathcal{R}_i) + d_\infty(y, \mathcal{R}_i), d_\infty(x, y)), \tag{4.12}$$

*where $x, y \in D_i$ are representatives of their respective classes and of course, $d_\infty(x, \mathcal{R}_i) = \inf\limits_{y \in \mathcal{R}_i} d_\infty(x, y)$. Now $f_i$ is a local isometry between $(\bar{s}_i, h)$ and $(\mathrm{Im} f_i, \bar{d}_i)$*

**Remark 9.** *The functions defined above (in Path I,Path II, Cycle I and Cycle II) are homeomophisms.*

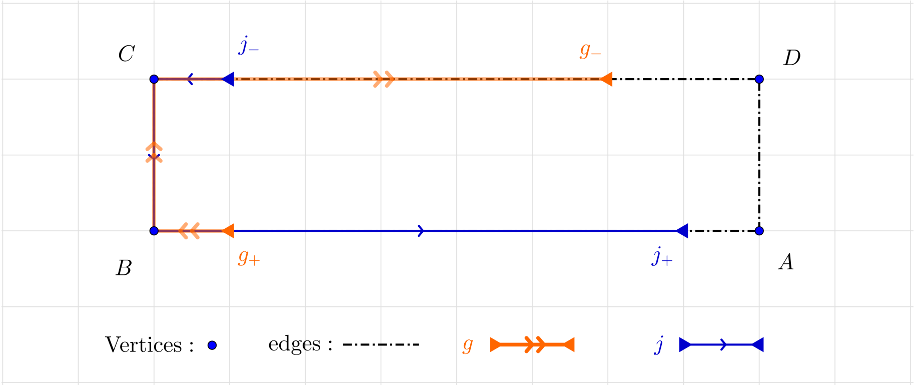


Figure 4.5: Take $s = \{[AB], [BC], [CD]\}$ as support and $j, g$ two paths in $\bar{s}$. Consider each square side of length 1. We have $d(g, j) = 5$, $d(j, g) = 5.5$, and thus $h(j, g) = 5.5$. However, $f_s(g) = (6, 1)$ and $f_s(j) = (1, 7)$ implying:

$$d_\infty(f_s(g), f_s(j)) \neq h(j, g).$$

This is due to the fact that $j_+ \in [AB]$ is closer to $g_- \in [CD]$ than to $g_+ \in [AB]$ and thus the (Hausdorff) distance between the two geodesics no longer corresponds to the maximum of the distances between the extremities belonging to the same edge.

### 4.3.1 CW-complex structure of $\mathcal{G}_1$

The collection of $(\bar{s})_{s \in \mathcal{S}_{\mathcal{G}_1}}$ covers all $\mathcal{G}_1$, and together with the associated bijections defined above, it splits it into components homeomophic to convex subsets of $\mathbb{R}^2$. Moreover the intersection of two such components is the empty set, a point or a subset homeomorphic to a bounded segment of $\mathbb{R}^2$. This suggests that $\mathcal{G}_1$ can be viewed as a collection of cells of dimension 2 that are glued together in a continuous way by cells of dimension 1 or 0, in other words there exists a partition that identifies $\mathcal{G}_1$ as a CW-complex. As we will see in the next section, this approach turns out to be very useful in defining differential operators on $\mathcal{G}_1$. We give some brief reminders related to this topological notion and the terminology attached to it.

We call a $n$ *dimensional open cell* a topological space that is homeomorphic to the open ball ($\mathbb{B}_n \subset \mathbb{R}^n$) of dimension $n$.

**Definition 13.** *A CW-complex is a Hausdorff space $X$, together with a partition into open cells that respects the following conditions:*

*C. For each cell of dimension $n$ there exists a continuous function $f : \bar{\mathbb{B}}_n \to X$ such that the restriction of $f$ to the open ball $\mathbb{B}_n$ is a homeomorphism to the cell itself and the image of the boundary of $\bar{\mathbb{B}}_n$ is included into a finite number of cells of dimension less than $n$.*

*W. A subset of $X$ is closed if and only if its intersection with the closure of each cell is closed.*

The term CW comes from 'closure finite' (that refers to the condition C.) and 'weak topology' (refers to W.).

We will now partition $\mathcal{G}_1$ into open cells corresponding to the condition C.. This partition comes naturally from the sets of elements with the same support.

Let $s \in \mathcal{S}_{\mathcal{G}_1}$ be a fixed support that is not a vertex, nor a cycle. Then Im$f_s$, the image of the corresponding bijection, is homeomorphic to $\bar{\mathbb{B}}_2$ and thus so is $\bar{s}$. We consider the interior of $\bar{s}$, as an open cell of dimension 2. Furthermore, $\text{Im} f_s$ is a convex polygon in $\mathbb{R}^2$ (of maximum five edges), and so its boundary is piecewise $\mathcal{C}^1$ and can naturally be separated into segments. We consider as cells of dimension 1 the image through $f_s^{-1}$ of the open segments corresponding to each edge of the polygon and as cells of dimension 0 the pre-image of all its corners. This implies that the cell, together with the associated mapping, respects the first condition of the definition. We repeat these procedure for all such $s \in \mathcal{S}_{\mathcal{G}_1}$. In Figure 4.6 we illustrate this decomposition for $\text{Im} f_s$, represented in Figure 4.3.

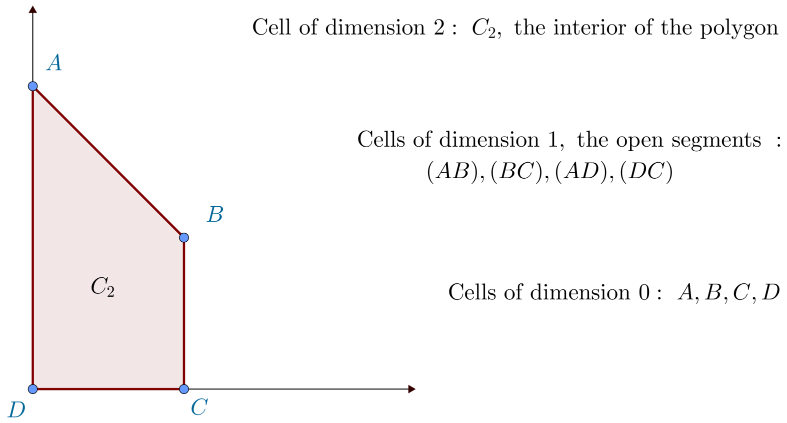

Cell of dimension 2 : $C_2$, the interior of the polygon
Cells of dimension 1, the open segments :
$(AB), (BC), (AD), (DC)$
Cells of dimension 0 : $A, B, C, D$
$A$
$B$
$C_2$
$D$
$C$

Figure 4.6: Consider $\bar{s}$, the subset generated by the support $s$ represented in Figure 4.2. This illustrates the partitioning of $\bar{s}$ into cells, through the image of the associate bijection $f_s$. For example, $f_s^{-1}(C_2)$ is a cell of dimension 2 in $\mathcal{G}_1$.

Let $s \in \mathcal{S}_{\mathcal{G}_1}$ be a cycle. First, we place ourselves in the framework considered in Cycle II. The subsets $\bar{s}_{ij}$, corresponding to elements of $\bar{s}$ that have $e_i$ and $e_j$ as extremal edges, are homeomorphic to rectangles. Thus we can use the same procedure of constructing cells through the associated functions $f_{s_{ij}}$: the pre-image of each rectangle's interior is a cell of dimension 2, the pre-image of open segments forming the sides are cells of dimension 1, and corners form cells of dimension 0.
Secondly, consider the closure of each subset of elements that have both extremities on the same edge, $\bar{s}_i$ defined in Cycle I. This set is homeomorphic to an ellipse (or to a triangle where the points of one side are identified, see Figure 4.4) and thus its interior forms a cell of dimension 2. Its borders $f_{s_i}^{-1}(\{0\} \times (0, L_{e_i}))$ and $f_{s_i}^{-1}((0, L_{e_i}) \times \{L_{e_i}\})$ form cells of dimension 1. $\{\mathfrak{s}\}$, where $\mathfrak{s}$ is the element of $\mathcal{G}_1$ corresponding to the complete cycle, and $f_{s_i}^{-1}(\{(0, L_{e_i})\})$ are the border of the cells of dimension 1 and form cells of dimension 0.

Denote $C^i$ the set of cells of dimension $i$, for $i \in \{0, 1, 2\}$. Since $\langle s \rangle$ are by definition disjoints, all considered open cells of dimension 2 are also disjoint. However, for the cells of dimension 1 it is important to notice that if they intersect they are actually equal. From a topological point of view, $\mathcal{G}_1$ can be seen as a collection of polygons and ellipses of $\mathbb{R}^2$ glued together, in the sense that points corresponding to same elements of $\mathcal{G}_1$ are identified. To give a better intuition of how the cells are glued together we illustrate a first example in Figure 4.7 and another one later on, in Figure 4.9.

Let $C$ be a cell of dimension 2. Then $C$ is either generated by a non cyclic support $s$, or by one subset of a cyclic support of the form $\langle s \rangle_i$, $\langle s \rangle_{i,j}$, meaning that $C = \langle \mathring{s} \rangle$ (or $C = \langle \mathring{s} \rangle_i$, respectively $C = \langle \mathring{s} \rangle_{ij}$). From now on the mapping associated to $\bar{s}$ (respectively $\bar{s}_i$ or $\bar{s}_{i,j}$) is also associated to $\bar{C}$ and thus sometimes referred to as $f_{\bar{C}}$, and its restriction $f_{\bar{s}|C}$ will sometimes be referred to simply as $f_C$.

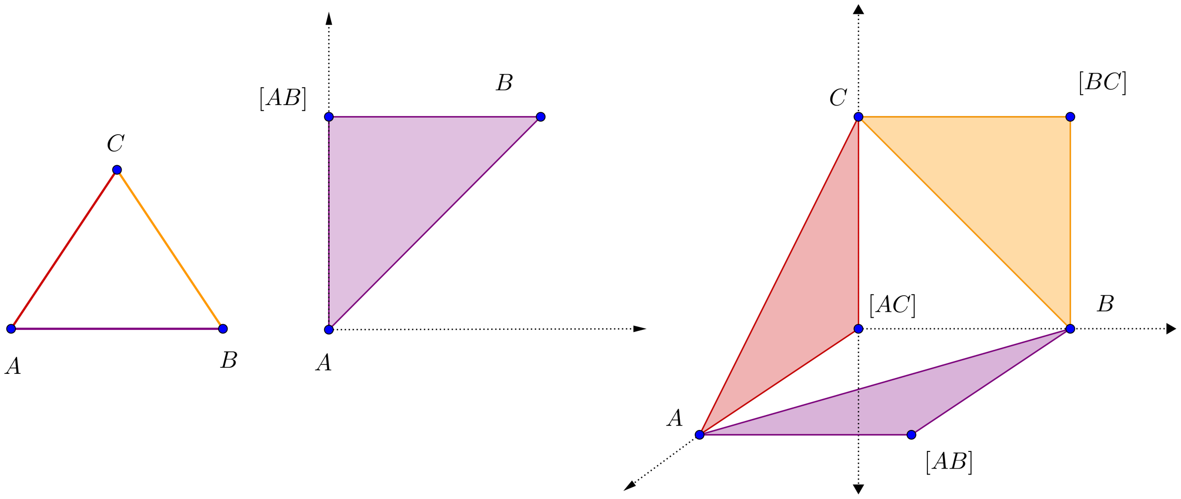


Figure 4.7: Left side: consider a triangle graph, $A, B, C$ its vertices and denote: $[AB], [BC], [AC]$ its edges. For each edge $e$, denote $\bar{s}_e$ the closure of the set of elements having $e$ as support and $f_e$ the associated function defined in PATH I. Each element of $\mathcal{G}_1$ and its image have the same label. Middle: representation of $f_{[AB]}(\bar{s}_{[AB]})$. Right: the corresponding images of $\bar{s}_{[AB]}$, $\bar{s}_{[BC]}$ and $\bar{s}_{[AC]}$, glued in an appropriate way.

## 4.4 Differential Operators and Markov Generator

On each cell $C \in C^2$, we can define differential operators in a natural way using the associated mapping $f_C$.

**Definition 14.** *Let $C \in C^2$ be a cell of dimension 2 and $f_C$ the associated mapping. For $0 \leq k \leq \infty$, we denote $\mathcal{C}^k(C)$ the set of functions $F : C \to \mathbb{R}$, such that $F \circ f_C^{-1} : \mathrm{Im} f_C \to \mathbb{R}$ is in $\mathcal{C}^k(\mathrm{Im} f_C)$. Furthermore, for all $F \in \mathcal{C}^1(C)$ we define*

$$\nabla F = (\partial_- F, \partial_+ F)^T,$$

*where:*

$$\partial_- F(g) = \partial_x F \circ f_C^{-1}|_{f(g)} \text{ and } \partial_+ F(g) = \partial_y F \circ f_C^{-1}|_{f(g)}.$$

Second order operators are defined now in an iterative way and we denote:

$$\Delta F = \partial_-^2 F + \partial_+^2 F. \tag{4.13}$$

**Remark 10.** *If the closure of $C$ does not contain a cycle, this definition can be extended to $\bar{C}$ in a straightforward way, using the extended mapping $f_{\bar{C}}$. Otherwise it can be extended only to $\bar{C} \setminus \{c\}$, where $c$ is the unique cycle contained in $\bar{C}$. Furthermore for all $C \in C^2$ and all $C_1 \in C^1$ such that $C_1 \subset \bar{C}$, we say that a function is of class $\mathcal{C}^k$ on $C \cup C_1$ if the restriction of $F \circ f_{\bar{C}}^{-1}$ to $C \cup C_1$ is of class $\mathcal{C}^k$ (respectively the restriction to $C \setminus \{c\}$).*

### 4.4.1 Diffusion process inside a cell

For any cell $C$ of dimension 2 and any function $b : C \to \mathbb{R}$ we can introduce a local second order Markov generator defined on $C$ by:

$$\forall F \in \mathcal{C}^2(C), \quad \mathcal{L}_b F(x) = \langle b(x), \nabla F(x) \rangle + \Delta F(x),$$

If $b$ is bounded and Lebesgue-measurable (with respect to the $\sigma$-algebra generated by the Lebesgue measure over the cell), this uniquely defines (in the weak sense) a diffusion process on $C$, up to the first time it hits the boundary of $C$.
We call a *Brownian motion on* $C$, a Markov process generated by:

$$\forall F \in \mathcal{C}^2(C) \quad \mathcal{L}F(x) = \frac{1}{2}\Delta F(x).$$

The definition of differential operators on $C$ entails that this process can be seen as the image through the inverse mapping $f_C^{-1}$ of a classical Brownian motion on $\mathrm{Im} f_C \subset \mathbb{R}^2$, started at a point $x_0 \in \mathrm{Im} f_C$. The fact that $f_C$ is a local isometry between $(C, h)$ and $(\mathrm{Im} f, d_\infty)$, implies that for $g \in C$, seen as $(g_-, s_g, g_+)$ for time $t$ small enough, a Brownian motion started at $g$ can be seen as $(B_-(s), s_g, B_+(s))_{0\leq s\leq t}$ where $(B_\pm(s))_{0\leq s\leq t}$ are two independent Brownian motions on $\Gamma$, started at $g_\pm$.

Taking $b(g)$ of the form $\beta_t \nabla U_{\nu_{|C}}(g)$, where $\beta_t$ is a continuous function that represents the temperature schedule and $U_{\nu_{|C}}$ is the restriction of $U_\nu$ to $C$, we obtain a family of Markov generators associated to an extension of the classical simulated annealing on $C$.

Once more, we place ourselves in a case where $\nabla U_\nu$ cannot be explicitly evaluated but we have access to $(Y_n)_{n\geq 0}$ a sequence of independent random variables distributed according to $\nu$. To overcome this issue we use the same homogenization technique as in Chapter 2. Replacing the probability measure $\nu$ by the Dirac measure on a vertex, for all $y \in V$, we define:

$$U_y : \mathcal{G}_1 \to \mathbb{R}_+, \quad U_y(x) = d^2(y, x) \quad \forall x \in \mathcal{G}_1$$

**Remark 11.** *For all $y \in V$, $U_y$ is continuous with respect to the Hausdorff distance on $\mathcal{G}_1$. Furthermore, for each $C \in C^2$, its restriction to $C$ is piecewise $\mathcal{C}^\infty$. We give an example of points on which $\nabla U_y$ is not well defined in Figure 4.8. On such points we set $\nabla U_y = 0$. This choice is somehow arbitrary has no significance in the sequel.*

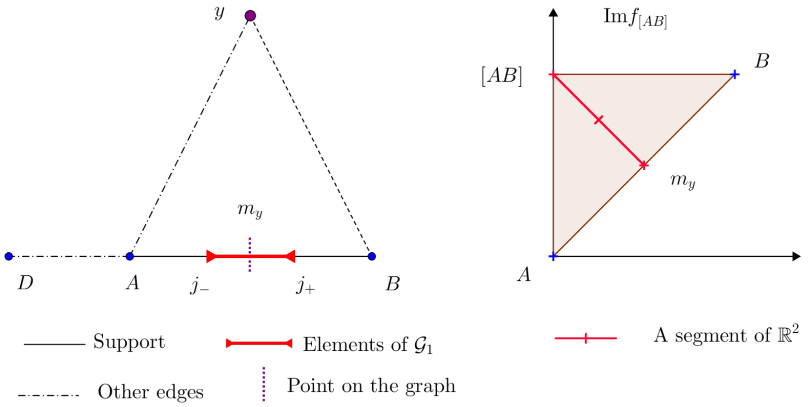


Figure 4.8: We use the same notations as in Figure 4.7. The length of each edge is 1. On the left side we have an example of an element $j$ on which both partial derivatives of $U_y$ are not well defined. $j$ has both extremities at equal distance from $y$, and so, moving one extremity towards the interior of $j$ leaves $U_y$ constant, moving it towards the exterior decrease it. On the left-hand side, we have the image of elements of $\mathcal{G}_1$ that have $[AB]$ as support. The red line represents the image of points on which $\nabla U_y$ is not well defined.

Following the same roadmap as before, let $\alpha_t : \mathbb{R}_+ \to \mathbb{R}_+$ be a continuous function that represents the rate at which we use the sequence $(Y_n)_{n\geq 0}$.

Given two positive continuous functions $\alpha_t, \beta_t \geq 0$, for all $F$ defined on $C \times V$ such that for all $y \in V$, we have $F(\cdot, y) \in \mathcal{C}^2(C)$ we define the inhomogenuous family of generators $(\mathcal{L}_t^C)_{t\geq 0}$ :

$$\mathcal{L}_t^C F(x,y) = -\beta_t \langle \nabla U_y(x), \nabla_x F(x,y)\rangle + \Delta_x F(x,y) + \alpha_t \int_V F(x,y) - F(x,y')\mathrm{d}\nu(y'), \tag{4.14}$$

where $\nabla_x$ and $\Delta_x$ represent differential operators with respect to $x$, introduced in Definition 14 and Equation (4.13), applied to $F(\cdot, y)$, for a fixed $y$. Once more, this family of generators uniquely defines (in the weak sense) a Markov Process $(X_t, Y_t)$ on $C \times V$, up to the first time $X_t$ hits the boundary of $C$. The dynamic of the first component, up to this time, is driven by a standard diffusion drifted by $\beta_t \nabla_x U_{Y_t}$, while $Y_t$ acts as a jump process over the set of vertices.

### 4.4.2 Boundary conditions

In order to define a global operator on $\mathcal{G}_1$, we first need to define the behavior of the underlying process $(X_t)_{t\geq 0}$ when it hits the boundary of a cell of dimension 2.

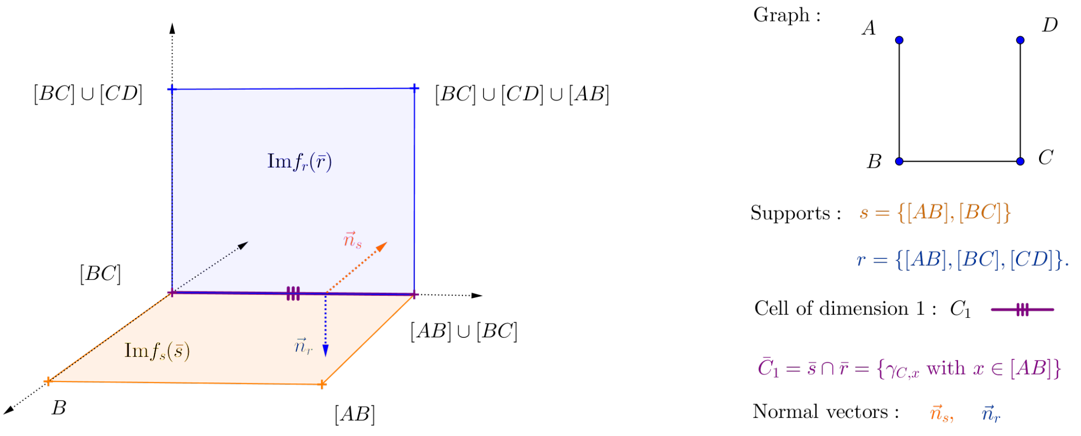
Let $C_1$ be a cell of dimension 1 and denote $S_{C_1}$ the set of cells $C \in C^2$ such that $C_1 \cap C \neq \emptyset$.

**Definition 15.** *For all $F : C \cup C_1 \to \mathbb{R}$, of class $\mathcal{C}^1$, for all $g$ in $C_1$, we define the normal derivative of $F$ with respect to $C$ at $g$ as:*

$$\frac{\partial}{\partial_C \vec{n}} F(g) = \frac{\partial}{\partial \vec{n}} \left( F \circ f_{\bar{C}}^{-1} \right) (f_{\bar{C}}(g)), \tag{4.15}$$

*where $f_{\bar{C}}$ is the mapping corresponding to the closure of $C$ and $\frac{\partial}{\partial \vec{n}}$ is the normal derivative on $\mathbb{R}^2$ with respect to the border of $\mathrm{Im} f_C$.*

Given the regularity of $f_{\bar{C}}(C^1)$ (as a curve in $\mathbb{R}^2$), the normal vector to $f_{\bar{C}}(C^1)$ is always well defined in $\mathbb{R}^2$.



$[BC] \cup [CD]$
$[BC] \cup [CD] \cup [AB]$
$\mathrm{Im} f_r(\bar{r})$
$\vec{n}_s$
$[BC]$
$[AB] \cup [BC]$
$\mathrm{Im} f_s(\bar{s})$
$\vec{n}_r$
$B$
$[AB]$
Graph :
$A$
$D$
$B$
$C$
Supports : $s = \{[AB], [BC]\}$
$r = \{[AB], [BC], [CD]\}$.
Cell of dimension 1 : $C_1$
$\bar{C}_1 = \bar{s} \cap \bar{r} = \{\gamma_{C,x}$ with $x \in [AB]\}$
Normal vectors : $\vec{n}_s$, $\vec{n}_r$

Figure 4.9: As before if $A, B$ are vertices $[AB]$ represents an edge, the label of an element of $\mathcal{G}_1$ is used also for its image and $\gamma_{x,y}$ is a geodesic path form $x$ to $y$. We also use the notations introduced in Sub-section 4.3.

Consider a probability measure $P_{C_1}$ on $C^2$ such that $P_{C_1}(C) > 0$ if and only if $C \in S_{C_1}$. Let $F : \mathcal{G}_1 \to \mathbb{R}$ be such that for all $C_1 \in C^1$ and all $C \in S_{C_1}$, the restriction of $F$ to $C \cup C_1$ is of class $\mathcal{C}^1$. We define the following Neumann boundary condition:

$$\sum_{C \in S_{C_1}} P_{C_1}(C) \frac{\partial}{\partial_C \vec{n}} F(g) = 0 \quad \forall g \in C_1. \tag{4.16}$$

Imposing this condition on the functions included in the generator's domain, determines the probability that the diffusion process, when in $C_1$ enters a given neighboring cell $C$. One can make here an analogy with the Newman conditions on the quantum graph, that determine the behavior of a diffusion when hitting a vertex. For example, a natural choice for this probability, in the case of a continuous graph, corresponds to the uniform probability among the adjacent edges.

To provide a better intuition on the effects of this condition, consider for example, the case represented in Figure 4.9. Using the same notations, an element of $C_1$ is a geodesic having one extremity on $[AB]$ and the other at $C$. There are only two cells of dimension 2 whose closure intersect $C_1$: the cells generated by $\bar{s}$ and $\bar{r}$ (*i.e.* the interior of $\langle s \rangle$ and $\langle r \rangle$, defined in Equation (4.5)), call them $C_s$ and $C_r$. Setting $P_{C_1}(C_s) = p$ and $P_{C_1}(C_r) = 1-p$ entails that when one extremity of the process reaches the vertex $C$ while the other is on the edge $[AB]$, the first extremity will enter $(BC)$ (the interior of the edge corresponding to $s$) with probability $p$ and will go towards $D$ with probability $1-p$. The choice is done, of course, independently of the direction the process came from. This is the analog of what happens when a diffusion, defined on the quantum graph, hits a vertex. However, one should not forget that in the case of a process evolving on $\mathcal{G}_1$ we might have some naturally forbidden directions. This can happen when one extremity hits a vertex, but also on the interior of an edge. See for example Figure 4.10.

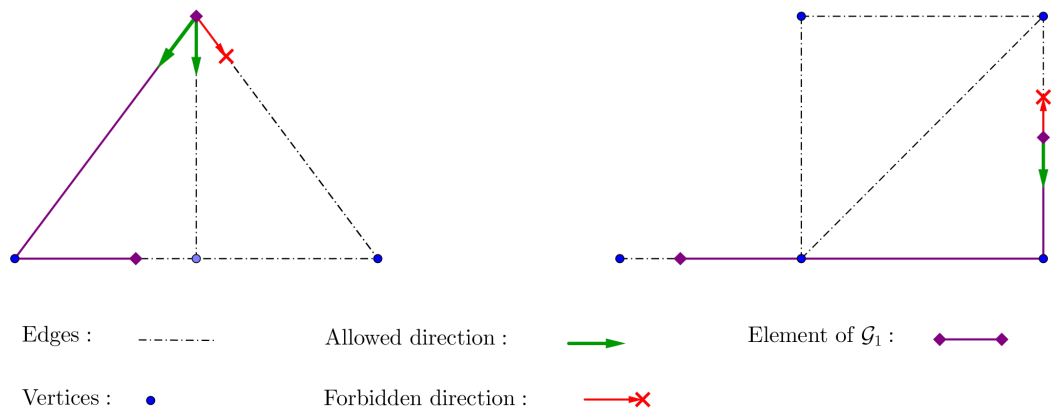


Figure 4.10: Consider that all lengths are 1. On the left hand side we have an example of a forbidden direction for one extremity when it hits a vertex and on the right hand side a forbidden direction for an extremity on the interior of an edge. In both cases the element of $\mathcal{G}_1$ cannot expand itself in the forbidden direction because such a move would result in a path that is not geodesic and cannot be completed to form a geodesic cycle.

Let $c \in \mathcal{G}_1$ be a cycle and let $x \in c$. For $\epsilon > 0$ denote $c_\epsilon^x$ the element of $\mathcal{G}_1$ obtained by removing from the cycle the ball of radius $\epsilon$ and center $x$:

$$c_\epsilon^x = c \setminus B_\epsilon(x).$$

Let $s$ be the support of $c$ and $F$ a function, $F : \bar{s} \to \mathbb{R}$. We say that $F$ is $x$-differentiable at $c$ if the following limit exists and it is finite:

$$\lim_{\epsilon \to 0} \frac{F(c^x_\epsilon) - F(c)}{\epsilon}.$$

If the limit exists and is finite we denote it $D_x F(c)$. Let $\mu_c$ be a probability measure on $c$ and $p_c$ a probability measure on the set of edges such that $p_c(e) > 0$ if and only if $e$ is in the support of $c$. Let $F$ be a function defined on $\mathcal{G}_1$. We introduce the following boundary condition with respect to the cycle $c$:

$$\sum_{e \in E} p_c(e) \int_{x \in e} D_x F(c) \mu_c(\mathrm{d}x) = 0. \tag{4.17}$$

Denote $\mathcal{D}_x$ the set of functions $F$ defined on $\mathcal{G}_1$ such that:

Reg $F$ is continuous on $\mathcal{G}_1$.
For all cells $C$ in $C^2$, the restriction of $F$ to $C$ is $\mathcal{C}^2$.
For all $C_1$ such that $C_1 \cap \bar{C} \neq \emptyset$ the restriction of $F$ to $C \cup C_1$ is in $\mathcal{C}^1(C \cup C_1)$.
For all cycles $c$, and all $x \in c$, $F$ is $x$-differentiable.

Bc1 The Neumann boundary condition (4.16) holds for the collection of probabilities given by:

$$p_{C_1}(C) = \begin{cases} 1/|S_{C_1}| & \text{if } C \in S_{C_1} \\ 0 & \text{otherwise.} \end{cases}$$

Bc0 The boundary condition given by (4.17) holds for the following collection of probabilities: for each cycle $c$, we choose $\mu_c$ as the uniform continuous distribution on $c$ and $p_c$ as a discrete distribution on $s_c$, the support of $c$ defined by:

$$p_c(e) = \begin{cases} L_e/|c| & \text{if } e \in s_c \\ 0 & \text{otherwise.} \end{cases}$$

where $L_e$ is the length of each edge $e$ and $|c| = \sum_{e \in s_c} L_e$ is the length of the cycle.

### 4.4.3 Markov processes on $\mathcal{G}_1$

We denote $\mathcal{D}$ the set of functions $F : \mathcal{G}_1 \times V$ such that for all $y \in V$, $F(\cdot, y) \in \mathcal{D}_x$. We introduce the following global operator $\forall F \in \mathcal{D}, \forall g \in \mathcal{G}_1$:

$$\mathcal{L}_t[F](g, y) = \mathcal{L}_t^C[F_{|C}](g, y) \quad \text{if } g \in C \text{ for some } C \in C^2. \tag{4.18}$$

**Conjecture 1.** *The martingale problem associated to the family of generators* $(\mathcal{L}_t)_{t \geq 0} : \mathcal{D} \to \mathcal{C}(\mathcal{G}_1)$ *is well-posed.*

This means that for all initial law $m_0$, there exists a weakly unique process $(X_t, Y_t)_{t \geq 0}$, distributed according to $m_0$ at time 0, such that, for all $F$ in $\mathcal{D}$, the real valued stochastic process defined by:

$$\forall t \geq 0 \quad M_t^F = F(X_t, Y_t) - F(X_0, Y_0) + \int_0^t \mathcal{L}_u(F)(X_u, Y_u) \mathrm{d}u \tag{4.19}$$

is a martingale.

A possible way of proving Conjecture 1, would be to adapt to $\mathcal{G}_1$ the proof of Freidlin and Wentzell provided for the case of a quantum graph [64]. The part concerning the existence of the process inside the cells, is based on the Hille-Yoshida theorem, and its adaptation to $\mathcal{G}_1$ should pose no problem, since it can be reduced to the existence on a diffusion inside a subset of $\mathbb{R}^2$. The second part, concerning the transition of the process from one edge to the other, is dealt with using properties of partial differential equations with boundary conditions and thus adapting this part would require a more careful inspection.

Indeed, the main difficulty in rigorously describing the behavior of a diffusion on a CW-complex (whether we are talking about a quantum graph or the space $\mathcal{G}_1$) comes from the fact that a standard real Brownian motion hits 0 infinitely many times in any time interval of the form $[0, t)$, $t > 0$. In our framework this translates into the fact that once the diffusion hits a cell of dimension 1 it will return to it infinitely many times in any small interval of time. A way to bypass this is to use excursions for the construction of the Brownian motion. Revuz and Yor give details about such a construction in []. An excursion can be seen as the path of a Brownian motion between two consecutive times it hits 0. Roughly, the idea of the construction is to generate an infinite (countable) number of excursions using a Poisson point process and then to join them together according to properties of local time. Excursions of length larger than $\varepsilon > 0$ are concatenated and then the limit is taken when $\varepsilon$ goes to zero.

Let us make a few comments on the generators domain. The condition Reg, ensures that the generator is well defined for all functions in its domain. Heuristically the condition Bc1 ensures that, when the process hits a cell of dimension one, it instantly reflects on any of the neighboring cells of dimension 2 with equal probability. This is the equivalent of the reflection of the diffusion defined in Section 2.3.1, on a quantum graph, when it hits a vertex. As for the boundary condition regarding cycles, Bc0, the intuitive interpretation is the following: when the process $X_t$ is equal to a cycle $c$, then it instantly chooses a separation point, uniformly among all $x \in c$.

### Brownian Motion

Before passing on to the next section, we introduce a particular diffusion process on $\mathcal{G}_1$, namely the Brownian motion. This is useful for describing the dynamic of the Markov process $(X_t, Y_t)$ and for underlining the similarities with the classical simulated annealing.

**Definition 16.** *We call a Brownian motion on $\mathcal{G}_1$ the Markov process associated to infinitesimal generator $L$ defined on $\mathcal{D}_x$ by:*

$$\forall F \in \mathcal{D}_x, \quad L[F](g) = \frac{1}{2}\Delta F(g) \quad \forall g \in C \text{ with } C \in C^2. \tag{4.20}$$

The existence and uniqueness in law of such a process is also given by Conjecture 1, since it can be seen as the first component of the Markov process generated by $(\mathcal{L}_t)_{t\geq 0}$, when the two auxiliary functions $\alpha_t$ and $\beta_t$ are null.
Inside a cell $C \in C^2$ a Brownian motion acts as the image of a standard Brownian motion on a polygon of $\mathbb{R}^2$ through the inverse of the associated mapping $f_C$ and on a border of a cell it adopts the behavior prescribed by Bc1 and Bc0.

## 4.5 Convergence Statement

We introduce the notation $m_t$ to refer to the distribution of the couple $(X_t, Y_t)$ at time $t$, and we define $n_t$ as the marginal distribution of $X_t$. In the following, we will also need to deal with the conditional distribution of $Y_t$ given the position $X_t$ in $\Gamma_G$. We will refer to this probability distribution as $m_t(y|x)$. To sum up, we have:

$$\mathscr{L}(X_t, Y_t) = m_t \quad \text{with} \quad n_t(x)dx = \int_V m_t(x,y)\mathrm{d}y \quad \text{and} \quad m_t(y\,|x) := \mathbb{P}[Y = y\,|X_t = x]. \tag{4.21}$$

We want to introduce the Gibbs measures with potential $U_\nu$. In order to do that, we first need to introduce integrals on $\mathcal{G}_1$. This is done in a natural way, through the set of homeomorphisms introduced in Section 4.3.

**Definition 17.** *Let $C$ be a cell of dimension 2, $f_C$ its associated homeomorphism and $A \subset C$. We say that a function $F : C \to \mathbb{R}$ is integrable on $A$ if and only if $F \circ f_C^{-1}$ is so on* $\mathrm{Im} f_C(A)$*. If $F$ is integrable on $A$ we set:*

$$\int_A F(g)\mathrm{d}g := \int_{\mathrm{Im} f_C(A)} F \circ f_C^{-1}(x)\ \mathrm{d}x.$$

To extend this definition to a subset $A$ of $\mathcal{G}_1$, (not necessarily included in a cell of dimension 2) we decompose $A$ in subsets of cells of dimension 2:

$$\int_A F(g)\mathrm{d}g := \begin{cases} 0, & \text{if } \forall\ C \in C^2\ A \cap C = \emptyset, \\ \displaystyle\sum_{C \in C^2} \int_{A\cap C} F(g)\mathrm{d}g, & \text{otherwise.} \end{cases} \tag{4.22}$$

We now have everything we need to introduce the Gibbs measure with potential $U_\nu$ and inverse temperature $\beta$, whose distribution is defined for all $g \in \mathcal{G}_1$ as:

$$\mu_\beta(g) = \frac{\exp(-\beta U_\nu(g))}{Z_\beta} \quad \text{where} \quad Z_\beta = \int_{\mathcal{G}_1} \exp(-\beta U_\nu(g))\mathrm{d}g.$$

It is well known that, as $\beta$ goes to infinity, this measure concentrates on the set of global minima of $U_\nu$. This is why the classical simulated annealing procedure is based on the construction of an inhomogeneous Markov process whose law will get 'close' to $\mu_{\beta_t}$, the Gibbs field with inverse temperature schedule $\beta_t \to +\infty$, as $t$ goes to infinity (see for example [73] or [77]). Since our algorithm is a homogenizes simulated annealing, seems natural to study the convergence of the process by measuring the gap between its distribution and the one of the associated Gibbs measure. To that end, we introduce the following relative entropy:

$$J_t := \int_{\mathcal{G}_1} \log\left[\frac{n_t(g)}{\mu_{\beta_t}(g)}\right] dn_t(g). \tag{4.23}$$

The Csiszár-Kullback-Pinsker inequality ensures that the relative entropy is a good way of measuring the distance between two probabilities measures (see [46, 91, 112]), since it provides an upper bound on the total variation distance $d_{TV}$:

$$d_{TV}(n_t, \mu_{\beta_t}) \leq \sqrt{2J_t}.$$

This in particular implies that if $J_t$ goes to zero, then $X_t$ converges in probability to the set of minimums of $U_\nu$. To be more exact, one can prove (see Corollary 1 in Chapter 2) that for any neighborhood $\mathcal{N}$ of $\mathcal{M}_\nu$:

$$\lim_{t \longrightarrow +\infty} \mathbb{P}[X_t \in \mathcal{N}] = 1.$$

Let $\beta, \alpha : \mathbb{R}_+ \to \mathbb{R}_+^*$ be two functions such that:

$$\alpha_t = a(t+1) \quad \text{and} \quad \beta_t = b\log(t+1) \quad \text{with } a, b > 0 \tag{4.24}$$

**Conjecture 2.** *There exists $b > 0$ such that if $\alpha_t$ and $\beta_t$ are defined by (4.24), then the relative entropy $J_t$ goes to zero.*

The space $\mathcal{G}_1$ can also be seen as a generalization of the initial space $\Gamma_G$, since the metric graph $\Gamma_G$ is a CW-complex of dimension 1 and $\mathcal{G}_1$ is a CW-complex of dimension 2. It is thus reasonable to expect the methods applied in Chapter 2, for the study of a homogenized simulated annealing on a quantum graph, to be adaptable to this new setting.

Suppose that the logarithmic Sobolev inequality holds for $\mu_{\beta_t}$ and that the distribution $m_t$ of the process $(X_t, Y_t)$ is smooth enough. Then, we can follow the same roadmap as the one described in Section 2.10.1 in order to obtain a differential inequality for $J_t$ and prove its convergence towards 0. Of course, the computations need to be adapted, but generally this should not be problematic. Terms that might require a more careful treatment are the ones involving an integration by parts and thus, implicitly the border condition Bc0.

As for the logarithmic Sobolev inequality, Section 2.8 provides an extension to quantum graphs, of the proof given by [77] for smooth compact manifolds. Except for the ordinary Sobolev inequality for $\mu_0$ (see Lemma 1 ), none of the intermediate results use in a decisive way the dimension of $\Gamma_G$ and thus can be adapted in a rather straightforward way to $\mathcal{G}_1$. In order to obtain the ordinary Sobolev inequality in Lemma 1, we use the fact that for a real segment $I$, the Sobolev space $W^{1,p}(I)$ is continuously embedded in $L^\infty(I)$ (see, *e.g.*, [39]). This property is not necessarily true for a compact subset of $\mathbb{R}^2$ and thus we need to find another way of proving that there exist $p, C_p > 0$ such that for all $\mu_0$-measurable functions $F$ we have:

$$\|F - \langle F \rangle_0\|_{p,\mu_0}^2 \le C_p \mathcal{E}_0(F, F),$$

where $\mu_0$ is the normalized Lebesgue measure and $\mathcal{E}_0$ is the Dirichlet form associated with the Laplacian, $\mathcal{E}_0(F, F) = \displaystyle\int_{\mathcal{G}_1} |\nabla F(g)| \mathrm{d}g$. In [77] the authors suppose that there exists $p \in (2, +\infty)$ such that this preliminary Sobolev inequality holds, but they only provide the existence of such a constant for compact connected Riemanian manifolds of dimension strictly larger than 2.

## 4.6 Algorithm

We will now give a more detailed description of the evolution of the process $(X_t, Y_t)_{t\ge 0}$ and propose an algorithm for estimating a first principal component of $\mathcal{G}_1$.

### 4.6.1 Homogenized simulated annealing over $\mathcal{G}_1$

Given a continuous function $\alpha_t$, as in Chapter 2, we introduce $(N_t^\alpha)_{t\ge 0}$, an inhomogeneous Poisson process over $\mathbb{R}_+$ with intensity $\alpha$. We remind the reader that $N^\alpha$ can be represented through a homogeneous Poisson process $H$ of intensity 1 using the relationship:

$$\forall t \ge 0 \qquad N_t^\alpha = H_{h(t)}, \text{ where } h(t) = \int_0^t \alpha_s \mathrm{d}s.$$

Given $(\mathbf{Y}_n)_{n\in\mathbb{N}}$ a sequence of independent random variables distributed according to $\nu$, the dynamic of the process $(X_t, Y_t)_{t\geq 0}$ can be roughly described as follows: $Y_t = \mathbf{Y}_{N_t^\alpha}$ and between two jumping times $T_k$ and $T_{k+1}$ of the Poisson process $N_t$, the $X_t$ component acts as a Brownian motion on $\mathcal{G}_1$ drifted by $\nabla U_{\mathbf{Y}_k}$.

At each time $t$, $X_t$ is determined by its extremities and its support and thus can be written as: $(x_-(t), s_X(t), x_+(t))$. The Markovian dynamic acts in a direct way only on the extremities of $X_t$ and the evolution of the support is determined by the evolution of the extremities. Moreover, while a process moves inside a cell of dimension 2, its support is constant. This means that the support of $(X_t)_{t>0}$, seen as function of time $s_X : [0, +\infty) \to \mathcal{S}_{\mathcal{G}_1}$, is piecewise constant. The same is true for the support of the Brownian motion.

Considering these facts, we make a slight abuse and denote $\mathrm{d}X_t = (\mathrm{d}x_-(t), \mathrm{d}x_+(t))^T$ (instead of $\mathrm{d}X_t = (\mathrm{d}x_-(t), \mathrm{d}s_X(t), \mathrm{d}x_+(t))^T$, since $\mathrm{d}s_X(t) = 0$ a.e., because $s_X$ is piecewise constant), where $\mathrm{d}x_\pm(t)$ are defined inside the edges (and thus almost everywhere) in a natural way using the parametrization $(x_e)_{e\in E}$ and as 0 elsewhere (namely on the vertex set). Using the same convention for the Brownian increment $\mathrm{d}B_t$, informally, given the process $(Y_t)_{t\geq 0}$, the $(X_t)_{t\geq 0}$ component solves the following equation almost everywhere:

$$\begin{cases} X_0 \in \mathcal{G}_1 \\ \mathrm{d}X_t = -\beta_t \nabla U_{Y_t}(X_t)dt + dB_t. \end{cases} \tag{4.25}$$

Following the Remark 11, one can see that for all $y \in V$, $\nabla U_y$ is well defined and continuous a.e.. We remind the reader that on the boundary and on the points where the directional derivatives are not well defined, we consider $\nabla U_y(x) = 0$.

**Algorithm 8:** Homogenized Simulated Annealing over $\mathcal{G}_1$

```
   Data: Function $U$. Increasing inverse temperature $(\beta_t)_{t\geq 0}$. Intensity $(\alpha_t)_{t\geq 0}$
1  Initialization: Pick $X_0 \in \mathcal{G}_1$. ;
2  $T_0 = 0$ ;
3  for $k = 0 \ldots +\infty$ do
4      while $N_t^\alpha = k$ do
5          $X_t$ evolves as a Brownian motion drifted by $\nabla U_{Y_k}$, relatively to the
           structure of $\mathcal{G}_1$, initialized at $X_{T_k^-}$.
6      end
7      $T_{k+1} := \inf\{t : N_t^\alpha = k + 1\}$;
8      At time $t = T_{k+1}$, draw $Y_{k+1} = Y_{N_t^\alpha}$ according to $\nu$.
9  end
10 Output: $\lim_{t\longrightarrow+\infty} X_t$.
```

A first remark regarding Algorithm 8 is that as $k$ goes to infinity, the gap between $T_k$ and $T_{k+1}$ goes to 0, and thus we can consider it as a Euler explicit discretization of Equation (4.25). The evolution in line 5 can be simulated in two steps:

Br The extremities are perturbed using two independent normal variables of variance $T_{k+1} - T_k$, and eventually a random choice of a next edge when an extremity crosses a vertex. Furthermore, if $X_{T_k^-}$ is a cycle, we choose uniformly a separation point and split the geodesic, on that point, by cutting out around it a piece of size distributed according to a normal random variable.

Dr If needed, the extremities are moved closer to $\mathbf{Y}_k$, reducing the distance by a proportion of $\beta_{T_k}(T_{k+1} - T_k)$. An extremity approaches $\mathbf{Y}_k$ if and only if there exists a shortest path from $\mathbf{Y}_k$ to the geodesic itself that passes through it.

We give a brief explanation for why Br is a valid procedure for simulating a Brownian motion on $\mathcal{G}_1$. We know that the evolution of a Brownian motion in $\mathbb{R}^2$ is given by the evolution of two independent Brownian motions on $\mathbb{R}$ (each corresponding to one extremity). We have seen in Sub-section 4.4 that the Brownian motion inside a cell $C$ of dimension 2 is the pre-image of a Brownian motion on a compact of $\mathbb{R}^2$. This in particular means, that while in a cell of dimension 2, its evolution can be described by two independent Brownian motions on $\Gamma$. As seen before, inside an edge such a process can be simulated with a random Gaussian variable. When one of the two extremities reaches a vertex, it chooses with uniform probability, one of the permitted adjacent edges, such that the process stays in $\mathcal{G}_1$ (see an example of forbidden edges for an extremity in Figure 4.10). The last part of Br translates the effects of the condition Bc0 imposed on the generator's domain.

As before, the simulation of the process $(Y_t)_{t\geq 0}$ is straightforward, since its construction relies only on a Poisson process and independent realizations of $\nu$. We recall that between two jumping times $T_k$ and $T_{k+1}$ the process $Y_t$ is constant (and equal to $\mathbf{Y}_k$), and $(X_t)_{t\geq 0}$ is drifted by $\nabla U_{\mathbf{Y}_k}$. Indeed, as mentioned in Dr, the effect of this drift is translated by an attraction of an extremity towards $\mathbf{Y}_k$ if and only if there exists a shortest path from $\mathbf{Y}_k$ to the geodesic itself that passes through it.

Of course, each move described in Br+ Dr, needs to be conditioned on keeping the process in $\mathcal{G}_1$. Verifying if a non-geodesic path can be completed by a geodesic one to form a geodesic cycle, namely checking if an element is in $\mathcal{G}_1 \setminus \widetilde{\mathcal{G}}_1$, has no cheap formulation in terms of computational cost. However, it is rather easy to confirm that a given path $g$ is indeed a geodesic:

$$g = (x, s, y) \text{ is a geodesic} \qquad \Longleftrightarrow \qquad d(x, y) = |g|,$$

where $x, y$ are extremities of $g$, $s$ its support and $|g|$ its length. This is why we have started by implementing an algorithm (described in Algorithm 9) that finds the minimum of $U_\nu$ over the set of geodesic paths of $\Gamma$. The implementation and practical testing of Algorithm 8 is not treated here and is the subject of a future project.

### 4.6.2 Simulated annealing over the set of geodesic paths

The restriction of the initial process to the space of geodesic paths can be rigorously defined using the objects developed in Section 4.4 on $\mathcal{G}_1$, by changing the generators domain. Condition Bc0 is no longer needed, since we have excluded the cycles, and the Neumann conditions Bc1 can be adapted to reflect the process when it hits a boundary of the set of geodesic paths.

Algorithm 9 does not correspond entirely to the dynamic imposed by Equation (4.25), since it separates the drifting part of the process from the Brownian perturbation. However, following the same remarks as in Section 2.2 we consider it as a natural Euler explicit discretization of our Markov evolution: for a large value of $k$, the average gap between two jumping times $T_k$ to $T_{k+1}$ is approximately $\alpha_{T_k}^{-1} \longrightarrow 0$ as $k \longrightarrow +\infty$. On this short time interval, the process $Y_t$ is constant (and equal to $\mathbf{Y}_k$), and the drift term in Equation (4.25) is proportional to the gradient of the squared distance between $X_{T_k}$ and $\mathbf{Y}_k$. We have seen that the effect of this drift is translated by an attraction of an extremity towards $\mathbf{Y}_k$ if and only if there exists a shortest path from $\mathbf{Y}_k$ to the geodesic itself that passes through it. This condition is verified at each jump in line 10, and when needed, the attraction on each extremity is approximated by our vector $\overrightarrow{x_\pm Y_k}$, multiplied by $\beta_{T_k}$, leading to Equation (4.26).

Between two jumping times $X_t$ evolves as a Brownian motion over the set of geodesic paths (line 5). This evolution can be simulated as described in the first part of Br, keeping in mind that the process $X_t$ is reflected every time it is no longer a geodesic and will never become a cycle. Figure 4.11 proposes a schematic evolution of $(X_t, Y_t)_{t\geq 0}$ over a simple graph $\Gamma_G$ with five nodes.

**Algorithm 9:** Homogenized Simulated Annealing over the set of geodesic paths.

```
   Data: Function $U$. Increasing inverse temperature $(\beta_t)_{t\geq 0}$. Intensity $(\alpha_t)_{t\geq 0}$
1  Initialization: Pick $X_0 \in \Gamma$. ;
2  $T_0 = 0$ ;
3  for $\underline{k = 0 \dots + \infty}$ do
4      while $\underline{N_t^\alpha = k}$ do
5          $X_t$ evolves as a Brownian motion relatively to the structure of the space
           of geodesic paths, initialized at $X_{T_k}^-$.
6      end
7      $T_{k+1} := \inf\{t : N_t^\alpha = k+1\}$;
8      At time $t = T_{k+1}$, draw $\mathbf{Y}_{k+1} = Y_{N_t^\alpha}$ according to $\nu$.
9      If needed, the process $X_t$ jumps from $X_t^-$ towards $Y_{k+1}$:
10     if $\underline{d(x_\pm(t), \mathbf{Y}_{k+1}) = d(X_t, \mathbf{Y}_{k+1})}$ then
11         $$x_\pm(t) = x_\pm(t)^- + \beta_t \alpha_t^{-1} \overrightarrow{x_\pm(t)\mathbf{Y}_{k+1}}, \qquad (4.26)$$
           where $x_-(t)$ and $x_+(t)$ represent the extremities of $X_t$ and $\overrightarrow{x_\pm(t)Y_{N_t^\alpha}}$
           represents the shortest path from $x_\pm(t)$ to $\mathbf{Y}_{k+1}$ in $\Gamma_G$.
12     else
13         the extremities remain unchanged:
           $$x_\pm(t) = x_\pm(t)^-$$
14     end
15 end
16 Output: $\lim_{t \longrightarrow +\infty} X_t$.
```

### 4.6.3 Numerical Experiments

The first part of this subsection contains some practical insights on the implementation of the Algorithm 9. The second part presents the results obtained when testing our method on a toy graph and on a Facebook subgraph of 500 nodes. Improving the results and testing the algorithm on larger graphs is a current project.

#### Practical Insights

We have already seen in Section 2.4 that the algorithmic complexity of numerical strategies dealing with graphs can be problematic. Optimizing a function over $\mathcal{G}_1$ is naturally even more demanding than optimizing over the corresponding quantum graph, since the state space is 'larger'.

Although we are no longer in the same framework as in Section 2.4, the implementation of the corresponding algorithms have some similar aspects on which we will insist less.

Following Remark 6, an element of $\mathcal{G}_1$ is determined by its extremities and its support and thus encoded as a list that contains two points on $\Gamma$ (its extremities) and a list of edges (its support). We then simulate the evolution of the process $(X_t, Y_t)$ using the

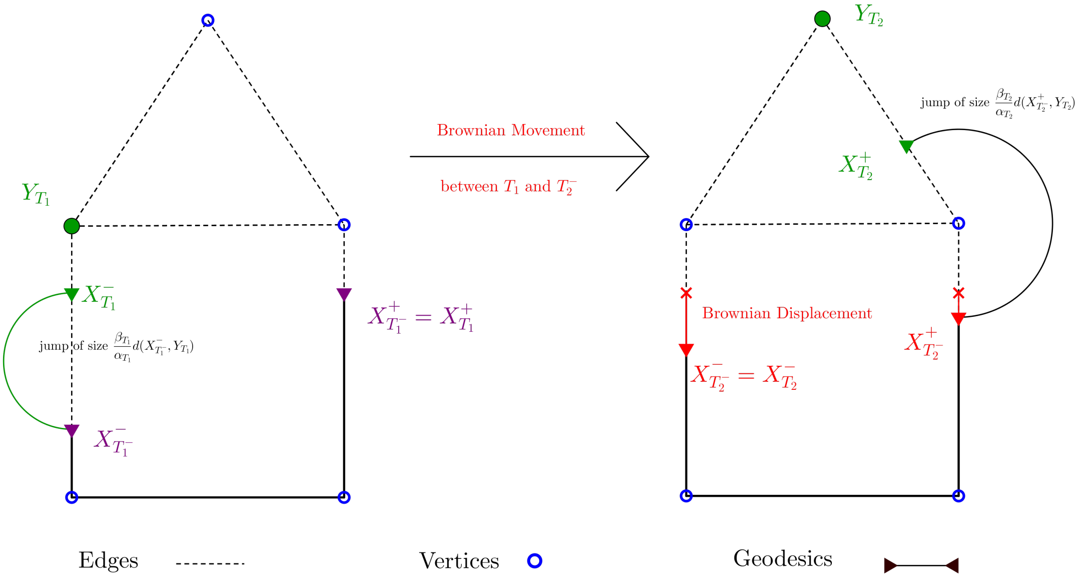

$Y_{T_2}$
jump of size $\frac{\beta_{T_2}}{\alpha_{T_2}} d(X^+_{T_2^-}, Y_{T_2})$
Brownian Movement
between $T_1$ and $T_2^-$
$Y_{T_1}$
$X^+_{T_2}$
$X^-_{T_1}$
$X^+_{T_1^-} = X^+_{T_1}$
Brownian Displacement
$X^+_{T_2^-}$
jump of size $\frac{\beta_{T_1}}{\alpha_{T_1}} d(X^-_{T_1^-}, Y_{T_1})$
$X^-_{T_2^-} = X^-_{T_2}$
$X^-_{T_1^-}$
Edges
Vertices
Geodesics

Figure 4.11: Schematic evolution of the Homogenized S.A. described in Algorithm 5. On the left hand side, we observe a jump at time $T_1$ of the first coordinate of $X_{T_1^-} = (X^-_{T_1^-}, X^+_{T_1^-})$ towards $Y_{T_1}$. The geodesic is driven by a Brownian motion during $T_2 - T_1$. A second node $Y_{T_2}$, is then sampled according to $\nu$ and the coordinate $X^+$ jumps towards it. We can see that each time an extremity of $X$ jumps if and only if there is a shortest path from $X$ to $Y$ passing through it.

procedures described in sub-section 4.6.

**Parameter tuning.** There are three main parameters that need to be taken into account. The first two are the temperature schedule $\beta_t$ and the Poisson intensity $\alpha_t$, that come from the algorithm's theoretical formulation. The third one is not explicit in Algorithm 9, but occurs naturally in practical computations, namely the stopping time. Once more, we denote these parameters :$\beta^\star_t$, $\alpha^\star_t$ and $T^\star_{\max}$. In what follows we propose an empirical set of parameters that will be used later on as reference.

**Temperature Schedule.** We remain in the classical bounds of the simulated annealing and fix a temperature schedule of the order $b \log(t+1)$, where $b$ is a positive constant. From a theoretical point of view, $b$ should be smaller than $1/c^\star(U_\nu)$, where $c^\star(U_\nu)$ is the maximal depth not containing a fixed global minimum of $U_\nu$. More details about this constant are presented in Section 2.2.3. However, in practice, this is difficult to estimate and so, following the same approach as before, we choose $\beta^\star_t$ depending on the graph's geometry:

$$\beta^\star_t = \frac{4}{\mathcal{D}_G} \log(t+1).$$

**Maximal time.** Minimizing over the set of a graph's geodesics is obviously more complex than minimizing over the graph itself. Intuitively, the difficulty of this problem is influenced by the size of the graph (number of nodes), but even more so, by the number of edges. This is why we set:

$$T^\star_{\max} = 100 + 0.1 \max\{N, |E|\}, \tag{4.27}$$

where $N$ represents the number of nodes and $|E|$ the number of edges.

**Poisson intensity.** We apply the same principles as in Section 2.4, namely, we use a linear growth and set $\alpha_t^\star = \lambda(1+t)$, with $\lambda > 0$. The constant $\lambda$ is chosen in order to generate a reasonably large amount of observations $\mathbf{Y}_k$ towards the end of the algorithm. To be more precise we can control the expected number of used observations between $T_{\text{max}}^\star$ and $T_{\text{max}}^\star - 1$ (denoted $S^\star$) through out the choice of $\lambda$. We define thus:

$$\lambda^\star = \frac{2S^\star}{2T_{max}^\star + 1} \qquad \text{with} \qquad S^\star = 1000.$$

### Numerical Results

We start by showing some results corresponding to different sets of parameters on a toy graph, represented in Figure 4.12. This graph is simple enough to facilitate the visualization of first principal components. Moreover they are all geodesic paths and thus the use of Algorithm 9 is appropriate. Also we can explicitly compute the energy barrier one needs to breach to go from a fixed global minimum to any local (or global) minimum $c^\star(U_\nu) = 1/12$.

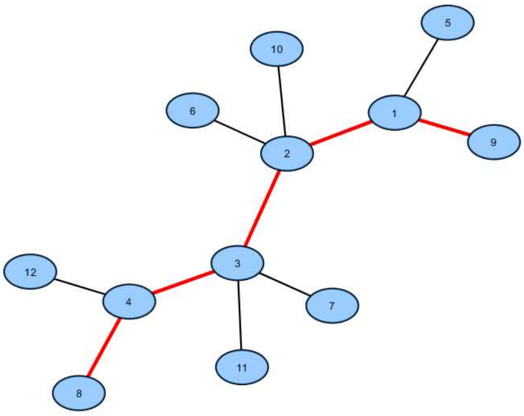


Figure 4.12: Consider that all lengths are 1 and a uniform probability over the vertex set. In red we have an example of a minimizing geodesic formed by the unique geodesic path that connects node 8 to node 9 denoted $\gamma_{8,9}$. Using this notation, the set of first principal components of the graph is: $\mathcal{M}_\nu^1 = \{\gamma_{5,8}, \gamma_{5,12}, \gamma_{8,9}, \gamma_{9,12}\}$

Since $U_\nu$ is a decreasing function, in the sense that $A \subset B$ implies $U_\nu(A) \geq B$, an easy step to improve the results of the algorithm is to take the longest geodesic in the last 10% of the iterations.

| $T_{\text{max}}$ | $S$ | $\beta$ | Error |
|---|---|---|---|
| 0.5 $T_{\text{max}}^\star$ | 1 $S^\star$ | 1 $\beta^\star$ | 36 |
| 1 $T_{\text{max}}^\star$ | 0.5 $S^\star$ | 1 $\beta^\star$ | 16 |
| 1 $T_{\text{max}}^\star$ | 1 $S^\star$ | 1 $\beta^\star$ | 9 |
| 2 $T_{\text{max}}^\star$ | 1 $S^\star$ | 1 $\beta^\star$ | 2 |
| 2 $T_{\text{max}}^\star$ | 2 $S^\star$ | 2 $\beta^\star$ | 0 |

Table 4.1: Experiments on the graph represented in Figure 4.12, while varying the parameters.

We consider that the algorithm returns a valid geodesic $g$ if the value of $U_\nu(g) \leq \min U_\nu + 1/24$. This in particular implies that $g$ has the right support and that its extremities are not far from the end of the edge. We repeat the experiments 100 times for each parameter set. The Error column in Table 4.1 represents the number of non-valid geodesics (out of a 100) returned by the algorithm.

**Facebook subgraph** We tested also our algorithm on the Facebook sub-graph of 500 nodes and 4337 edges represented in Figure 4.13 (the same sub-graph was also used in Section 2.4). The representations of graphs in this sub-section are made using the Cytoscape software, and are not a result of our algorithm. Once more, all edges are of length 1 and we consider the uniform probability on the vertex set. This sub-graph, being formed of two clearly distinctive clusters is somehow particular and has a lot of geodesics that minimize $U_\nu$. Here we consider $g$ to be a valid geodesic if $U_\nu(g) < \min U_\nu + 2/500$. This value ensures that the geodesic has a good support (the support of a minimizer). We present the results obtained for different sets of parameters in Table 4.2, where the Error column represents the number of geodesics that are not valid out of a 100.

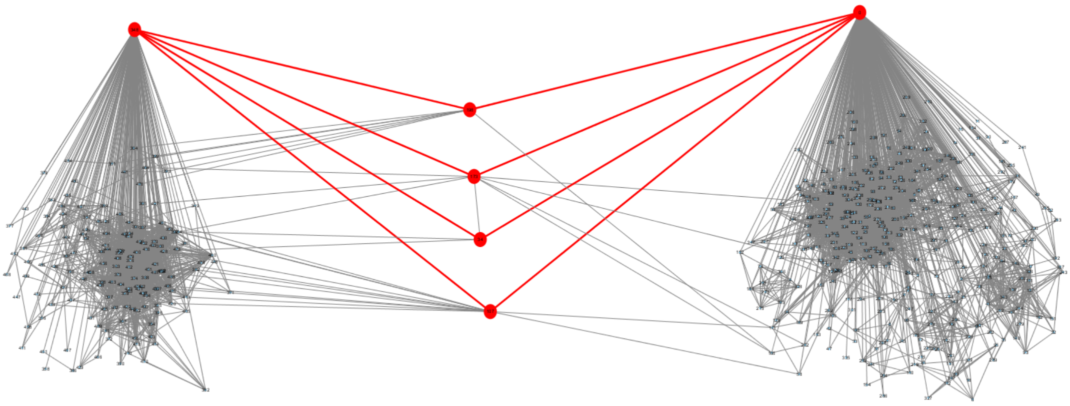


Figure 4.13: The nodes that are represented in red above each cluster are connected to all other nodes in their respective clusters. There are 4 shortest paths that connect these two nodes (also represented in red). Thus, a minimizing geodesic for $U_\nu$ is obtained by attaching at each extremity of one of these four paths an edge such that the obtained path is still a geodesic. The nodes we choose as extremal nodes for a first principal component in this case are not important. What matters is the 'core' of first principal components, namely the paths represented in red.

| $T_{\max}$ | $S$ | $\beta$ | Error | Av. Time |
|---|---|---|---|---|
| $\mathbf{T}^\star_{\max}$ | $S^\star$ | $\beta^\star$ | 14 | 3.38 s |
| $2\mathbf{T}^\star_{\max}$ | $S^\star$ | $\beta^\star$ | 7 | 6.49 s |
| $\mathbf{T}^\star_{\max}$ | $2\ S^\star$ | $\beta^\star$ | 15 | 6.30 s |
| $\mathbf{T}^\star_{\max}$ | $S^\star$ | $2\beta^\star$ | 1 | 3.42 s |
| $\mathbf{T}^\star_{\max}$ | $S^\star$ | $4\beta^\star$ | 10 | 3.32 s |
| $2\mathbf{T}^\star_{\max}$ | $2\ S^\star$ | $2\beta^\star$ | 1 | 13.41 s |

Table 4.2: Experiments on the graph represented in Figure 4.13, while varying the parameters. The maximal time is chosen as $\mathbf{T}^\star_{\max} = 100 + 0.1N$.

We can see that, as expected, increasing the maximal time improves the performance of the algorithm but also increases the computational time. However, we observe that increasing the number of observations in this case is not very productive since it increases the computational time and has no real impact on the success rate. Doubling the value of the temperature schedule $\beta$ seems to improve the performance, since it has the effect of accelerating the convergence. Nonetheless, increasing it to much can back fire, since the algorithm might get stuck in local minimums (one can observe that the performance decreases from $2\beta^\star$ to $4\beta^\star$).

**Concluding remarks.** The work presented in this chapter is a work in progress. From a theoretical point of view, Conjectures 1 and 2 need to be proved. From the simulation point of view, we still need to test Algorithm 8 on larger graphs. Performing more tests in a general manner could help us adjust the parameters in order to increase the performance. A following step would then be to implement Algorithm 9.

# Part II

# Noisy Simulated Annealing

# Chapter 5

# Noisy Simulated Annealing

In this chapter we propose a modified version of the simulated annealing algorithm for solving a stochastic global optimization problem. More precisely, we address the problem of finding a global minimizer of a function with noisy evaluations. We provide a rate of convergence and its optimized parametrization to ensure a minimal number of evaluations for a given accuracy and a confidence level close to 1. This work is completed with a set of numerical experiments and assesses the practical performance both on benchmark test cases and on real-world examples (motivated by Aircraft Trajectory optimization).

**Note to the reader:** The work presented in this chapter is a joint work with Clément Bouttier. It has been submitted for publication to the Journal of Machine Learning Research.

## 5.1 Introduction

We are interested in an algorithm that solves the stochastic global optimization problem:

$$\text{Find} \quad x^\star = \operatorname*{arg\,min}_{x \in E} \mathbb{E}_\omega(U(x,\omega)), \tag{*}$$

where $x$ is a decision variable belonging to some large space $E$, $\omega$ is a random variable and $U$ is the cost, a positive and bounded real valued function. We do not make any assumption on the regularity of $U$. We only expect it to be rapidly evaluable but not necessarily immediate: typically, $U$ is the result of some short numerical simulation, e.g. an industrial aircraft performance computation. We do not make any distribution assumption for the random inputs $\omega$ themselves but only on the outputs $U$. We assume $U$ has some robustness property in the sense that, at some point $x$, it is either infinite for all $\omega$ or bounded uniformly in $\omega$.

This problem is twofold: we must both estimate and minimize the expectation of the cost. A simple and general approach consists in the minimization of a sample average of Monte Carlo estimators:

$$\widehat{\mathbb{E}}_{\omega,N}(U(x,\omega)) := \frac{1}{N}\sum_{i=1}^{N} U(x,\omega_i),$$

for any given i.i.d. sample $(\omega_i)_{1\leq i\leq N}$ of size $N$ distributed according to the distribution $P_\Omega(x)$ of $\omega$ and for any $x \in E$. Such an estimator consistently estimates $\mathbb{E}_\omega(U(x,\omega))$ for any given $x$. Nevertheless, its accuracy is directly linked to $N$ and thus to the computational effort. One can thus wonder if a computationally efficient procedure using that estimator can return a solution to the initial problem given a certain level of accuracy $\epsilon$.

### 5.1.1 Previous works: different types of algorithms

There were many attempts to solve this stochastic problem across several research communities. We give a brief survey of them in what follows.

When $E$ is finite and $U$ takes its values in $[0, 1]$, problem (*) is labelled as a "simple regret bandit optimization problem" by the bandit community. Indeed it can be seen as the problem of choosing, among a small finite set of slot machines providing random rewards, the one with the best expected reward by playing a minimal number of times. This is not the classical setting of bandit optimization which usually seeks for the "cumulative regret". As algorithm proposals for the simple regret context often extends cumulative regret concepts [16], we focus on them first. The Upper Confidence Bound (UCB, [18]) algorithm aimed at building sequences of confidence bounds around the estimated expected cost of each element of the search space. If the space is too large this can be prohibitive. These were several attempts to bypass this issue by adding some assumptions on the regularity of the cost function around its optimum. We can mention Hierarchical Optimistic Optimization (HOO, [40]) algorithm that produced guarantees about the cumulative regret for a continuous Lipschitz cost function with known Lipschitz constant. In the same framework Stochastic Simultaneous Optimistic Optimization (StoSOO, [102]) algorithm relaxed this last assumption. Both algorithms were however not very efficient in practice if the search space is multidimensional. Indeed they still required some uniform exploration of the state space in the first phase. This could lead to numerical difficulties when the dimension was growing.

These algorithms could all be viewed as stochastic variations around the classical *branch and bound* algorithm [94], which was extensively studied by the optimization community. We can mention the very popular DIRECT algorithm [86], which is very similar to the StoSOO procedure.

Finally, let us mention the computer experiment community that introduced another popular global optimization method for dealing with the stochastic case, the so called Efficient Global Optimization (EGO, [87]) based on expected improvement. The convergence rate of this method was already investigated by Bull [41] in a noise-free context. This algorithm focused on minimizing the number of cost evaluations because it considered a setting where the cost evaluations were very time consuming. As a result, in order to select each evaluation point, it required a higher computational effort and memory storage per iteration than other optimization methods. Such a method could therefore turn out to under-perform in a setting where the computational cost ratio between selection and evaluation was inverted.

A typical algorithm that was known to perform well in the case of time-cheap cost evaluations was the simulated annealing (SA) as mentioned by Locatelli in the Handbook of Global Optimization edited by [79]: *"The latter algorithms* (mainly EGO) *often outperform SA algorithms from the point of view of the number of function evaluations to reach a given relative accuracy on the standard test functions from [52], but usually require a higher computational effort per iteration. Typical advantages of SA algorithms are their very mild memory requirements and the small computational effort per iteration. If the cost of a function evaluation is very high, then even a considerable computational effort per iteration may be negligible with respect to the cost of a function evaluation, and algorithms which require few function evaluations are preferable; otherwise, also the computational effort per iteration should be taken into account, and from this point of view SA algorithms are often better than other algorithms."* This is one of the reason why we have chosen the simulated annealing for solving the aircraft trajectory optimization problem.

However SA algorithms have been designed and extensively studied in a context where the exact cost could be observed. We recall below some basic facts in the noiseless case (Section 5.1.2) and then present the noisy case which is the setting addressed in this chapter (Section 5.1.3).

### 5.1.2 Simulated Annealing without noise

Let $E$ be some finite search space and $J : E \to \mathbb{R}_+$ a function that we want to minimize, called cost thereafter.
Simulated Annealing is a classical global optimization method. It aims at building a sequence of elements from $E$ whose last element is drawn from a uniform probability law on the subset of *global* minima of $J$. In other words it aims at sampling from the following distribution

$$\mu^\star = \frac{\mathbb{1}_{S_{opt}}}{card\,(S_{opt})} \ ,$$

where $S_{opt} = \{x, J(x) = \min_{y \in E} J(y)\}$ and $card(.)$ denotes the cardinality of a set. Such a sampling is of course not straightforward but one can notice that this distribution can be rewritten in the following form:

$$\forall x \in E,\ \ \mu^\star(x) = \lim_{T \to 0} \frac{e^{\frac{-J(x)}{T}}}{\sum_{y \in E} e^{\frac{-J(y)}{T}}} \ ,$$

and it is well-known that the Gibbs distributions of the form $\mu_T = e^{\frac{-J}{T}} / \sum e^{\frac{-J}{T}}$ are efficiently sampled for reasonably low temperatures $T \in \mathbb{R}_+$ using the Metropolis-Hastings algorithm [1]. A quite natural attempt is therefore to build a sequence of sequences obtained using Metropolis-Hastings algorithm for a set of decreasing temperatures. In particular, at a very low temperature, the Metropolis-Hastings algorithm generates exploratory moves that are accepted with very low probabilities, which makes it a very bad sampler. Therefore it is necessary to first encourage exploration by using a sampling at higher temperatures. A lower bound on the temperature at each step ensuring a convergence in probability of the algorithm has been provided by Hajek [73]. At the same time another proof of convergence using modern semi-group representation of Markov processes has been obtained by Holley and Stroock[77]. The obtained bounds are less explicit but contain information about the convergence rate and the proof scheme is much more general. We set our work in the continuity of this last work and use similar notations.

### 5.1.3 Simulated Annealing with noisy evaluations

As mentioned previously, our main interest is to extend such a method of simulated annealing to the stochastic case:

$$\text{Find} \quad x^\star = \arg\min_{x \in E} \mathbb{E}_\omega(U(x, \omega))$$

where $\omega$ is a random input of a bounded cost function $U$ whose expectation can only be numerically estimated through Monte Carlo simulations. In other words, we consider $J(.) = \mathbb{E}_\omega(U(., \omega))$. This question is not novel and several attempts were made to address this problem theoretically in the 90's. [67] were probably the first ones to introduce the notion of simulated annealing with noisy measurements. They assumed an additive Gaussian noise independent of the evaluation point and gave a sufficient condition for the decrease of the variance $\sigma_k^2$ of this noise, to ensure convergence of the algorithm to the optimal set. Gutjahr and Pflug [72] extended the results to distributions that are more peaked around zero than the Gaussian distribution. Their convergence result can be stated roughly as follows:

**Theorem 7** (Gutjahr and Pflug [72]). *Let $(X_k)_{k\in\mathbb{N}}$ denote the sequence of states in $E$ visited by the simulated annealing algorithm with Monte Carlo sampling of the noisy measurements. If:*

*(i) the convergence conditions from [73] are satisfied*

*(ii) $\exists\epsilon > 0$ such that the standard error of the noise at step $k$ of the algorithm $\sigma_k^2 = \mathcal{O}(k^{-(2+\epsilon)})$*

*then $\forall x \in E$, $\lim_{k\to+\infty} \mathbb{P}(X_k = x) = \mu^\star(x)$, where $\mu^\star$ is the uniform distribution on the global minima of the expected cost.* ■

This result provided a first answer to our question about the convergence of the algorithm in the stochastic case. However the convergence statement above did not give any information about the convergence rate of the algorithm. Following the noise-free proof of Aarts and Korst [1], Homem-de Mello [78] provided an extension of this statement to the noisy case with bounded variance and introducing a state dependent noise. He obtained the same constraint on the decrease of the variance and the same convergence statement. He also highlighted the need for an extended result concerning the rate of convergence and for numerical experiments. Indeed on this second point we can mention the works of Fink [60] and Branke [38] that addressed this issue. Fink [60] made a very interesting proposition in the framework of Gaussian noise. He proposed to use the noise of measurement to drive the simulated annealing, *i.e.*, accept a move if the estimated cost of the proposed solution is lower than the one of the current solution. Using an analogy with the Glauber acceptance mechanism, which is a symmetric alternative to the Metropolis-Hasting mechanism [1], he proposed a far more efficient criteria for the variance decrease, *i.e.*, $\sigma_k = \mathcal{O}(log(k)^{-2})$. Unfortunately he only provided a few numerical examples to validate his statement and a theoretical proof is still missing.

### 5.1.4 Main contributions

In this chapter we consider a simulated annealing algorithm based on mini-batches of increasing size. More precisely, at each iteration, the expected cost is estimated by Monte Carlo sampling of increasing sizes. The estimated cost at step $k$ of the algorithm is thus $\widehat{\mathbb{E}}_\omega(U(x_k,\omega)) = 1/N_k \sum_{i=1}^{N_k} U(x_k,\omega_i)$, where $N_k$ is an increasing sequence and $\omega_i$ are i.i.d. random variables having the same law as $\omega$. The cost can be written also as $\widehat{\mathbb{E}}_\omega(U(x_k,\omega)) = \mathbb{E}_\omega(U(x_k,\omega)) + \zeta_\omega(x_k)$, where $\zeta_\omega(x_k)$ is some bounded random variable. We denote $\sigma_k^2 := \mathrm{Var}(\zeta_\omega(x_k))$, the variance of post-sampling noise. As it is directly linked to the number of measurements made during the mini-batch, it can be tuned by the user.

**Rate of convergence for all variances of polynomial decay.** In the sequel we first show that theoretical guarantees of Theorem 7 can be extended to sub-Gaussian random variables (e.g., bounded noise distributions) with stronger convergence results for this algorithm. Indeed we show that convergence can be ensured if the number of measurements is chosen such that $\sigma_k = \mathcal{O}(k^{-(\alpha/2)})$ with $\alpha > 0$, which corresponds to $N_k$ of the order $k^\alpha$. One can observe that, as opposed to [72], the convergence still holds for $\alpha \leq 2$. This is summarized in Theorem 8.

We derive the rate of convergence of the procedure (Theorem 10) and optimize it (Corollary 1) with respect to the noisy simulated annealing algorithm parameters in order to provide a minimal total number of measurements at given accuracy and confidence requirements. This leads to the optimal value $\alpha = 2$ for which the number of cost evaluations increases fast enough to ensure almost the same convergence rate as in the noise-free case. This shows that the convergence rate is limited by the concentration speed of the Gibbs measure around its modes. According to our concentration result, increasing the estimation effort cannot increase the performance of the algorithm above this limit. On the other hand the convergence still holds for a decreased estimation effort ($\alpha < 2$) as soon as the cooling schedule is slowed consequently.

**Computational cost in the general case.** Finally, we derive an upper bound on the computational time-complexity of our simulated annealing algorithm (with noisy measurements). This quantity is roughly of the order of:

$$e^{\frac{C_1 \log \frac{1}{\delta}}{\epsilon}},$$

where $C_1$ is some constant depending on the cost function itself as detailed in Corollary 1. The provided bound exhibit an exponential dependency in $1/\epsilon$ and $\log(1/\delta)$. This is comprehensive regarding the generality of the considered problem.

**Computational cost in the absence of local minimum.** If the function has no local minimum apart from the global minimum (e.g., a convex function evaluated on a finite set) the temperature schedule can be adapted and the computational cost becomes of the order of:

$$\left(\frac{C_2 \log \frac{1}{\delta}}{\epsilon}\right)^3,$$

where $C_2$ is a constant detailed in Corollary 2. This second bound increases in a polynomial way with respect to $1/\epsilon$ and $\log 1/\delta$. This is a very positive result as it shows that, if stronger hypotheses on the cost are considered, the noisy simulated annealing algorithm recovers the polynomial state-of-the-art convergence guaranties instead of an exponential one. This result is discussed more precisely after Corollary 2.

**Numerical experiments.** We provide numerical evidence indicating that the numerically observed requirements by Fink [60], *i.e.*, $\sigma_k = \mathcal{O}(\log(k))$, do not hold for a Metropolis-Hastings Acceptance criteria. We apply the noisy simulated annealing on classical non convex optimization test cases with different level of noise. Of course, we also perform a test on the stochastic aircraft trajectory optimization problem using the industrial aircraft performance model. The results are provided in the last section of this chapter (Section 5.6).

### 5.1.5 Outline of the chapter

This chapter is organized as follows. In Section 5.2 we present the noisy simulated algorithm and our main theoretical result. In Sections 5.3, 5.4 and 5.5 we provide the proof of this statement. More precisely, in Section 5.3 we compute the infinitesimal generator of the noisy simulated annealing algorithm. In Section 5.4 we compare it to the one of the noise-free simulated annealing algorithm from [77]. This enables us to derive a differential inequality for a $L^2$ distance between the distributions of the two previously mentioned processes. Integrating by applying Grönwall's Lemma Section 5.5, we obtain obtain our convergence result. In the same section, we show how to tune the parameters of the algorithm in order to optimize the performance bound and give the corresponding computational cost. In Section 5.6 we propose some numerical insight on synthetic experiments and the aircraft trajectory optimization problem.

## 5.2 Noisy Simulated Annealing algorithm: statement and convergence result

We first present our extended version of the simulated annealing to the stochastic case, whose pseudo-code can be found in Algorithm 10.

### 5.2.1 Noisy Simulated Annealing algorithm (NSA)

**Algorithm 10:** Noisy Simulated Annealing

```
Data: Neighborhoods structure $(S_x)_{x \in S}$, Initial guess $x_0$, increasing function
      $\beta : \mathbb{R}_+ \to \mathbb{R}_+$, Function $t \mapsto n_t$
Initialization: Initialize time $t_0 = 0$, $\beta_0 = \beta(t_0)$
for k from 0 to Maximal number of iterations do
    Draw one solution candidate: $\widetilde{x}_{t_k} \in S_{x_{t_k}}$ according to $q_0(x_k, \cdot)$
    Draw $N_{t_k} \sim \mathcal{Poisson}(n_{t_k}) + 1$
    Draw $2N_{t_k}$ simulation conditions independently:
        $(\omega_1^k, ..., \omega_{N_{t_k}}^k) \sim (P_\Omega(x_{t_k}))^{\otimes N_{t_k}}$ and $(\widetilde{\omega}_1^k, ..., \widetilde{\omega}_{N_{t_k}}^k) \sim (P_\Omega(\widetilde{x}_{t_k}))^{\otimes N_{t_k}}$
    Compute estimates $\widehat{J}(x_t)$ and $\widehat{J}(\widetilde{x}_t)$ using the $N_{t_k}$ conditions:
        $\widehat{J}(x_{t_k}) = \frac{1}{N_{t_k}} \sum_{i=1}^{N_{t_k}} U(x_{t_k}, \omega_i^k)$ and $\widehat{J}(\widetilde{x}_{t_k}) = \frac{1}{N_{t_k}} \sum_{i=1}^{N_{t_k}} U(\widetilde{x}_{t_k}, \widetilde{\omega}_i^k)$
    Draw an exponential random variable $\xi_{k+1}$ of parameter 1
    Update time $t_{k+1} := t_k + \xi_{k+1}$
    Draw $u$ according to a uniform distribution on $[0, 1]$
    if $u \leq e^{-\beta_k (\widehat{J}(\widetilde{x}_{t_k}) - \widehat{J}(x_{t_k}))}$ then
        set $x_{t_{k+1}} := \widetilde{x}_{t_k}$
    else
        set $x_{t_{k+1}} := x_{t_k}$
    end
    Increase the inverse of the temperature $\beta_{k+1} := \beta(t_{k+1})$
end
return $x_{t_{k+1}}$
```

As in the deterministic setting, the algorithm requires an initial feasible solution $x_{t_0}$, a temperature schedule $T_t$ (we will mostly use its inverse $\beta_t = 1/T_t$), and a good neighbourhood structure. What we mean by good will be specified in the definition of $q_0$. The algorithm explores the state space in the following manner. After $k$ iterations, at time $t_k$, it selects a random neighbouring solution $\widetilde{x}_{t_k} \in S_{x_{t_k}}$ ($S_{x_t}$ being the set of neighbours of $x_{t_k}$) according to a proposition law. Then it compares the estimate $\widehat{J}(\widetilde{x}_{t_k})$ of the cost of this new solution to the estimated cost $\widehat{J}(x_{t_k})$ of the current solution and then it decides to substitute (or not) the new to the current:

- if the estimated cost of the new state is lower than the current one, *i.e.*, $\widehat{J}(\widetilde{x}_{t_k}) \leq \widehat{J}(x_{t_k})$, the move is accepted, *i.e.*, $x_{t_{k+1}} \leftarrow \widetilde{x}_{t_k}$

- if not, it is only accepted with a probability $\exp(-\beta_{t_k}(\widehat{J}(\widetilde{x}_{t_k}) - \widehat{J}(x_{t_k}))$.

The time $t$ is then updated using independent exponential random variables, enabling us to consider the NSA as a continuous time Markov process.
For convenience, in this chapter, we denote by $\lfloor . \rfloor_+$ the positive part function , *i.e.*for all $x \in \mathbb{R}$, $\lfloor x \rfloor_+ = 0$ if $x \leq 0$ and $x$ if not.

### 5.2.2 General setting and notations

To state the convergence of Algorithm 10, we first need to describe formally the framework we are working in. Notations introduced in this section are valid for the whole chapter unless mentioned explicitly.

- Regarding the noise structure and the estimation procedure, we denote:

  $\widehat{J}$ the estimated cost: $\widehat{J} : E \to \mathbb{R}_+$, such that $\forall\ x \in E, \widehat{J}(x) = \frac{1}{N}\sum_{i=1}^{N} U(x, \omega_i)$, where: $(\omega_1, ..., \omega_N)$ is a $N$ i.i.d. vectors sequence drawn from distribution $P_\Omega(x)$

  $\xi_k$ the time increments: $(\xi_k)_{k\in\mathbb{N}}$ is a sequence of i.i.d. exponential random variables of parameter 1

  $t_k$ the jumping times: $\forall k \in\ \mathbb{N}, \quad t_k = \sum_{i=1}^{k} \xi_i$.

  $n_t$ the samplig intensity: $n_t$ a continuous increasing function.

  $N_{t_k}$ the sample sizes: $N_{t_1}, N_{t_2}, \ldots N_{t_n}$ are independent for all $n \in \mathbb{N}$ and all $0 < t_1 < t_2 \ldots < t_n$ and
  $$N_{t_k} \sim \mathcal{P}oisson(n_{t_k}) + 1,$$

  We can make a few remarks about the different notations. The construction of $N_t$ ensures that its value is a strictly positive integer at all times. The reason why we choose to have a randomly sized sample for the Monte Carlo estimation procedure is rather technical. It enables to generate a continuous transition probability as it can be noticed in Equation (5.3) and ease the formulation of the infinitesimal generator (Equation (5.7)).

- About the state space, we denote:

  $E$ a finite state space.

  $S$ a neighbourhood structure such that $E$ is connected with respect to it, *i.e.*, $S$ is a connected graph containing all the points in $E$. For any $x$ in $E$, we denote $S_x$ the set of its direct neighbours.

  $\mu_0$ the initial distribution, a probability measure that charges every point of a subset of interest $E' \subset E$ defined more precisely in ($U$) ,

  $q_0$ the proposition law, an irreducible and $\mu_0 - reversible$ transition probability, *i.e.*, $\forall x, y \in E$, $\sum_{n=0}^{\infty} q_0^{(n)}(x, y) = \infty$ and $\mu_0(x)q_0(x, y) = \mu_0(y)q_0(y, x)$. In addition we assume that for any $x$ in $E$, we have $q_0(x, S_x) = 1$

  Considering a finite search space $E$ enables us to use the spectral gap inequality in provided by Holley and Stroock [77], quoted in Theorem 9, and overcome differentiation-under-the-integral-sign issues in Equation (5.16). However, it could be replaced by coercivity assumptions on the function $J$, which could be more general but not really well suited for the application we are considering. It is our most restrictive assumption. Nevertheless it is in line with previous works on noisy global optimization for example: [72], [78] or [60]. It corresponds to a historical use of simulated annealing for problems with huge finite search space like for the traveling salesman problem [2]. Mimicking [77], we might however relax this assumption of finiteness. Nevertheless it requires more technicalities as in [1] and this is left for future work.

We assume that the algorithm can visit and start from every point in the solution space through the connection assumption $S$ and the definition of $\mu_0$. The proposition law $q_0$ defines the way a new solution $\widetilde{x}$ is proposed to the NSA at each iteration. The irreducibility of $q_0$ implies the fact that one can go from any state $x$ to any other state $y$ using the neighbourhood structure $S$, in a finite number of steps. The $\mu_0 - \mathit{reversibility}$ is used to simplify the notations. A classical choice [1] for $q_0$ and $\mu_0$ is: $\forall\ x, y \ \in \ E,\ \mu_0(x) = \frac{1}{|E|}$ and $q_0(x, y) = \frac{1}{|S_x|}$, assuming every point in $E$ to have the same number of neighbors. However there are other possible choices for $\mu_0$ and $q_0$. This last two assumptions are inherited from the classical Metropolis-Hasting sampling algorithm which corresponds to the NSA algorithm with no cooling mechanism and no noise. They ensure that a run in this setting, starting from any point of the search space, converges to a stationary distribution which is the Gibbs measure associated to $J$.

- About the cost function, we consider:

  $U$ the underlying cost: there exist $M > 0$ and $E' \subset E$, such that $U$ is bounded and non-negative on $E'$:

$$\forall x \ \in \ E',\ \forall \omega,\ 0 \leq U(x, \omega) \leq M$$

  and $U$ is infinite on $E \backslash E'$:

$$\forall x \in E \backslash E',\ \forall \omega,\ U(x, \omega) = +\infty.$$

  The assumption about $U$ being bounded is not very restrictive. It reflects the practical setting where a simulation code crashes out of the definition domains. We associate infinite costs to crashes and thus ($U$) is rather a consequence of ($E$).

- About the algorithm parametrization, we denote

  $\beta_t$ the inverse of the temperature: a positive increasing real function of $t$,

  $\alpha$ the sampling size (expected number of simulations): $\exists \alpha \in \mathbb{R}_+$ such that $n_t = (t+1)^\alpha$,

  $\beta_t$ is usually chosen such that $\forall t \ \in \ \mathbb{R}_+,\ \frac{d\beta_t}{dt} = \frac{bd}{1+td}$ for some $b, d \in \mathbb{R}_+$, as it was shown by [73] and [77] to be a necessary condition to ensure the convergence of the simulated annealing algorithm for any cost function. There is no reason to expect that the noisy context would be more favorable than the deterministic one. As suggested by the definition of $\alpha$, we choose a polynomial growth of the number of simulations for the cost estimation. We show later on in this work that this ensures the convergence of the noisy simulated annealing for a good choice of $\alpha$ and $b$.

### 5.2.3 Tool for the analysis: the NSA process

We now present the mathematical formalization of the NSA algorithm's underlying stochastic process. First, for pedagogical purposes, we omit the temperature evolution and noisy measurements. The NSA algorithm then becomes a simpler Markov chain exploring the state space $E$ according to the Markovian transition matrix whose elements are of the form:

$$\mathbb{P}(x \to y) = q_\beta(x, y) = \begin{cases} q_0(x, y) e^{-\beta \lfloor J(y) - J(x) \rfloor_+} & \text{if } y \neq x \\ 1 - \sum\limits_{z \in E \backslash x} q_\beta(x, z) & \text{if } y = x, \end{cases} \tag{5.1}$$

This reflects the transition mechanism introduced at the beginning of this section. As the process is in fact a continuous one, we must also consider the time component. NSA jumps happen at stochastic times and the probability of acceptance depends on these times. Combining the law of the jumping times and the previous mechanism, we can make their joint transition probability explicit:

- Let $(\widetilde{\chi}_k, T_k)_{k \in \mathbb{N}}$ a $E \times \mathbb{R}_+$-valued Markov chain such that $\forall k \in \mathbb{N}$, $\forall y \in E$, $\forall u \in \mathbb{R}_+$:

$$\mathbb{P}(\widetilde{\chi}_{k+1} = y, T_{k+1} \geq u | \widetilde{\chi}_k, T_k) = \int\limits_u^{+\infty} \widetilde{q}_{\beta_\tau}(\widetilde{\chi}_k, y) \mathbb{1}_{[T_k, +\infty[}(\tau) e^{-(\tau - T_k)} d\tau, \tag{5.2}$$

  where

$$\widetilde{q}_{\beta_t}(x, y) = \begin{cases} q_0(x, y) \mathbb{E}_{N_t} \mathbb{E}_{\omega_1, \ldots, \omega_{N_t}} \left( e^{-\frac{\beta_t}{N_t} \lfloor \sum_{i=1}^{N_t} U(y, \omega_i) - U(x, \omega_i) \rfloor_+} \right) & \text{if } y \neq x \\ 1 - \sum\limits_{x \neq z} \widetilde{q}_{\beta_t}(x, z) & \text{if } y = x \end{cases} \tag{5.3}$$

  This is a similar construction to the one of the classical simulated annealing process [73]. The state transition mechanism must also reflect the estimation procedure, therefore the form of Equation (5.3) differs from Equation (5.1). As mentioned before the function $t \mapsto \beta_t$ represents the inverse of the temperature schedule and $N_t$ is the random process described by $N_{t_k}$. The jumping times, or evaluation times of the process happen at times defined by the sequence $T_k$ (*cf.* the definition of $t_k$).

  The chain $(\widetilde{\chi}_k)_{k \geq 0}$ explores the state space $E$ using a transition probability $\widetilde{q}_{\beta_t}$ constructed in a same way as the classical one, replacing the exact value of $-\beta_{T_k}[J(y) - J(x)]_+$ by its Monte Carlo estimation. The expected value from the formula comes from the fact that, as mentioned in the definition of $\widehat{J}$ and in Algorithm 10, we use a random number of Monte Carlo shootings for the estimations.

Finally, we obtain the NSA process by associating the two sub-processes as follows:

- Let $\left(\widetilde{X}_t\right)_{t \geq 0}$ be the inhomogeneous Markov Process such that $\widetilde{X}_t = \widetilde{\chi}_k$ if $T_k \leq t < T_{k+1}$. One can see that this process is piecewise constant and jumps at exponential times from one candidate solution to another, in other words $(\widetilde{X}_t)_{t \geq 0}$ is just the continuous-time version of the discrete time noisy simulated annealing process, $(\widetilde{\chi}_k)_{k \geq 0}$.

  Note that, if $y \in E \backslash E'$ then $\forall x \in E'$, $\quad \widetilde{q}_{\beta_t}(x, y) = 0$. Hence, if the initial solution $\widetilde{X}_0$ is chosen in $E'$, then $\forall t \geq 0$, $\widetilde{X}_t \in E'$.

### 5.2.4 Convergence result

We denote:

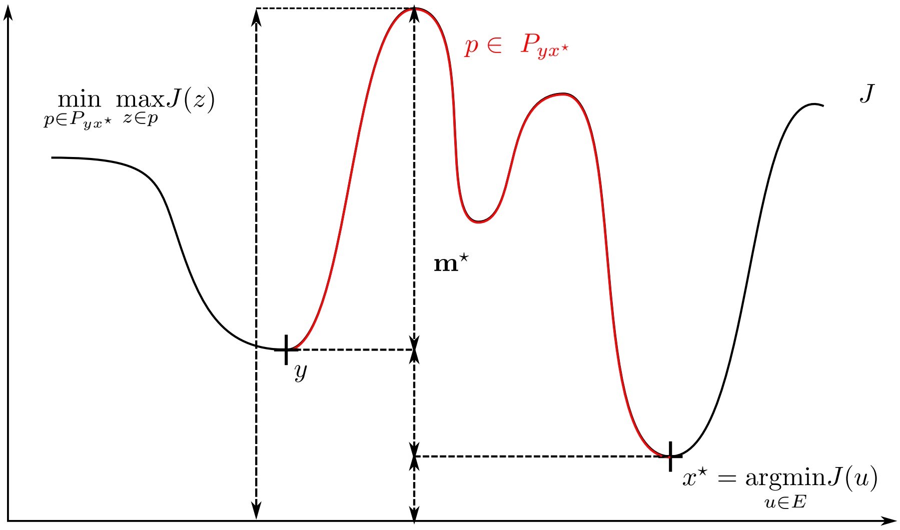


Figure 5.1: $m^\star$, maximal depth of local minima

- $m^\star$ the maximum depth of a well not containing a fixed global minimum of the function $J$. To be more precise, we call a path from $x$ to $y$ any finite sequence $x_0 = x, x_1, \ldots, x_n = y$ such that for all $i$, $x_{i+1} \in S_{x_i}$. Let $P_{xy}$ be the set of paths from $x$ to $y$.

  For a given path $p \in P_{x,y}$, the elevation of the function $J$ on $p$ is $\max_{z\in p} J(z)$. Minimizing this quantity over the set of possible paths $P_{x,y}$, gives us the elevation of the cheapest path going from $x$ to $y$. Denote this elevation by:

  $$H_{x,y} = \min_{p\in P_{xy}} \left\{ \max_{z\in p} J(z) \right\}$$

  Then

  $$m^\star := \max_{x,y\in E} \left\{ H_{x,y} - \max\left(J(y), J(x)\right) \right\} \tag{5.4}$$

  As represented on Figure 5.1, $m^\star$ can also be understood as the highest energy barrier to climb to go from one point to another in the search space in the easiest direction. As mentioned before, it also represents the maximal depth of a well not containing a fixed global minimum. If $x^\star$ is a global minimum then:

  $$m^\star = \max_{y\in E} \left\{ H_{x^\star,y} - J(y) \right\}.$$

  The definition provided here is equivalent to the classical one, *i.e.*, the one provided by [73] and [77]. A proof of this statement can be found in Appendix A.2.

- $\gamma(\beta)$ the spectral gap between 0 and the rest of the $L^2(\mu_\beta)$ spectrum of $-L_\beta$, where $L_\beta$ is the generator of the classical simulated annealing (for more details about $L_\beta$ see Section 5.3):

  $$\gamma(\beta) := \inf \left\{ -\int \phi L_\beta \phi d\mu_\beta \text{ s.t. } \int |\phi|^2 d\mu_\beta = 1 \text{ and } \int \phi d\mu_\beta = 0 \right\} \tag{5.5}$$

Following [77], we know that given $E$, $\mu_0$ and $U$, there exists a constant $c$ such that:

$$\forall \beta \geq 0, \quad \gamma(\beta) \geq c e^{-m^\star \beta}$$

Remark that this lower bound is mainly informative for small values of $\beta$. In addition set:

$$\forall x \in \ E,\ J(x) = \mathbb{E}_\omega(U(x,\omega)) \text{ and } J^\star = \min_{x\in E} \mathbb{E}_\omega(U(x,\omega)).$$

We define $\chi_\epsilon$ the set of $\epsilon$-optimal points in $E$, *i.e.*,

$$\chi_\epsilon = \{x : J(x) \leq J^\star + \epsilon\}, \tag{5.6}$$

and denote ${}^{\mathsf{c}}\chi_\epsilon = E\backslash\chi_\epsilon$, its complementary in $E$. We also write $a \wedge b = \min a, b$.

**Theorem 8.** *Consider the settings of Section 5.2.2, if $\beta_t = b\log(td+1)$ and $n_t = (t+1)^\alpha$, with:*

$$\{m^\star b < 1 \wedge \alpha/2\} \text{ or } \{m^\star b = 1,\ \ \alpha > 2 \text{ and } d < 2cm^\star/M\},$$

*then there exits $C > 0$ such that, $\forall t \in \mathbb{R}_+$, $\forall \epsilon > 0$,* $\quad \mathbb{P}(\widetilde{X}_t \in {}^{\mathsf{c}}\chi_\epsilon) \leq C(\mu_{\beta_t}({}^{\mathsf{c}}\chi_\epsilon))^{1/2}.$ ■

This theorem is a natural extension of the result provided by Holley and Stroock [77]. There are two main interesting facts to point out. First, we obtain a balance between the expected number of Monte Carlo simulations at each step of the algorithm and the inverse of the temperature, *i.e.*, $\{m^\star b < 1 \wedge \alpha/2\}$ or $\{m^\star b = 1,\ \ \alpha > 2$ and $d < 2cm^\star/M\}$. Reducing the growth rate $\alpha$ of the number of simulations below the quadratic rate should be compensated by decreasing accordingly the temperature factor $b$. Second, the convergence is stated in terms of a bound on the probability of not returning an optimal solution. Using the concentration speed of the Gibbs measure one can deduce a rate of convergence of the algorithm. Also the theorem provides an insight on how the algorithm could be used in practice. A run of parallel noisy simulated annealing would have a probability of returning a bad solution that would decrease in the power of the number of runs. Nevertheless this benefit should be traded with an additional selection cost. Indeed, if we obtain $K$ solutions retrieved by $K$ parallel NSA realizations, we still face the problem of selecting the best one. We only access estimates of the costs associated to each solution.

**Sketch of the proof** The proof of this theorem is divided into three parts. First, in Section 5.3, we compute the infinitesimal generator of the classical (5.12) and noisy simulated annealing (5.7). Second, in Section 5.4, we compare them (Lemma 1) and third, in Section 5.5, we conclude about the convergence using the Grönwall Lemma (5.19) and the convergence of the classical simulated annealing (5.25).

**Convergence rate** In the case $m^\star b < 1$ a finer bound can be deduced from Grönwall's Lemma and one can obtain a more precise convergence rate for the algorithm (see Theorem 10), which is roughly of the order of:

$$\mathbb{P}(\widetilde{X}_t \in {}^{\mathsf{c}}\chi_\epsilon) \leq \Gamma\ t^{((m^\star-\epsilon)b-\min(1,\alpha/2))/2}$$

where $\Gamma$ is some constant detailed in Theorem 10. In particular this implies that for fixed $\epsilon, \delta > 0$ we can find $T^\star$ such that

$$\mathbb{P}(\widetilde{X}^\star_T \in {}^{\mathsf{c}}\chi_\epsilon) \leq \delta.$$

This leads to a bound (Lemma 3) on the computational complexity, $\mathbb{E}\left(N^{T*}_{call}\right)$, of the order of:

$$\mathbb{E}\left(N^{T*}_{call}\right) \leq \left(\frac{\Gamma}{\delta}\right)^{2(\alpha+1)/(\min(1,\alpha/2)-(m^\star-\epsilon)b)}.$$

## 5.3 Proof, Part 1: Infinitesimal generator

In this section we use the semi-group characterization of the generator in order to prove that as soon as $\widetilde{q}_{\beta_t}$ defined in Equation (5.3) is continuous with respect to $t$ then the infinitesimal generator $\widetilde{L}_{\beta_t}$ of the Markov process $\widetilde{X}_t$ can be written as:

$$\widetilde{L}_{\beta_t} f(x) = \sum_{y \in E} \Big( f(y) - f(x) \Big) \widetilde{q}_{\beta_t}(x,y). \tag{5.7}$$

We briefly recall the definition of the semi-group associated to a Markov process.

**Definition 1.** *The semi-group $(P_{t,t+s})_{t\geq 0, s\geq 0}$ associated to the Markov process $(X_t)_{t\geq 0}$ is a family of probability kernels such that for all non-negative borelian functions:*

$$\forall t, s \in \mathbb{R}_+ \quad P_{t,t+s} f(x) = \mathbb{E}(f(X_{t+s})|X_t = x)$$

Let $(P_{t,t+s})_{t\geq 0, s\geq 0}$ be the semi-group associated to the Markov process $(X_t)_{t\geq 0}$. The semi-group characterization of its generator is given in the following definition:

**Definition 2.** *The infinitesimal generator $L_t$ of the Markov process $(X_t)_{t\geq 0}$ is the operator defined on the set of bounded function $f$ as:*

$$L_t f(x) = \lim_{s \to 0} \frac{P_{t,t+s} f(x) - P_{t,t} f(x)}{s}$$

We start by computing the infinitesimal generator $L_{\beta_t}$ of the process associated to the SA algorithm, *i.e.*, with no measurement noise, and then deduce the infinitesimal generator of the NSA algorithm. Using similar notations to the ones of Section 5.2.3, we consider the noise free inhomogeneous Markov process, $(X_t)_{t\geq 0}$ constructed from the inhomogeneous Markov chain $(\chi_k)_{k\in\mathbb{N}}$ whose one step transition probability is:

$$\forall x, y \in E, \quad q_{\beta_{T_k}}(x,y) = \begin{cases} q_0(x,y) e^{-\beta_{T_k}[J(y)-J(x)]_+} & \text{if } y \neq x \\ 1 - \sum\limits_{z \in E \backslash \{x\}} q_{\beta_{T_k}}(x,z) & \text{if y=x} \end{cases}$$

This is the natural extension of the simulated annealing process with discrete jumping times [73] to the continuous time process. In this configuration, the jumping times are drawn from an i.i.d. sequence of exponential random variables of parameter 1. In the homogeneous configuration, corresponding to $\beta_t = \beta$, the infinitesimal generator has a classical form: $L_\beta = Q_\beta - Id$ where $Q_\beta$ is the transition matrix associated to $q_\beta$ and $Id$ denotes the identity. The extension to the generator of the non-homogeneous process is not straightforward. Therefore we propose to detail the computations.

By definition, for any bounded function $f$:

$$\begin{aligned} L_{\beta_t} f(x) &= \lim_{s\to 0} \frac{P_{t,t+s} f(x) - P_{t,t} f(x)}{s} \\ &= \lim_{s\to 0} \frac{\sum\limits_{y\in E} f(y) \mathbb{P}(X_{t+s} = y | X_t = x) - f(x)}{s} \\ &= \lim_{s\to 0} \frac{\sum\limits_{y\in E} f(y) \mathbb{P}(X_{t+s} = y, H_{t+s} - H_t \geq 0 | X_t = x) - f(x)}{s}, \end{aligned} \tag{5.8}$$

where $H_t = \max\{k \in \mathbb{N} : T_k < t\}$ denotes the number of jumps before time t. Since $T_k$ is a sum of independent exponential variables of parameter 1, one can remark that $H_t$ is in fact a Poisson process of parameter 1.

In order to compute the above limit, we begin by calculating a more explicit form of the probabilities above. We can divide these computations into three parts according to the number of jumps between $t$ and $t+s$:

$$\begin{aligned}\mathbb{P}(X_{t+s} = y, H_{t+s} - H_t \geq 0 | X_t = x) = &\mathbb{P}(X_{t+s} = y, H_{t+s} - H_t = 0 | X_t = x) \\ &+ \mathbb{P}(X_{t+s} = y, H_{t+s} - H_t = 1 | X_t = x) \\ &+ \mathbb{P}(X_{t+s} = y, H_{t+s} - H_t \geq 2 | X_t = x).\end{aligned} \tag{5.9}$$

The first case is straightforward, if there is no jump between $t$ and $t+s$, the process will not change its position and we thus have:

$$\mathbb{P}(X_{t+s} = y,\ H_{t+s} - H_t = 0 | X_t = x) = \delta_x(y) e^{-s}.$$

The second case is slightly more involved. Using the stationarity and the definition of Poisson processes, the event that the algorithm goes from $x$ to $y$, having only one jump between $t$ and $t+s$, can be written as:

$$\mathbb{P}(X_{t+s} = y, H_{t+s} - H_t = 1 | X_t = x) = \mathbb{P}(X_{t+s} = y, \xi_1' < s, s - \xi_1' < \xi_2' | X_t = x)$$

where $\xi_1'$ and $\xi_2'$ are two independent exponential random variables of parameter one.

Let $\xi = (\xi_1', \xi_2')$ and $D_s = \{(h_1, h_2) \in \mathbb{R}^2 | h_1 < s \text{ and } h_2 > s - h_1\}$. In what follows, for a random variable $Y$ we denote $f_Y$ its probability distribution. Using these notations and the fact that $\xi$ is independent of $X_t$, we can write:

$$\begin{aligned}\mathbb{P}(X_{t+s} = y, \xi \in D_s | X_t = x) &= \int_{D_s} f_{(X_{t+s},\xi)|X_t=x}(y, h) \mathrm{d}h \\ &= \int_{D_s} f_{X_{t+s}|\xi=h, X_t=x}(y) f_\xi(h) \mathrm{d}h \\ &= \int_0^s \int_{s-h_1}^{+\infty} q_{\beta_{t+h_1}}(x, y) e^{-h_1} e^{-h_2} \mathrm{d}h_1 \mathrm{d}h_2\end{aligned}$$

The previous equality yields:

$$\mathbb{P}(X_{t+s} = y,\ H_{t+s} - H_t = 1 | X_t = x) = e^{-s} \int_0^s q_{\beta_{t+h_1}}(x, y) \mathrm{d}h_1. \tag{5.10}$$

In the following we use the classical $\mathcal{O}(.)$ and $o(.)$ notations: for all functions $f$ and $g$ defined on some subset of $\mathbb{R}$,

- $f(x) = \mathcal{O}(g(x))$ as $x \to 0^+ \iff \exists\ \sigma,\ x_0 > 0,\ |f(x)| \leq\ \sigma |g(x)|$ for all $0 < x \leq x_0$
- $f(x) = o(g(x))$ as $x \to 0^+ \iff \lim\limits_{x \to 0^+} \frac{f(x)}{g(x)} = 0$.

As for the third case, corresponding to the last term of Equation (5.9), we can see that:

$$\mathbb{P}(X_{t+s} = y,\ H_{t+s} - H_t \geq 2 | X_t = x) \leq \mathbb{P}(H_{t+s} - H_t \geq 2) \leq \mathbb{P}(H_s \geq 2)$$

Since $H_s$ is a Poisson Process of parameter 1, one can check that for all $s$ close to zero we have that:

$$\mathbb{P}(H_s \geq 2) = 1 - \mathbb{P}(H_s = 0) - \mathbb{P}(H_s = 1) = 1 - e^{-s} - s e^{-s} = \mathcal{O}(s^2).$$

This implies that when $s$ is close to zero, the probability that the process goes from $x$ to $y$ between $t$ and $t+s$, with more than one jump is small in comparison to $s$:

$$\mathbb{P}(X_{t+s} = y,\ H_{t+s} - H_t \geq 2 | X_t = x) = \mathcal{O}(s^2) \tag{5.11}$$

Putting all the terms together and replacing them in Equation (2), we can rewrite the infinitesimal generator as follows:

$$\begin{aligned} L_{\beta_t} f(x) = \lim_{s \to 0} \frac{1}{s} & \left[ \sum_{y \in E} f(y) \left[ \delta_x(y) e^{-s} + e^{-s} \int_0^s q_{\beta(t+\tau)}(x,y) d\tau \right] - f(x) \right] \\ & + \lim_{s \to 0} \frac{1}{s} \sum_{y \in E} f(y) \mathbb{P}(X_{t+s} = y, H_{t+s} - H_t \geq 2 | X_t = x) \end{aligned}$$

Using the fact that $f$ is bounded, $E$ finite and the upper bound given by Equation (5.11), one can easily check that the second term is zero. Hence we obtain:

$$\begin{aligned} L_{\beta_t} f(x) &= \lim_{s \to 0} \frac{1}{s} \left[ e^{-s} \left[ f(x) + \sum_{y \in E} f(y) \int_0^s q_{\beta(t+\tau)}(x,y) d\tau \right] - f(x) \right] \\ &= \lim_{s \to 0} \frac{f(x)(e^{-s} - 1)}{s} + \lim_{s \to 0} \frac{e^{-s}}{s} \left( \sum_{y \in E} f(y) \int_0^s q_{\beta(t+\tau)}(x,y) d\tau \right) \end{aligned}$$

Noting the fact that $q_{\beta_t}$ is continuous with respect to $t$ and the following identity

$$e^{-s} = 1 - s + \mathcal{O}(s^2),$$

we easily obtain the simpler form for the infinitesimal generator of the inhomogeneous Markov chain:

$$L_{\beta_t} f(x) = \sum_{y \in E} \Big( f(y) - f(x) \Big) q_{\beta_t}(x,y). \tag{5.12}$$

We can remark that the explicit form of the transition probability $q_{\beta_t}$ does not appear in the proof, hence the result is completely general. The only necessary property of this transition probability is its continuity with respect to $t$.

The fact that $n_t$ and $\beta_t$ are continuous functions ensures the continuity of transition probability $\tilde{q}_{\beta_t}$, defined in Equation (5.3). Therefore, following the same argument, one can deduce (5.7). Here we can see the relevance of the randomness of $N_t$. An increasing deterministic sequence would generate a discontinuous $\tilde{q}_{\beta_t}$ and would make difficult the use of derivations above.

## 5.4 Proof, Part 2: Generators comparison

The fact that for a temperature schedule that decreases slowly enough, the process generated by the classical Simulated Annealing converges to the set of global minima of $J$ is well known. The Noisy Simulated Annealing is a similar algorithm, built on the same principles except that the values of the function $J$ are replaced by an estimation each time its computation is needed. Therefore a tight relation exists between both approaches. Furthermore, as we will show in this section, for a well chosen couple $(\beta_t, n_t)$ the generators of the two algorithms will be 'close' at large times. This is a key element of the proof as it will imply a first condition for the ratio $\beta_t/n_t$.

Using the relations given by Equation (5.12) and Equation (5.7), the quantity of interest is:

$$\widetilde{L}_{\beta_t} f(x) = L_{\beta_t} f(x) + \sum_{y \in E} (f(y) - f(x))(\widetilde{q_{\beta_t}} - q_{\beta_t})(x, y).$$

Hence quantifying the difference between the two generators can be reduced to bounding the difference between the two probability transitions $q_{\beta_t}$ and $\widetilde{q}_{\beta_t}$. Thus the main result of this section is the following lemma.

**Lemma 1.** *Let $\beta_t/\sqrt{n_t} \underset{\infty}{\rightarrow} 0$. There exist two functions $\epsilon_t^-$ and $\epsilon_t^+$ such that*

$$\forall\ t\ \in\ \mathbb{R}_+,\ \forall\ x\ \in\ E',\ \forall\ y\ \in\ E, \qquad \epsilon_t^- q_{\beta_t}(x, y) \leq (\widetilde{q}_{\beta_t} - q_{\beta_t})(x, y) \leq \epsilon_t^+ q_{\beta_t}(x, y)$$

*and*

$$\lim_{t \to +\infty} \epsilon_t^- = \lim_{t \to +\infty} \epsilon_t^+ = 0.$$

Before going into the proof of this lemma, we present some preliminaries. First it can be noticed that for all $x, y \in E$, $x \neq y$ we have:

$$\begin{aligned}
&(\widetilde{q_{\beta_t}} - q_{\beta_t})(x, y) \\
&= q_{\beta_t}(x, y) \left( \frac{\widetilde{q_{\beta_t}}}{q_{\beta_t}} - 1 \right)(x, y) \\
&= q_{\beta_t}(x, y) \left( \mathbb{E}_{N_t} \mathbb{E}_{\omega_1, \ldots, \omega_{N_t}} \left( \frac{e^{-\frac{\beta}{N_t} \lfloor \sum_{i=1}^{N_t} U(y, \omega_i) - U(x, \omega_i) \rfloor_+}}{e^{-\beta_t \lfloor \mathbb{E}(U(y, \Omega)) - \mathbb{E}(U(x, \Omega)) \rfloor_+}} - 1 \right) \right).
\end{aligned}$$

Unless specified otherwise, in this section we always consider $x \neq y$ . The case $x = y$ is handled at the end of the section. To simplify the notations we denote $X_i^{x,y} := U(y, \omega_i) - U(x, \omega_i) - \mathbb{E}(U(y, \omega_i) - U(x, \omega_i))$ and $K^{x,y} = \mathbb{E}(U(y, \omega_i) - U(x, \omega_i))$. Hence,

$$\begin{aligned}
&(\widetilde{q_{\beta_t}} - q_{\beta_t})(x, y) \\
&= q_{\beta_t}(x, y) \left( \mathbb{E}_{N_t} \mathbb{E}_{\omega_1, \ldots, \omega_{N_t}} \left( e^{-\beta_t \left( \lfloor \frac{1}{N_t} \sum_{i=1}^{N_t} X_i^{x,y} + K^{x,y} \rfloor_+ + \lfloor K^{x,y} \rfloor_+ \right)} - 1 \right) \right).
\end{aligned}$$

Noticing that,

$$\forall\ a, b \in \mathbb{R}, \quad -|a| \leq -\lfloor a + b \rfloor_+ + \lfloor b \rfloor_+ \leq |a|,$$

we obtain the following bounds for $(\widetilde{q_{\beta_t}} - q_{\beta_t})(x, y)$:

$$\mathbb{E}_{N_t} \mathbb{E}_{\omega_1, \ldots, \omega_{N_t}} \left( e^{|\frac{\beta_t}{N_t} \sum_{i=1}^{N_t} X_i^{x,y}|} - 1 \right) \geq (\frac{\widetilde{q_{\beta_t}} - q_{\beta_t}}{q_{\beta_t}})(x, y) \geq \mathbb{E}_{N_t} \mathbb{E}_{\omega_1, \ldots, \omega_{N_t}} \left( e^{-|\frac{\beta_t}{N_t} \sum_{i=1}^{N_t} X_i^{x,y}|} - 1 \right). \tag{5.13}$$

In order to obtain a bound for the expectation of a function of $N_t$ we need an estimation of the probability that $N_t$ takes values 'far' from its expectation.

**Lemma 2.** *There exist $\delta \in (0, 1)$ and $a = |(1 - \delta)(1 - \log(1 - \delta)) - 1|$ such that for all $t > 0$ we have:*

$$\mathbb{P}(N_t \leq (1 - \delta) n_t) \leq e^{-a n_t}.$$

**Proof.** We remind the reader that, as mentioned in the definition $N_{t_k}$, at a fixed time $t$ the process $N_t$ can be written as $1+H$, where $H$ is a Poisson random variable of parameter $n_t$.

Fix $t \in \mathbb{R}$ and $\delta \in (0,1)$. We provide an upper bound for $\mathbb{P}(N_t \leq (1-\delta)n_t)$ using the Cramer-Chernoff method.

First we see that for any $\lambda > 0$, applying Markov's inequality we have:

$$\mathbb{P}(N_t \leq (1-\delta)n_t) = \mathbb{P}(e^{-\lambda N_t} > e^{\lambda(\delta-1)n_t}) \leq \frac{\mathbb{E}_{N_t}[e^{-\lambda N_t}]}{e^{\lambda(\delta-1)n_t}}..$$

For $t$ fixed $N_t - 1$ has the distribution of a Poisson random variable of parameter $n_t$. Therefore by direct computations we have:

$$\begin{aligned}\mathbb{E}_{N_t}[e^{-\lambda N_t}] &= \sum_{k>0}(e^{-\lambda(k+1)}e^{-n_t}\frac{n_t^k}{k!}) \\ &= e^{-n_t-\lambda}\sum_{k>0}\frac{(e^{-\lambda}n_t)^k}{k!} \\ &= e^{-n_t-\lambda}e^{e^{-\lambda}n_t}\end{aligned}$$

Putting all these elements together yields:

$$\mathbb{P}(N_t \leq (1-\delta)n_t) \leq \exp([-\lambda(\delta-1)-1+e^{-\lambda}]n_t)$$

The idea is to choose $\lambda$ and $\delta$ in order to obtain the smallest possible value for $-\lambda(\delta-1)-1+e^{-\lambda}$. For $\delta \in (0,1)$, the minimum is reached at $\lambda = -\log(1-\delta)$ which is strictly positive. For such $\lambda$ and $\delta$, we denote $a = |(1-\delta)(1-\log(1-\delta))-1|$ and conclude the proof.

■

Now we have what we need in order to start the proof of Lemma 1.

**Proof of Lemma 1.**

First, recall that if $y \in E\backslash E'$ then $\forall x \in E'$, $\widetilde{q}_{\beta_t}(x,y) = q_{\beta_t}(x,y) = 0$. Hence Lemma 1 is trivially verified for $x \in E$ and $y \in E\backslash E'$.

Considering the inequalities given by Equation (5.13), the proof can be divided in two parts by studying separately the upper bound $\mathbb{E}_{N_t}\mathbb{E}_{\omega_1,\ldots,\omega_{N_t}}\left(e^{|\frac{\beta_t}{N_t}\sum_{i=1}^{N_t}X_i^{x,y}|}\right)$ and the lower bound $\mathbb{E}_{N_t}\mathbb{E}_{\omega_1,\ldots,\omega_{N_t}}\left(e^{-|\frac{\beta_t}{N_t}\sum_{i=1}^{N_t}X_i^{x,y}|}\right)$.

**Upper bound** We have:

$$(\widetilde{q_{\beta_t}} - q_{\beta_t})(x,y) \leq q_{\beta_t}(x,y)\left(\mathbb{E}_{N_t}\mathbb{E}_{\omega_1,\ldots,\omega_{N_t}}\left(e^{|\frac{\beta_t}{N_t}\sum_{i=1}^{N_t}X_i^{x,y}|}-1\right)\right).$$

First we will provide an estimate of $\mathbb{E}_{\omega_1,\ldots,\omega_{N_t}}\left[e^{|\frac{\beta_t}{N_t}\sum_{i=1}^{N_t}X_i^{x,y}|}|N_t\right]$ for all $N_t$.

We start by rewriting this expectation as:

$$\begin{aligned}\mathbb{E}_{\omega_1,\ldots,\omega_{N_t}}\left[e^{|\frac{\beta_t}{N_t}\sum_{i=1}^{N_t}X_i^{x,y}|}|N_t\right] &= \int_{\mathbb{R}_+}\mathbb{P}\left(e^{|\frac{\beta_t}{N_t}\sum_{i=1}^{N_t}X_i^{x,y}|} > u|N_t\right)du \\ &= \int_{\mathbb{R}_+}\mathbb{P}\left(\left|\sum_{i=1}^{N_t}X_i^{x,y}\right| > \frac{\log(u)N_t}{\beta_t}|N_t\right)du.\end{aligned}$$

Since $X_i^{x,y} = U(y,\omega_i) - U(x,\omega_i) - \mathbb{E}_{N_t}\mathbb{E}_{\omega_i}(U(y,\omega_i) - U(x,\omega_i))$ is a centered random variable and $U$ is bounded on $E'$ (see the definition of $U$) there exists $\sigma$ such that $|X_i^{x,y}| \leq \sigma$ (for example set $\sigma = 2M$), almost surely for all $i$. Therefore $(X_i^{x,y})_{1 \leq i \leq N_t}$ are sub-gaussian random variables (see for example [36]) with variance factor $\sigma^2$:

$$\forall u \geq 0,\ \max\left(\mathbb{P}(X_i^{x,y} > u), \mathbb{P}(-X_i^{x,y} > u)\right) \leq e^{-\frac{u^2}{2\sigma^2}}.$$

Considering that $(X_i^{x,y})_{i \leq N_t}$ is a sequence of independent sub-Gaussian variables, their sum is still a sub-Gaussian variable. As $\mathrm{Var}(X_i^{x,y}) \leq \sigma^2$ for all $i$ we have that $\mathrm{Var}\left(\sum_{i=1}^{N_t} X_i^{x,y}\right) \leq N_t\sigma^2$ and therefore:

$$\forall u \geq 0,\ \max\left(\mathbb{P}\left(\sum_{i=1}^{N_t} X_i^{x,y} > u\right), \mathbb{P}\left(-\sum_{i=1}^{N_t} X_i^{x,y} > u\right)\right) \leq e^{-\frac{u^2}{2\sigma^2 N_t}}.$$

For more details about sub-gaussian variables we refer to [36]. We use this property of concentration in order to get an estimate of the expectation:

$$\begin{aligned}
&\mathbb{E}_{\omega_1,\ldots,\omega_{N_t}}\left[e^{|\frac{\beta_t}{N_t}\sum_{i=1}^{N_t} X_i^{x,y}|}|N_t\right] \\
&\qquad = \int_0^1 \mathbb{P}\left(\left|\sum_{i=1}^{N_t} X_i^{x,y}\right| > \frac{\log(u)N_t}{\beta_t}|N_t\right)du + \int_1^{+\infty} \mathbb{P}\left(\left|\sum_{i=1}^{N_t} X_i^{x,y}\right| > \frac{\log(u)N_t}{\beta_t}|N_t\right)du \\
&\qquad \leq 1 + 2\int_1^{+\infty} e^{-\frac{1}{2\sigma^2}\left(\frac{\log(u)\sqrt{N_t}}{\beta_t}\right)^2} du.
\end{aligned}$$

Using a simple variable substitution $s = \frac{\log(u)}{\lambda_t}$ with $\lambda_t = \frac{\sigma\beta_t}{\sqrt{N_t}}$, we get:

$$\begin{aligned}
\mathbb{E}_{\omega_1,\ldots,\omega_{N_t}}\left[e^{|\frac{\beta_t}{N_t}\sum_{i=1}^{N_t} X_i^{x,y}|}|N_t\right] &\leq 1 + 2\lambda_t\int_0^{+\infty} e^{\lambda_t s}e^{-\frac{s^2}{2}}du \\
&\leq 1 + 2\lambda_t e^{\frac{\lambda_t^2}{2}}\int_0^{+\infty} e^{-\frac{(s-\lambda_t)^2}{2}}ds \\
&\leq 1 + 2\lambda_t e^{\frac{\lambda_t^2}{2}}\int_{-\lambda_t}^{+\infty} e^{-\frac{u^2}{2}}du \\
&\leq 1 + 2\sqrt{2\pi}\lambda_t e^{\frac{\lambda_t^2}{2}}\mathbb{P}(G > -\lambda_t) \\
&\leq 1 + 2\sqrt{2\pi}\lambda_t e^{\frac{\lambda_t^2}{2}}(1 - \mathbb{P}(G > \lambda_t)).
\end{aligned}$$

where $G$ is a standard Gaussian. Thanks to the Taylor formula, we know that there exists a constant $0 < \theta < 1$, such that:

$$\begin{aligned}
\mathbb{P}(G > \lambda_t) &= \mathbb{P}(G > 0) + \lambda_t\frac{e^{\frac{-(\theta\lambda_t)^2}{2}}}{\sqrt{2\pi}} \\
&= \frac{1}{2} + \lambda_t\frac{e^{\frac{-(\theta\lambda_t)^2}{2}}}{\sqrt{2\pi}} \\
&\geq \frac{1}{2} + \lambda_t\frac{e^{\frac{-(\lambda_t)^2}{2}}}{\sqrt{2\pi}}.
\end{aligned}$$

This leads to:

$$\mathbb{E}_{\omega_1,\ldots,\omega_{N_t}}\left[e^{|\frac{\beta_t}{N_t}\sum_{i=1}^{N_t} X_i^{x,y}|}|N_t\right] \leq 1 + \left(\sqrt{2\pi}e^{\frac{\lambda_t^2}{2}}\right)\lambda_t.$$

Thus, replacing $\lambda_t$ by its definition we see that we need an estimate of:

$$\mathbb{E}_{N_t}\left[1+(\sqrt{2\pi}e^{\frac{\sigma^2\beta_t^2}{2N_t}})\frac{\sigma\beta_t}{\sqrt{N_t}}\right].$$

In order to simplify the notations we denote: $g_t = \sigma\beta_t$. Using the bound given by Lemma 2 we get:

$$\begin{aligned}
&\mathbb{E}_{N_t}\left[1+\sqrt{2\pi}e^{\frac{g_t^2}{2\sqrt{N_t}}}\frac{g_t}{\sqrt{N_t}}\right] \\
&= 1+\mathbb{E}_{N_t}\left[\sqrt{2\pi}e^{\frac{g_t^2}{2\sqrt{N_t}}}\frac{g_t}{\sqrt{N_t}}\right] \\
&\leq 1+\mathbb{E}_{N_t}\left[\sqrt{2\pi}e^{\frac{g_t^2}{2\sqrt{N_t}}}\frac{g_t}{\sqrt{N_t}}(\mathbb{1}_{[1,(1-\delta)n_t]}+\mathbb{1}_{[(1-\delta)n_t,+\infty)})\right] \\
&\leq 1+\sqrt{2\pi}g_t\left[\frac{e^{\frac{g_t^2}{2(1-\delta)n_t}}}{\sqrt{(1-\delta)n_t}}+e^{g_t^2-an_t}\right]
\end{aligned}$$

Finally we have obtained that under assumptions of Section 5.2.2:

$$\begin{aligned}
(\widetilde{q_{\beta_t}}-q_{\beta_t})(x,y) &\leq q_{\beta_t}(x,y)\left(\mathbb{E}_{N_t}\mathbb{E}_{\omega_1,\dots,\omega_{N_t}}\left(e^{|\frac{\beta_t}{N_t}\sum_{i=1}^{N_t}X_i^{x,y}|}-1\right)\right) \\
&\leq q_{\beta_t}(x,y)\sqrt{2\pi}g_t\left[\frac{e^{\frac{g_t^2}{2(1-\delta)n_t}}}{\sqrt{(1-\delta)n_t}}+e^{g_t^2-an_t}\right]
\end{aligned}$$

Hence it is natural to define $\epsilon_t^+$ as:

$$\epsilon_t^+ = \sqrt{2\pi}\beta_t\sigma\left[\frac{e^{\frac{\beta_t^2\sigma^2}{2(1-\delta)n_t}}}{\sqrt{(1-\delta)n_t}}+e^{\beta_t^2\sigma^2-an_t}\right].$$

Since $\beta_t/\sqrt{n_t}\underset{\infty}{\rightarrow}0$ one can check that $\epsilon_t^+$ goes to 0 when $t$ goes to infinity. Here we can see once more the importance of the balance between the two parameters $\beta_t$ and $n_t$.

**Lower bound** Considering the left-hand side of Equation (5.13) we have:

$$(\widetilde{q_{\beta_t}}-q_{\beta_t})(x,y) \geq q_{\beta_t}(x,y)\left(\mathbb{E}_{N_t}\mathbb{E}_{\omega_1,\dots,\omega_{N_t}}\left(e^{-|\frac{\beta_t}{N_t}\sum_{i=1}^{N_t}X_i^{x,y}|}-1\right)\right).$$

In a similar way we start by obtaining a lower bound for $\mathbb{E}_{\omega_1,\dots,\omega_{N_t}}\left(e^{-|\frac{\beta_t}{N_t}\sum_{i=1}^{N_t}X_i^{x,y}|}|N_t\right)$ and after we improve it using the probabilistic properties of $N_t$. First observe that:

$$\begin{aligned}
\mathbb{E}_{\omega_1,\dots,\omega_{N_t}}\left[e^{-|\frac{\beta_t}{N_t}\sum_{i=1}^{N_t}X_i^{x,y}|}|N_t\right] &= \int_{\mathbb{R}_+}\mathbb{P}\left(e^{-|\frac{\beta_t}{N_t}\sum_{i=1}^{N_t}X_i^{x,y}|}>u|N_t\right)du \\
&= \int_{\mathbb{R}_+}\mathbb{P}\left(-\left|\sum_{i=1}^{N_t}X_i^{x,y}\right|>\frac{\log(u)N_t}{\beta_t}|N_t\right)du \\
&= \int_0^1\mathbb{P}\left(\left|\sum_{i=1}^{N_t}X_i^{x,y}\right|<\frac{-\log(u)N_t}{\beta_t}|N_t\right)du \\
&= \int_0^1 1-\mathbb{P}\left(\left|\sum_{i=1}^{N_t}X_i^{x,y}\right|>\frac{-\log(u)N_t}{\beta_t}|N_t\right)du \\
&\geq 1-2\int_0^1 e^{-\frac{N_t}{2\sigma^2}\left(\frac{-\log(u)}{\beta_t}\right)^2}du
\end{aligned}$$

Again, this is due to the fact that the sum of $X_i^{x,y}$ is sub-Gaussian with variance factor $N_t\sigma^2$.

Using the same variable substitution as above: $s = \frac{\log(u)}{\lambda_t}$ with $\lambda_t = \frac{\sigma\beta_t}{\sqrt{N_t}}$, we get:

$$\begin{aligned}\mathbb{E}_{\omega_1,\dots,\omega_{N_t}}\left[e^{-|\frac{\beta_t}{N_t}\sum_{i=1}^{N_t}X_i^{x,y}|}|N_t\right] &\geq 1-2\lambda_t\int_{-\infty}^{0} e^{\lambda_t s}e^{-\frac{s^2}{2}}du\\ &\geq 1-2\lambda_t e^{\frac{\lambda_t^2}{2}}\int_{-\infty}^{0} e^{-\frac{(s-\lambda_t)^2}{2}}ds\\ &\geq 1-2\lambda_t e^{\frac{\lambda_t^2}{2}}\int_{-\infty}^{-\lambda_t} e^{-\frac{u^2}{2}}du\\ &\geq 1-2\sqrt{2\pi}\lambda_t e^{\frac{\lambda_t^2}{2}}\mathbb{P}(G>\lambda_t)\end{aligned}$$

where $G$ is a standard $N(0,1)$ Gaussian. As seen before using the Taylor formula, we can obtain:

$$\mathbb{P}(G>\lambda_t) \geq \frac{1}{2}+\lambda_t\frac{e^{\frac{-(\lambda_t)^2}{2}}}{\sqrt{2\pi}}.$$

This leads to

$$\mathbb{E}_{\omega_1,\dots,\omega_{N_t}}\left[e^{|\frac{\beta_t}{N_t}\sum_{i=1}^{N_t}X_i^{x,y}|}|N_t\right] \geq 1-\left(\sqrt{2\pi}e^{\frac{\lambda_t^2}{2}}\right)\lambda_t.$$

This expression has exactly the symmetric form to the one obtained in the upper bound part. Thus, the lower bound is obtained the same way as the upper bound. We directly get:

$$\begin{aligned}&\mathbb{E}_{N_t}\left[1-(\sqrt{2\pi}e^{\frac{g_t^2}{2N_t}})\frac{g_t}{\sqrt{N_t}}\right]\\ &\geq 1-\sqrt{2\pi}g_t\left[\frac{e^{\frac{g_t^2}{2(1-\delta)n_t}}}{\sqrt{(1-\delta)n_t}}+e^{g_t^2-an_t}\right]\end{aligned}$$

Now we define $\epsilon_t^-$:

$$\epsilon_t^- = -\sqrt{2\pi}\beta_t\sigma\left[\frac{e^{\frac{\beta_t^2\sigma^2}{2(1-\delta)n_t}}}{\sqrt{(1-\delta)n_t}}+e^{\beta_t^2\sigma^2-an_t}\right].$$

It is easy to see that $\epsilon_t^-$ goes to 0 when $t$ goes to infinity as soon as $\beta_t/\sqrt{n_t} \underset{\infty}{\to} 0$. This completes the proof of Lemma 1. ■

## 5.5 Proof, Part 3: Rate of convergence in the general case

We first complete the proof of convergence as stated in Theorem 8 and then deduce the convergence rate (Theorem 10) from it. This enables us to provide an upper bound on the minimal number of cost function evaluations in Section 5.5.3.

### 5.5.1 Proof of Theorem 8

The proof of Theorem 8 follows the roadmap of Holley and Strook [77] and relies on the use of the Grönwall lemma. We derive a differential inequality for the $L^2_{\mu_{\beta_t}}$-norm of the density measure of the NSA process with respect to $\mu_{\beta_t}$ and deduce an integrated version of it using the lemma. We then show that bounding the $L^2_{\mu_{\beta_t}}$-norm of this density implies the convergence of the process to the optimal state space $\chi_\epsilon$.

**Proof.** **[Proof of Theorem 8]** Our goal is to show that when $t$ goes to infinity, the noisy simulated annealing gets "close enough" to the classical simulated annealing. Therefore we denote by $f_t$ the Radon-Nikodym derivative of the probability density of the noisy simulated annealing process $\widetilde{X}_t$ with respect to the Gibbs measure $\mu_{\beta_t}$:

$$f_t = \frac{\mathrm{d}m_t}{\mathrm{d}\mu_{\beta_t}} \tag{5.14}$$

where $m_t$ is the distribution of $(\widetilde{X}_s)_{s\geq 0}$ at time $t$. A first remark is that $f_t(x) = 0$ for all $t \geq 0$ and all $x \in E \setminus E'$, since our process, by construction does not accept states out of $E'$.
Using the results obtained in Section 5.3 one can see that $\mathbb{R}_+ \ni t \to \widetilde{L}_{\beta_t}$ is continuous and therefore the semi-group $(P_{s,t})_{0\leq s\leq t}$ is smooth . Also by their definition the operators $(P_{s,t})_{0\leq s\leq t}$ are linear and have the following semi-group property: $P_{s,t+h} = P_{s,t} \circ P_{t,t+h}$, for all $0 \leq s < t$ and $h > 0$. Hence, for all $0 \leq s \leq t$, we have:

$$\frac{\mathrm{d}}{\mathrm{d}t} P_{s,t} = P_{s,t} L_t. \tag{5.15}$$

For details about the infinitesimal generator see Section 1.4. of [21].

As shown in (5.24), bounding the $L^2$-norm of $f_t$ w.r.t. $\mu_{\beta_t}$, *i.e.*, $\|f_t\|_{\mu_{\beta_t}}$, ensures convergence of the NSA algorithm. However it does not provide enough information about the convergence of $m_t$ towards $\mu_{\beta_t}$ to deduce a fine convergence rate. This is why we study the evolution of $\|f_t - 1\|_{\mu_{\beta_t}}$ which controls the distance between the two measures. If this quantity is bounded then we obtain the convergence of the NSA algorithm. If moreover it converges to zero, it implies a stronger convergence rate. In order to prove that, we deduce a differential inequality for $\|f_t - 1\|^2_{\mu_{\beta_t}}$. We start by computing its derivative:

$$\begin{aligned}\partial_t \|f_t - 1\|^2_{\mu_{\beta_t}} = \partial_t \|f_t\|^2_{\mu_{\beta_t}} =& \partial_t \sum_{x\in E} f_t^2(x)\mu_{\beta_t}(x) \\ &= 2\sum_{x\in E} f_t(x)\partial_t \left[\frac{m_t}{\mu_{\beta_t}}\right](x)\mu_{\beta_t}(x) + \sum_{x\in E} f_t(x)\partial_t\mu_{\beta_t}(x).\end{aligned}$$

Using the backward Kolmogorov equation given by Equation (5.15), for the first term we have:

$$\begin{aligned}\sum_{x\in E} f_t(x)\partial_t \left[\frac{m_t}{\mu_{\beta_t}}\right](x)\mu_{\beta_t}(x) &= \sum_{x\in E} f_t(x)\partial_t m_t(x) - \sum_{x\in E} f_t(x)\frac{m_t}{\mu_{\beta_t}}(x)\partial_t\mu_{\beta_t}(x) \\ &= \sum_{x\in E}\left[\widetilde{L}_{\beta_t} f_t(x)\right] m_t(x) - \sum_{x\in E} f_t(x)\frac{m_t}{\mu_{\beta_t}}(x)\partial_t\mu_{\beta_t}(x).\end{aligned} \tag{5.16}$$

Denote $\langle J\rangle_{\mu_{\beta_t}} := \int J d\mu_{\beta_t}$ the mean of $J$ with respect to $\mu_{\beta_t}$. One can check that:

$$\partial_t\mu_{\beta_t}(x) = -\beta_t'\left[J(x) - \langle J\rangle_{\mu_{\beta_t}}\right]\mu_{\beta_t}(x).$$

Thus, we easily obtain the following equality:

$$\partial_t \|f_t - 1\|^2_{\mu_{\beta_t}} = 2\sum_{x\in E} f_t(x)(\widetilde{L}_{\beta_t} f_t)(x)\mu_{\beta_t}(x) + \beta_t' \sum_{x\in E}(J(x) - \langle J\rangle_{\mu_{\beta_t}}) f_t^2(x)\mu_{\beta_t}. \tag{5.17}$$

First, we focus on the first term of the right hand side of Equation (5.17). Since we try to control the generator of the noisy simulated annealing by the generator of the classical one, it is natural to write $\widetilde{L}_{\beta_t}$ as $L_{\beta_t} + \widetilde{L}_{\beta_t} - L_{\beta_t}$. This comparison leads to the computation:

$$\begin{aligned}
&\sum_{x\in E} f_t(x)(\widetilde{L}_{\beta_t} f_t)(x)\mu_{\beta_t}(x)\\
&= \sum_{x} f_t(x)(L_{\beta_t} f_t)(x)\mu_{\beta_t}(x) + \sum_{x\in E} f_t(x)\left(\sum_{y\in E}(f_t(y) - f_t(x))(\widetilde{q_{\beta_t}} - q_{\beta_t})(x,y)\right)\mu_{\beta_t}(x).
\end{aligned}$$

We rewrite the last part of the second term using Lemma 1:

$$\begin{aligned}
&\sum_{y\in E}(f_t(y) - f_t(x))(\widetilde{q_{\beta_t}} - q_{\beta_t})(x,y)\\
&= \sum_{y\in E} f_t(y)(\widetilde{q_{\beta_t}} - q_{\beta_t})(x,y) - \sum_{y\in E} f_t(x))(\widetilde{q_{\beta_t}} - q_{\beta_t})(x,y)\\
&\leq \epsilon_t^+ \sum_{y\in E} f_t(y) q_{\beta_t}(x,y) - \epsilon_t^+ \sum_{y\in E} f_t(x) q_{\beta_t}(x,y) + \epsilon_t^+ \sum_{y\in E} f_t(x) q_{\beta_t}(x,y) - \epsilon_t^- \sum_{y\in E} f_t(x) q_{\beta_t}(x,y)\\
&\leq \epsilon_t^+ \sum_{y\in E}(f_t(y) - f_t(x)) q_{\beta_t}(x,y) + (\epsilon_t^+ - \epsilon_t^-) f_t(x)\\
&\leq \epsilon_t^+ L_{\beta_t} f_t(x) + (\epsilon_t^+ - \epsilon_t^-) f_t(x).
\end{aligned}$$

Inserting this in the previous inequality, we get:

$$\begin{aligned}
&\sum_{x\in E} f_t(x)(\widetilde{L}_{\beta_t} f_t)(x)\mu_{\beta_t}(x)\\
&\leq (1 + \epsilon_t^+)\sum_{x} f_t(x)(L_{\beta_t} f_t)(x)\mu_{\beta_t}(x) + (\epsilon_t^+ - \epsilon_t^-)\sum_{x\in E} f_t^2(x)\mu_{\beta_t}(x)\\
&\leq (1 + \epsilon_t^+)\sum_{x} f_t(x)(L_{\beta_t} f_t)(x)\mu_{\beta_t}(x) + (\epsilon_t^+ - \epsilon_t^-)\sum_{x\in E}(f_t^2(x) - 1)\mu_{\beta_t}(x) + (\epsilon_t^+ - \epsilon_t^-).
\end{aligned}$$

Therefore, using Equation (5.17), we obtain the following inequality:

$$\begin{aligned}
&\frac{d}{dt}\|f_t - 1\|^2_{\mu_{\beta_t}}\\
&\leq 2\left[(1 + \epsilon_t^+)\sum_{x} f_t(x)(L_{\beta_t} f_t)(x)\mu_{\beta_t}(x) + (\epsilon_t^+ - \epsilon_t^-)\sum_{x\in E}(f_t^2(x) - 1)\mu_{\beta_t}(x) + (\epsilon_t^+ - \epsilon_t^-)\right]\\
&\qquad + \beta_t' \sum_{x\in E}(J(x) - \langle J\rangle_{\mu_{\beta_t}}) f_t^2(x)\mu_{\beta_t}\\
&\leq 2\left[(1 + \epsilon_t^+)\sum_{x} f_t(x)(L_{\beta_t} f_t)(x)\mu_{\beta_t}(x) + (\epsilon_t^+ - \epsilon_t^-)\sum_{x\in E}(f_t^2(x) - 1)\mu_{\beta_t}(x) + (\epsilon_t^+ - \epsilon_t^-)\right]\\
&\qquad + \beta_t' \sum_{x\in E}(J(x) - \langle J\rangle_{\mu_{\beta_t}})(f_t(x) - 1)^2\mu_{\beta_t}(x) + 2\beta_t' \sum_{x\in E}(J(x) - \langle J\rangle_{\mu_{\beta_t}})(f_t(x) - 1)\mu_{\beta_t(x)}.
\end{aligned}$$

In order to deal with the first sum we use an estimate of the spectral gap of $L_{\beta_t}$. This is provided by Theorem 2.1 of Holley and Strook[77].

**Theorem 9** (Holley and Strook 88)**.** *Under assumptions of 5.2.2, there exist two positive constants $0 < c \leq C < +\infty$ such that $\forall\ \beta\ \in\ \mathbb{R}_+$,*

$$ce^{-\beta m^\star} \leq \gamma(\beta) \leq Ce^{-\beta m^\star}$$

*where $\gamma(\beta) = \inf\{-\int \phi L_\beta \phi\ d\mu_\beta : \|\phi\|_{\mu_\beta} = 1$ and $\int \phi d\mu_\beta = 0\}$ and $m^\star$ is the maximum depth of a well containing a local minimum defined in Equation(5.4).* ■

**Remark** The constant $m^\star$ is always strictly positive as soon as the function has a strict local minimum that is not global. This is generally the case in our setting. Also, we always have $m^\star \leq M$.

**Following of the proof.** Using the definition of $f_t$, one can see that $\int f_t\ d\mu_{\beta_t} = 1$, hence applying the theorem for $\phi = \dfrac{f_t - 1}{\|f_t - 1\|_{\mu_{\beta_t}}}$ gives:

$$-\sum_x \phi(L_{\beta_t}\phi)(x)\mu_{\beta_t}(x) \geq ce^{-\beta_t m^\star}.$$

This and the definition of $L_{\beta_t}$ imply:

$$\sum_x f_t(x)(L_{\beta_t} f_t)(x)\mu_{\beta_t}(x) \leq -ce^{-\beta_t m^\star}\|f_t - 1\|^2_{\mu_{\beta_t}}.$$

$J$ is a positive function bounded by $M$ on $E'$. For all $x \in E \setminus E'$, the only points where $J > M$, we have that $J(x) = +\infty$ and therefore $\mu_{\beta_t}(x) = 0$. This implies that for all measurable functions $g$,

$$\sum_{x\in E}(J(x) - \langle J\rangle_{\mu_{\beta_t}})g(x)\mu_{\beta_t} \leq M\|g\|_{\mu_{\beta_t}}.$$

Putting all these terms together gives:

$$\begin{aligned}\frac{d}{dt}\|f_t - 1\|^2_{\mu_{\beta_t}} \leq 2&\left[-ce^{-\beta_t m^\star}(1+\epsilon_t^+) + (\epsilon_t^+ - \epsilon_t^-) + \frac{M}{2}\beta_t'\right]\|f_t - 1\|^2_{\mu_{\beta_t}} \\ &+ 2M\beta_t'\|f_t - 1\|_{\mu_{\beta_t}} \\ &+ 2(\epsilon_t^+ - \epsilon_t^-).\end{aligned} \tag{5.18}$$

We denote $u_t = \|f_t - 1\|^2_{\mu_{\beta_t}}$. Considering the fact that $\epsilon_t^+$ is a positive function we have:

$$\begin{aligned}u_t' \leq 2&\left[-ce^{-\beta_t m^\star} + (\epsilon_t^+ - \epsilon_t^-) + \frac{M}{2}\beta_t'\right]u_t \\ &+ 2M\beta_t'\sqrt{u_t} \\ &+ 2(\epsilon_t^+ - \epsilon_t^-).\end{aligned} \tag{5.19}$$

Using that, $\forall x \in \mathbb{R}$, $\frac{1}{4}x^2 + 1 \geq x$, we get:

$$\begin{aligned}u_t' \leq 2&\left[-ce^{-\beta_t m^\star} + (\epsilon_t^+ - \epsilon_t^-) + \left(\frac{M}{2} + \frac{M}{4}\right)\beta_t'\right]u_t \\ &+ 2M\beta_t' + 2(\epsilon_t^+ - \epsilon_t^-).\end{aligned} \tag{5.20}$$

Let $A_t = 2ce^{-\beta_t m^\star}$ and $B_t = 2(\epsilon_t^+ - \epsilon_t^-) + 2M\beta_t'$.

Applying Grönwall's Lemma for the previous relation gives:

$$u_t \leq u_0 e^{\int_0^t -A_s + B_s \mathrm{d}s} + \int_0^t B_s e^{\int_s^t -A_h + B_h \mathrm{d}h}\mathrm{d}s. \tag{5.21}$$

Under Assumptions 5.2.2, there exist $b, d > 0$ such that $\beta_t = b \log(1 + td)$. This implies:

$$\beta_t' = \frac{bd}{1+td} \text{ and } e^{-m^\star \beta_t} = \left(\frac{1}{1+td}\right)^{m^\star b}.$$

Using the definition of $\epsilon_t^+, \epsilon_t^-$ and the fact that $n_t = (1+t)^\alpha$ one can check that:

$$A_t = \mathcal{O}\left(\frac{1}{t^{m^\star b}}\right) \text{ and } B_t = \mathcal{O}\left(\frac{1}{t} \vee \frac{\log t}{t^{\alpha/2}}\right)$$

We can see that if $B_t = o(A_t)$ the second term of (5.21) is bounded and gives us a finite upper bound on $u_t$. This happens as soon as:

$$m^\star b < 1 \wedge \alpha/2 \tag{5.22}$$

The condition given by (5.22) is sufficient yet not necessary. For $\alpha > 2$, $B_t$ becomes of the order $\mathcal{O}(1/t)$ and thus we can choose $d$ in a way that preserves a finite upper bound of (5.21) even for $m^\star b = 1$. One can check by direct computation that this is true for any $d < cm^\star/M$.

Let $\beta_t$ and $n_t$ be chosen in order to comply to one of the two previously mentioned conditions. Then there exists a constant $K'$ such that

$$u_t \leq K' \text{ for all } t \in \mathbb{R}^+ \tag{5.23}$$

To complete the proof of Theorem 8 one can observe that for all $t \in \mathbb{R}_+$, and all $\epsilon > 0$ :

$$\mathbb{P}(\widetilde{X}_t \in {}^{\mathsf{c}}\chi_\epsilon) = \mathbb{E}(\mathbb{1}_{[J^\star+\epsilon,+\infty)}(J(\widetilde{X}_t))).$$

Using the Cauchy-Schwarz inequality and the upper bound given by (5.23) we obtain:

$$\begin{aligned}
\mathbb{E}(\mathbb{1}_{[J^\star+\epsilon,+\infty}(J(\widetilde{X}_t))) &= \int_{\mathbb{R}} \mathbb{1}_{[J^\star+\epsilon,+\infty}(J(x)) f_t d\mu_{\beta_t}(x) \\
&\leq \left(\int_{\mathbb{R}} (f_t)^2 d\mu_{\beta_t}(x)\right)^{\frac{1}{2}} \left(\int_{\mathbb{R}} \mathbb{1}^2_{[J^\star+\epsilon,+\infty)}(J(x)) d\mu_{\beta_t}(x)\right)^{\frac{1}{2}} \\
&\leq \|f_t\|_{L^2_{\mu_{\beta_t}}} (\mu_{\beta_t}({}^{\mathsf{c}}\chi_\epsilon))^{1/2} \\
&\leq K (\mu_{\beta_t}({}^{\mathsf{c}}\chi_\epsilon))^{1/2}
\end{aligned} \tag{5.24}$$

with $K = \sqrt{K'+1}$. This completes the proof of Theorem 8. ■

### 5.5.2 Convergence rate

A first rate of convergence can be deduced from Theorem 8 using the concentration speed of the Gibbs measure on $\chi_\epsilon$.

$$\begin{aligned}
\mu_{\beta_t}({}^{\mathsf{c}}\chi_\epsilon) &= \frac{\sum_{x \in {}^{\mathsf{c}}\chi_\epsilon} e^{-\beta_t J(x)}}{\sum_{x \in E} e^{-\beta_t J(x)}} \\
&= \frac{\sum_{x \in {}^{\mathsf{c}}\chi_\epsilon} e^{-\beta_t J(x)}}{\sum_{x \in {}^{\mathsf{c}}\chi_\epsilon} e^{-\beta_t J(x)} + \sum_{x \in \chi_\epsilon} e^{-\beta_t J(x)}} \\
&\leq \frac{(|E| - |\chi_\epsilon|) e^{-\beta_t (J^\star + \epsilon)}}{0 + |\chi_\epsilon| e^{-\beta_t J^\star}} \\
&\leq \left(\frac{|E|}{|\chi_\epsilon|} - 1\right) (1+td)^{-b\epsilon}
\end{aligned} \tag{5.25}$$

As the dependency of $K$ (Theorem 8) in $b$ and $\alpha$ is not explicit, we can however not deduce an optimal choice of $(b, \alpha)$ from this bound. This can be achieved if we assume that (5.22) holds and distinguish the two cases $\alpha \leq 2$ and $\alpha > 2$. Indeed, we can then improve the bound on $u_t$ and derive a more accurate convergence rate of the algorithm. This rate can then be optimized to obtain either an upper bound of the probability of convergence to $\chi_\epsilon$ for a fixed computational budget or the minimal computational budget at a fixed risk of convergence out of $\chi_\epsilon$.

**Theorem 10.** *Under assumptions of Section 5.2.2, suppose:* $\beta_t = b\log(td+1)$, $n_t = (1+td)^\alpha$ *and* $m^\star b < \min(\alpha/2, 1)$*:*

- *if* $\alpha \geq 2$*, let* $b$ *be such that* $m^\star b < 1$ *and let* $\gamma \in (0, \alpha/2 - m^\star b)$*,*
  *Then, there exist* $\Gamma_\gamma, \Gamma_2 > 0$ *such that for* $t$ *large enough, for all* $\epsilon > 0$,

$$\mathbb{P}(\widetilde{X}_t \in {}^c\chi_\epsilon) \leq \Gamma_\gamma \Gamma_2 (1+td)^{(m^\star b - 1 - b\epsilon)/2} + \Gamma_2 (1+td)^{-b\epsilon}$$

- *if* $\alpha < 2$*, let* $b$ *be such that* $m^\star b < \alpha/2$ *and let* $\gamma \in (0, \alpha/2 - m^\star b)$*,*
  *Then, there exist* $\Gamma_\gamma, \Gamma_2 > 0$ *such that for* $t$ *large enough, for all* $\epsilon > 0$,

$$\mathbb{P}(\widetilde{X}_t \in {}^c\chi_\epsilon) \leq \Gamma_\gamma \Gamma_2 (1+td)^{(m^\star b - \alpha/2 + \gamma - b\epsilon)/2} + \Gamma_2 (1+td)^{-b\epsilon}$$

■

**Remark 1.** $\gamma$ *is not a new parameter of the NSA algorithm. This is a technical element that enables the tuning of the computational complexity bounds of Section 5.5.3. As shown in the Appendix A.1,* $\Gamma_\gamma$ *is of the order of* $1/\gamma$*.*

**Remark 2.** *These two bounds display the trade-off between the convergence rate of the Gibbs measure to the uniform distribution over the global minima and the rate of convergence of the NSA process to the Gibbs measure. For the first bound, considering* $\alpha > 2$ *we recover the classical rate of convergence of the simulated annealing in the noise free case. This corresponds to the result of [72]. The second bound provides the rate of convergence for a choice of* $\alpha < 2$*. It can be seen that* $b$ *will have to be reduced to ensure the convergence and thus this bound exhibits clearly the trade off between cooling and estimation.*

**Proof.** Under assumptions of Theorem 10, the following bound on $u_t$ can be derived from Grönwall's Lemma (for details see Appendix A.1):

$$u_t \leq \begin{cases} \Gamma_\gamma (1+td)^{m^\star b - 1} & \text{if } \alpha \geq 2 \\ \Gamma_\gamma (1+td)^{m^\star b - \alpha/2 + \gamma} & \text{if } \alpha < 2 \end{cases} \tag{5.26}$$

Thus we can compute a new bound on the probability that $\widetilde{X}_t$ does not belong to the optimal set $\chi_\epsilon$ (*cf.* 5.6):

$$\begin{aligned}
\mathbb{P}(\widetilde{X}_t \in {}^c\chi_\epsilon) &= \int_{\mathbb{R}} \mathbb{1}_{{}^c\chi_\epsilon}(J(x)) f_t d\mu_{\beta_t}(x) \\
&= \int_{\mathbb{R}} \mathbb{1}_{{}^c\chi_\epsilon}(J(x))(f_t - 1) d\mu_{\beta_t}(x) + \int_{\mathbb{R}_+} \mathbb{1}_{{}^c\chi_\epsilon}(J(x)) d\mu_{\beta_t}(x) \\
&\leq \left( \int_{\mathbb{R}_+} (f_t - 1)^2 d\mu_{\beta_t}(x) \int_{\mathbb{R}_+} \mathbb{1}^2_{{}^c\chi_\epsilon}(J(x)) d\mu_{\beta_t}(x) \right)^{1/2} + \mu_{\beta_t}({}^c\chi_\epsilon) \\
&\leq \sqrt{u_t \mu_{\beta_t}({}^c\chi_\epsilon)} + \mu_{\beta_t}({}^c\chi_\epsilon)
\end{aligned} \tag{5.27}$$

This means that if there exist $(\alpha, b)$ such that $u_t = \mathcal{O}(\mu_{\beta_t})$ the convergence rate in the noisy case will be of the same order as in the classical one, but for a smaller $b$.

Using the previous inequality, (5.26) and the concentration rate of the Gibbs measure given by (5.25) we have:

$$\mathbb{P}(\widetilde{X}_t \in {}^{\mathsf{c}}\chi_\epsilon) \leq \begin{cases} \Gamma_\gamma \Gamma_2 (1+td)^{\frac{m^\star b - 1 - b\epsilon}{2}} + \Gamma_2 (1+td)^{-b\epsilon} & \text{if } \alpha \geq 2 \\ \Gamma_\gamma \Gamma_2 (1+td)^{\frac{m^\star b - \alpha/2 + \gamma - b\epsilon}{2}} + \Gamma_2 (1+td)^{-b\epsilon} & \text{if } \alpha < 2 \end{cases}$$

where $\Gamma_2 = \frac{|E|}{|\chi_\epsilon|} - 1$. ■

### 5.5.3 Computational complexity of NSA

Given the convergence rate of the algorithm, we can define $T^\star$ such that the confidence inequality constraint is satisfied at time $T^\star$.

Let $N_{call}^T$ be the number of cost function evaluations made by the NSA until time $T$. This is a random variable. We define the computational cost of the algorithm as the expectation of this random variable. It can be written as:

$$\mathbb{E}\left(N_{call}^T\right) = \mathbb{E}\left(\sum_{k \geq 1} \mathbb{1}_{T_k < T} N_{T_k}\right).$$

**Lemma 3.** *Let $\delta, \epsilon > 0$, $\gamma \in (0, \alpha/2 - m^\star b)$ and*

$$T^\star = \frac{1}{d}\left(\max\left(\left(\frac{2\Gamma_\gamma}{\delta}\right)^{2/(\min(1,\frac{\alpha}{2}-\gamma)-m^\star b+b\epsilon)}, \left(\frac{2\Gamma_2}{\delta}\right)^{1/b\epsilon}\right) - 1\right).$$

*Then, for all $t \geq T^\star$, $\mathbb{P}(\widetilde{X}_t \in {}^{\mathsf{c}}\chi_\epsilon) \leq \delta$ and the computational cost up to time $T^\star$ is bounded:*

$$\mathbb{E}\left(N_{call}^{T*}\right) \leq \frac{1}{d}\max\left(\left(\frac{2\Gamma_\gamma}{\delta}\right)^{2(\alpha+1)/(\min(1,\frac{\alpha}{2}-\gamma)-m^\star b+b\epsilon)}, \left(\frac{2\Gamma_2}{\delta}\right)^{(\alpha+1)/b\epsilon}\right).$$

■

**Proof.** In order to prove this statement, we use the inequalities from Theorem 10 treating each term separately.
We consider $T_1, T_2$ such that

$$\Gamma_\gamma \Gamma_2 (1+dT_1)^{m^\star b - \min(1,\frac{\alpha}{2}-\gamma) - b\epsilon} = \delta/2 \quad \text{and} \quad \Gamma_2(1+dT_2)^{-b\epsilon} = \delta/2.$$

This implies:

$$1 + dT_1 = \left(\frac{2\Gamma_\gamma}{\delta}\right)^{2/(\min(1,\frac{\alpha}{2}-\gamma)-m^\star b+b\epsilon)} \quad \text{and} \quad 1 + dT_2 = \left(\frac{2\Gamma_2}{\delta}\right)^{1/b\epsilon}. \tag{5.28}$$

Now we can define $T^\star$, the time after which the current state of the NSA belongs to $\chi_\epsilon$ with probability at least $1-\delta$, *i.e.*, $\forall t > T^\star$, $\mathbb{P}(\widetilde{X}_t \in \chi_\epsilon) \geq 1 - \delta$:

$$T^\star = \max(T_1, T_2).$$

We are interested in the computational cost up to time $T^\star$, more precisely the expected number of Monte Carlo simulations used up to $T^\star$. This is given by $\mathbb{E}\left(\sum_{k\geq 1} \mathbb{1}_{T_k < T^\star} N_{T_k}\right)$.

The value of this quantity cannot be computed exactly, but it can easily be upper bounded.

$$\begin{aligned}\mathbb{E}\left(\sum_{k\geq 1}\mathbb{1}_{T_k<T^\star}N_{T_k}\right) =&\mathbb{E}\left(\mathbb{E}\left(\sum_{k\geq 1}\mathbb{1}_{T_k<T^\star}N_{T_k}\right)|(T_k)_{k=1\cdots+\infty}\right)\\ =&\mathbb{E}\left(\sum_{k\geq 1}\mathbb{1}_{T_k<T^\star}n_{T_k}\right)\\ \leq&\mathbb{E}\left(\sum_{k\geq 1}\mathbb{1}_{T_k<T^\star}\right)n_{T^\star}\end{aligned}$$

The last inequality is implied by the fact that $n_t$ is an increasing function. Since $\sum_{k\geq 1}\mathbb{1}_{T_k<T^\star}$ is a Poisson variable of parameter $T^\star$, using the definition of $n_t$ one can see that:

$$\begin{aligned}\mathbb{E}\left(\sum_{k\geq 1}\mathbb{1}_{T_k<T^\star}N_{T_k}\right) \leq&T^\star(1+dT^\star)^\alpha \\ \leq&\frac{1}{d}(1+dT^\star)^{\alpha+1}.\end{aligned} \tag{5.29}$$

We conclude using Equation (5.28). ■

The rate of growth of the total computation number is mainly driven by the exponent of $\frac{1}{\delta}$ in the cost function. We are looking for the couple $(\alpha, b)$ that minimizes this quantity and fulfills the requirements of Theorem 10. We can split the problem into two sub-problems:

Case 1: $\frac{\alpha}{2}-\gamma>1$

$$\begin{aligned}&\min_{b,\alpha}\ \max\left(\frac{2(\alpha+1)}{1-m^\star b+b\epsilon},\frac{\alpha+1}{b\epsilon}\right)\\ s.t.&\\ &m^\star b<1 \text{ and } \alpha-2\gamma>2\end{aligned} \tag{5.30}$$

Case 2: $\frac{\alpha}{2}\leq 1$

$$\begin{aligned}&\min_{b,\alpha}\ \max\left(\frac{2(\alpha+1)}{\alpha/2-\gamma-m^\star b+b\epsilon},\frac{\alpha+1}{b\epsilon}\right)\\ s.t.&\\ &0<\gamma<\frac{\alpha}{2}-m^\star b \text{ and } \alpha-2\gamma\leq 2\end{aligned} \tag{5.31}$$

The solution of Equation (5.30) is obvious, the minimal value for $\alpha$ and the maximal for $b$. This means that $\alpha$ must be as close to 2 as possible and $b=\frac{1}{m^\star+\epsilon}$.

As for Equation (5.31), we consider two sub-cases. First suppose that:

$$\frac{2(\alpha+1)}{\alpha/2-\gamma-m^\star b+b\epsilon}\geq\frac{\alpha+1}{b\epsilon}\iff\alpha/2-\gamma-m^\star b\leq b\epsilon. \tag{5.32}$$

The function we want to minimize is strictly decreasing in $\alpha$ and strictly increasing in $b$, so its minimum value is attained for the maximal value of $\alpha$ and the minimal value of $b$, under the domain constraints given by (5.31) and (5.32), so the solution is:

$$\alpha=2(1+\gamma) \text{ and } b>\frac{1}{m^\star+\epsilon}. \tag{5.33}$$

In the second sub-case, supposing that the inequality (5.32) is inverted, the problem can be resumed at minimizing $(\alpha+1)/b\epsilon$, a decreasing function with respect to $b$, for

$$b \leq \frac{\alpha/2-\gamma}{(m^\star+\epsilon)} \text{ and } \alpha \leq 2.$$

Replacing $b$ by its maximal value the objective function becomes a decreasing function in $\alpha$, and therefore we obtain the same solution as before, defined in Equation (5.33). This is a quite comprehensive result, as it indicates that the lower the required accuracy in the solution space is, *i.e.*, $\epsilon$ increases and thus the size of $\chi_\epsilon$ does too, the faster the temperature can decrease to zero. We need to explore less the state space.

**Corollary 1.** *For the optimal parameters choice defined in Equation (5.33), an $\epsilon$-optimal solution is returned by NSA with probability $1-\delta$ at a computational cost at most :*

$$\frac{1}{d}\left(\frac{2\Gamma_\gamma}{\delta}\right)^{\frac{m^\star+\epsilon}{\epsilon}(3+2\gamma)},$$

*where $\Gamma_\gamma$ is defined in Theorem 10.*

This is rather costly but represents a general bound with few constraints on the function $J$. However, if the function $J$ has additional properties the bound can be significantly improved:

**Corollary 2.** *Suppose that $J$ has no well containing a local minimum, apart from the one containing the global minimum, i.e. $m^\star = 0$, then an $\epsilon$-optimal solution is returned by NSA with probability $1-\delta$ at a computational cost at most :*

$$\left(\frac{2\log\frac{1}{\delta}}{d\epsilon}\right)^3.$$

**Remark 3.** *We recover the polynomial dependency in $1/\epsilon$ and $\log 1/\delta$ of the state-of-the-art complexity results (c.f. [123] and [103]) which are of the order of $\epsilon^{-2}\log(\frac{1}{\delta})$ for strongly convex cost functions. As we relax this assumption and only consider cost functions with no local minimum, it seems coherent to observe a slight degradation of the complexity.*

**Proof.** In order to have an estimate of the computational cost in this setting we follow the same method as before and highlight only the main steps of the proof. First remark that in this case, Theorem 9 states that there exist $C, c > 0$ such that $\forall\beta \in \mathbb{R}^+$:

$$c \leq \gamma(\beta) \leq C \tag{5.34}$$

This changes the differential inequality obtained for $u_t = \|f_t - 1\|^2_{\mu_{\beta_t}}$ and thus (5.20) becomes :

$$u'_t \leq 2\left[-c + (\epsilon_t^+ - \epsilon_t^-) + (\frac{M}{2} + \frac{M}{4})\beta'_t\right] u_t + 2M\beta'_t + 2(\epsilon_t^+ - \epsilon_t^-).$$

We can apply Grönwall's Lemma and obtain the same type of inequality as before:

$$u_t \leq u_0 e^{\int_0^t -A_s + B_s \mathrm{d}s} + \int_0^t B_s e^{\int_s^t -A_h + B_h \mathrm{d}h} \mathrm{d}s. \tag{5.35}$$

where $B_t$ has the same form as before, $B_t = 2(\epsilon_t^+ - \epsilon_t^-) + 2M\beta'_t$ and $A_t = 2c$ .

The convergence of $u_t$ towards 0 can be proved now for a larger class of functions $n_t, \beta_t$, since:

$$A_t = \mathcal{O}(1) \text{ and } B_t = \mathcal{O}(\beta_t/\sqrt{n_t} \vee \beta'_t).$$

We no longer need to impose $\beta_t = \mathcal{O}(\log t)$. Let $\alpha, b, d > 0$. Define

$$n_t = (1+t)^{\alpha} \text{ and } \beta_t = d(1+t)^b.$$

Using (5.35) one can check that we have a finite upper bound on $u_t$ as soon as:

$$\{b < \alpha/2 \wedge 1, d > 0\} \text{ or } \{b = 1, \alpha \geq 2, 0 < d < c\}.$$

This in particular implies that the NSA algorithm converges a.s. to the set of global minimums of $J$. Furthermore for the first set of conditions one can prove using the same technique as in Appendix A.1 that :

$$u_t = \mathcal{O}(t^{\max(b-1, b-\alpha/2)}).$$

This means that there exits $\Gamma'_\gamma > 0$ such that for $t$ large enough $u_t \leq \Gamma'_\gamma t^{-\gamma}$, where $\gamma = -\max(b-1, b-\alpha/2)$. Using this, (5.27) and (5.25), for $t$ large enough, we get:

$$\begin{aligned}
\mathbb{P}(\widetilde{X}_t \in {}^{\mathsf{c}}\chi_\epsilon) &\leq \Gamma'_\gamma t^{-\gamma/2} e^{-\epsilon d(1+t)^b/2} + \Gamma_2 e^{-\epsilon d(1+t)^b} \\
&\leq e^{-\epsilon d(1+t)^b/2} (\Gamma'_\gamma t^{-\gamma/2} + \Gamma_2 e^{-\epsilon d(1+t)^b/2}) \\
&\leq e^{-\epsilon d(1+t)^b/2}
\end{aligned}$$

The last inequality is valid as soon as $t > \max\left[\left(\frac{2\log(2\Gamma_2)}{\epsilon d}\right)^{\frac{1}{b}} - 1, \left(2\Gamma'_\gamma\right)^{\frac{2}{\gamma}}\right]$. This is not a restrictive condition. Take for example the minimization of the $\|.\|_1$ over the subset $E = \{x \in \mathbb{Z}^p, \|x\|_\infty \leq n\}$ for some $n \in \mathbb{N}$. As $\Gamma_2 = \|E\| - 1 = (n+1)^p - 1$, the time for which the first part of the condition is fulfilled only grows linearly with the dimension of the search space. We show latter on that the optimal choice for $b$ is one and thus the second part of the condition can be omitted.

Let $\delta > 0$ be a fixed. Using the previous inequality one can compute $T^\star$ such that the confidence inequality constraint is satisfied:

$$T^\star_{\epsilon,\delta} = \left(-\frac{2\log\delta}{d\epsilon}\right)^{1/b} - 1.$$

Regarding the computational cost we remind the reader that (5.29) implies:

$$\mathbb{E}(N_{call}^{T^\star}) \leq n_{T^\star} T^\star \leq \left(-\frac{2\log\delta}{d\epsilon}\right)^{\frac{1+\alpha}{b}}.$$

We can optimize this bound with respect to $\alpha$ and $b$ in the same way as for Corollary 1. This leads to $\alpha = 2$ and $b < 1$ and thus to the desired results:

$$\mathbb{E}(N_{call}^{T^\star}) \leq \left(-\frac{2\log\delta}{d\epsilon}\right)^3.$$

■

## 5.6 Numerical experiments

In this section we first present some test cases, for which we use an additive Gaussian noise at each evaluation. We recover the theoretical results introduced by [72]. In a second part we present some results for the aircraft trajectory optimization problem. In this case the solution of the problem is unknown. We can only observe the total cost improvement in comparison with a trajectory optimized for a similar but deterministic setting.

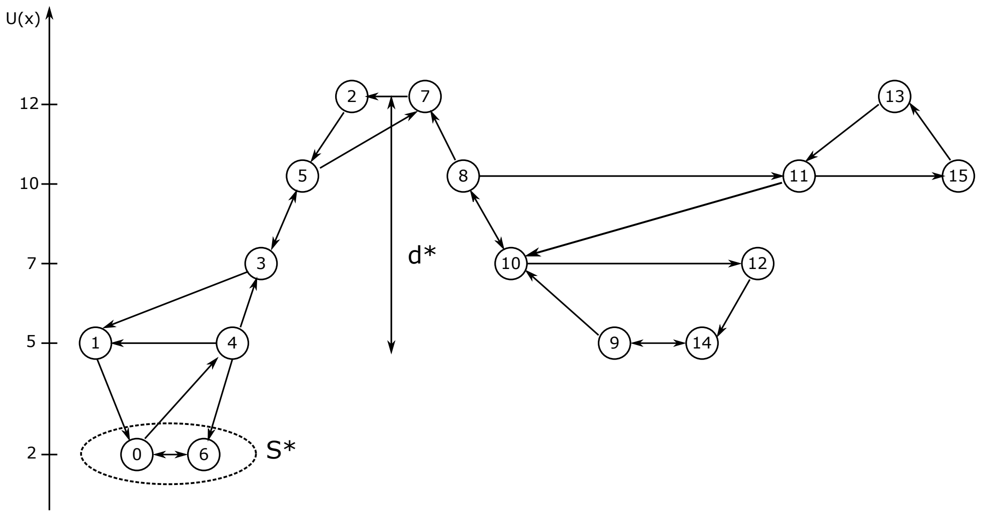


Figure 5.2: B. Hajek test case for the simulated annealing in a deterministic environment

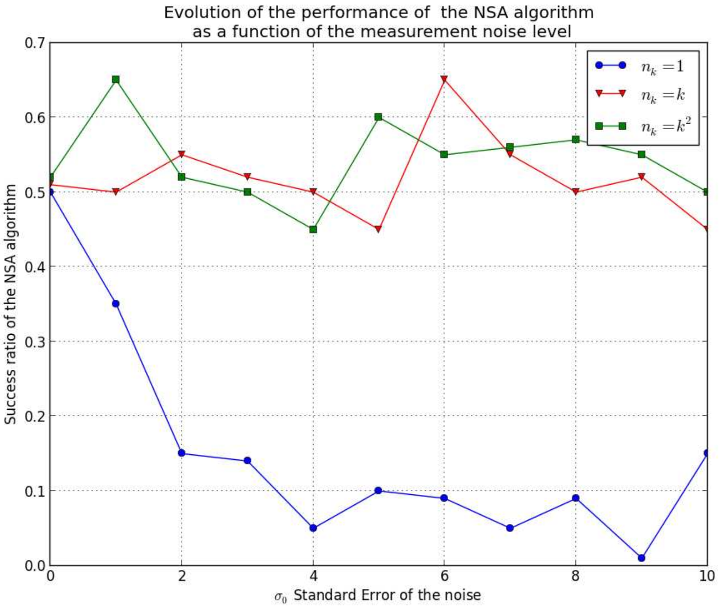


Figure 5.3: Convergence performance of NSA for the Hajek setting

**Basic exemple**
The first experimental setting we consider, was introduced in [73]. The cost function and the neighbourhood structure are represented on Figure 5.2. This is of particular interest as the function has two basins from which it is hard to escape. B. Hajek has shown that the following holds:

**Theorem 11** (B. Hajek [73])**.**
*If* $\beta_k = b\ log(k+2)$*, then* $b \leq d^\star \Leftrightarrow \lim_{k\to\infty} \mathbb{P}(X_k \in S^\star) = 1$*, where* $d^\star$ *is the maximum depth of a cup containing a local but not global minima. The depth of a cup is the maximal energy difference between two of it states and* $(X_k)_{k\in\mathbb{N}}$ *denotes the Markov chain generated by the classical simulated annealing.* ■

For a complete definition of $d^\star$, see [73]. We add Gaussian noises to the cost function of Figure 5.2 with different variance levels to highlight the fact that if no sampling is performed the simulated annealing performance becomes rapidly very poor as the variance increases. On the other hand it appears that the performance of the NSA for a linear

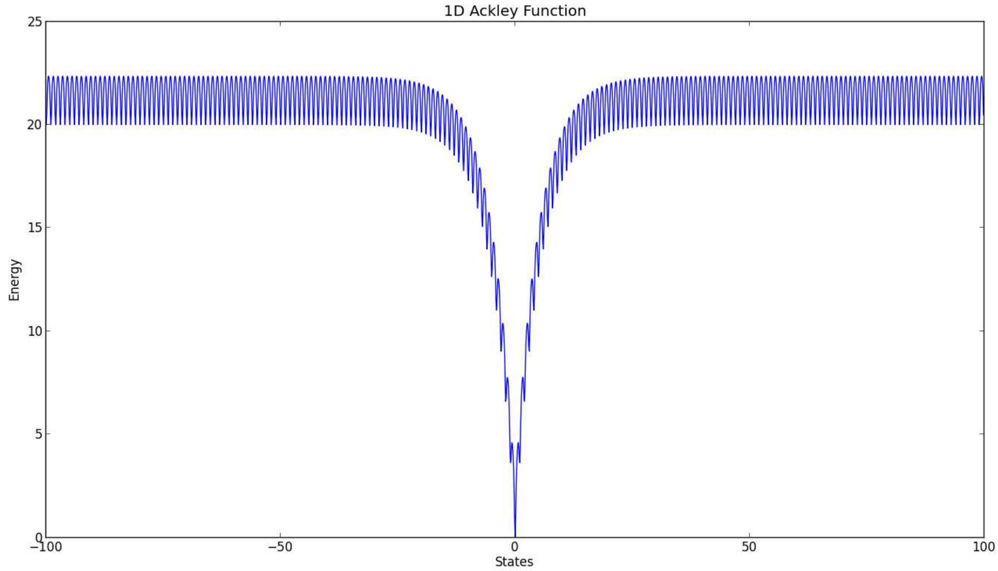


Figure 5.4: Ackley 1D Test Function

increase of the mean number of samples is as good a quadratic one. These results are summarized on Figure 5.3.

### Ackley test function

We introduce a second test case to further asses these observations. We consider the uniformly (2000 points) discretized version of the Ackley function in one dimension on $[-100, 100]$. This function has many local minima as shown on Figure 5.4. Figures 5.5 displays the convergence results for different levels of variance of the noise for each estimation schedule introduced in this chapter. We observe that the only case where the convergence is not impacted by the noise variance increase is the $n_t = t^2$ case. These results highlight the fact that a logarithmic sampling schedule is not appropriate in general, even in the Gaussian case. This invalidates partially the hypotheses introduced in [60]. A clear gap is highlighted between the linear and the quadratic schedule.

### Aircraft trajectory optimization

We use a black box trajectory evaluator for a long range commercial aircraft. We consider a direct shooting method for optimizing the vertical part of the trajectory. As displayed on Figure 5.6, the vertical path is made of a sequence of flight segments at constant altitude called steps. The transitions between those steps are called step climbs. This has been put in place by the international authorities to ease the air traffic control. Aircraft can only fly at a finite set of altitudes. The steps climbs are transition phases that must be very short. The Figure 5.6 is a conceptual. It does not reflect the real scale of the different phases. Our optimization variables are the vectors of position of the steps and the vector of steps' altitude, denoted respectively $x$ and $h$ on Figure 5.6. The structure of this airspace strongly limits the number of steps. We will only consider the problem with an *a priori* number of steps. There are two main reasons why the aircraft might vary its altitude during a flight (optimizing fuel consumption and air traffic control). Because of fuel consumption, the aircraft weight is decreasing during the flight. Analyzing the laws of flight physics, it can be shown that there exists an altitude at which the fuel consumption per flown distance unit is minimal. It can also be shown that this altitude increases as the

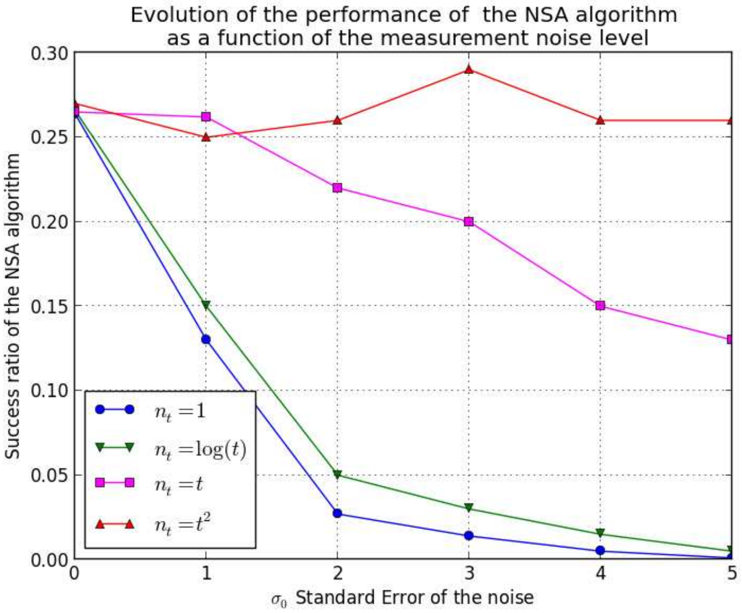


Figure 5.5: Performance of the NSA algorithm as a function of the level of noise on the evaluation of the cost function

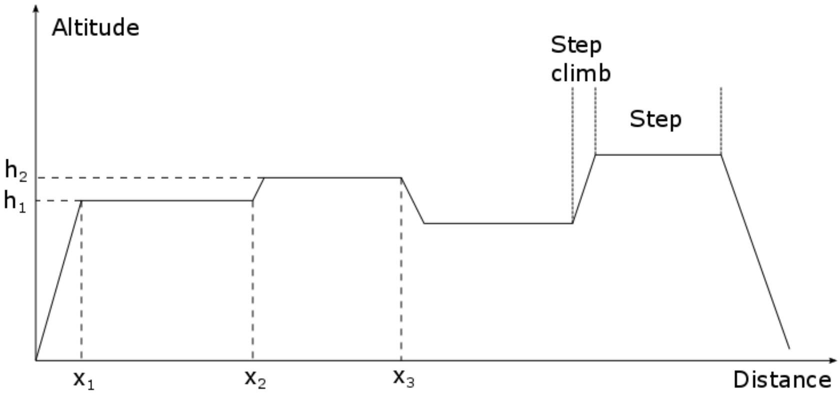


Figure 5.6: Aircraft trajectory, structure of the vertical path

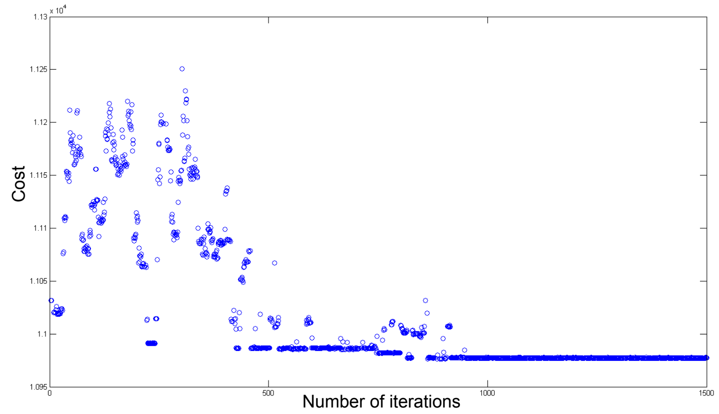


Figure 5.7: NSA descent: Aircraft trajectory optimization problem

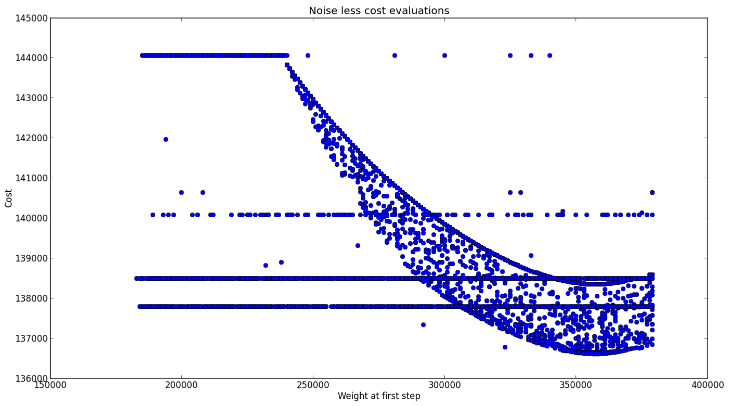


Figure 5.8: Sampling of the cost function along the first step position for a 3 step vertical path

weight decreases. This last statement is however only true if there is no wind. It is easily understandable that for some particular wind map configuration it might be preferable to target lower altitudes at lower weights.

The choice of the vertical path must be declared to the authorities before the flight to ensure traffic manageability. Airlines operating aircraft have therefore a stochastic optimization problem to solve. This is a stochastic problem for two main reasons. First, they only access predicted weather conditions that suffer some uncertainty. Second, the airspace is not empty and sometimes air traffic controllers might refuse some altitude changes because of the presence of other aircraft. As the weather, the traffic is not known in advance.

We applied NSA to the problem of finding an optimal 3 steps configuration. An example of the current solution cost evolution with respect to the number of iterations is displayed on Figure 5.7.

We observe a very quick convergence to a low cost trajectory. We do not claim it is a general behaviour. It might be due to the structure of the cost function. Figure 5.8, shows how the cost evolves with respect to the ground position of the first step. It is obviously not convex but has some regularity. We can observe some flat parts. This explains why gradient based methods would fail solving this problem.

As for the previous problems we have observed that the increase sampling condition must be satisfied to ensure a good behaviour of the algorithm.

# Appendix A

# Complements of proof

## A.1 Proof of bound Equation (5.26)

**Proof.**[Proof of Theorem 10]
Let $\beta_t = b\log(td+1)$ and $n_t = (1+td)^\alpha$.
Recall Equation (5.19):

$$u_t' \leq 2\left[-ce^{-\beta_t m^\star} + (\epsilon^+ - \epsilon^-) + \left(\frac{M}{2} + \frac{M}{4}\right)\beta_t'\right]u_t + 2M\beta_t' + 2(\epsilon^+ - \epsilon^-)$$

Let $A_t = 2ce^{-\beta_t m^\star}$ and $B_t = 2(\epsilon^+ - \epsilon^-) + 2M\beta_t'$.

Applying Grönwall's Lemma for the previous relation gives:

$$u_t \leq u_0 e^{\int_0^t -A_s + B_s \mathrm{d}s} + \int_0^t B_s e^{\int_s^t -A_h + B_h \mathrm{d}h}\mathrm{d}s$$

Under Assumptions 5.2.2, there exist $b, d > 0$ such that $\beta_t = b\log(1+td)$. This implies:

$$\beta_t' = \frac{bd}{1+td} \text{ and } e^{-m^\star \beta_t} = \left(\frac{1}{1+td}\right)^{m^\star b}.$$

Using the definition of $\epsilon^-, \epsilon^+$ we have:

$$\epsilon^+ - \epsilon_t^- = 2\sqrt{2\pi}\beta_t\sigma\left[\frac{e^{\frac{\beta_t^2\sigma^2}{2(1-\delta)n_t}}}{\sqrt{(1-\delta)n_t}} + e^{\beta_t^2\sigma^2 - an_t}\right].$$

This implies that when $t$ goes to infinity:

$$A_t = \mathcal{O}\left(\frac{1}{t^{m^\star b}}\right) \text{ and } B_t = \mathcal{O}\left(\frac{1}{t} \vee \frac{\log t}{t^{\alpha/2}}\right)$$

In order to highlight the mains ideas of the proof we will try to simplify the notations as much as possible. First observe that for all $\alpha > 0$ and all $0 < \gamma < \alpha/2$ , $\frac{\log t}{t^{\alpha/2}} = o\left(\frac{1}{t^{\alpha/2-\gamma}}\right)$.

Hence we can assume there exist $A, B > 0$ and $\delta_1, \delta_2$ such that:

$$A_t = A\frac{d}{(1+td)^{\delta_1}} \text{ and } B_t \leq B\frac{d}{(1+td)^{\delta_2}}$$

where $\delta_1 = m^\star b$ and $\delta_2 = \min(1, -\gamma + \alpha/2)$. Since $\min(1, \alpha/2) > m^\star b$, and $\gamma$ can be chosen arbitrarily close to 0, we choose it such that $\delta_1 < \delta_2$. This means that $0 < \gamma < \alpha/2 - m^\star b$.

**Remark 4.** *The choice of $\gamma$ influences the choice of $B$. If $B_t = \mathcal{O}\left(\frac{\log t}{t^{\alpha/2}}\right)$, there exists $C_B > 0$ such that $B_t \leq C_B \frac{\log t}{t^{\alpha/2}}, \forall t$. The constant $B$ is then such that $\forall t$, $C_B \frac{\log t}{t^{\gamma}} \leq B$. Hence we can choose:*

$$B = \frac{C_B}{e\gamma}. \tag{A.1}$$

Let $T_t^1 = u_0 e^{\int_0^t -A_s + B_s \mathrm{d}s}$ and $T_t^2 = \int_0^t B_s e^{\int_s^t -A_h + B_h \mathrm{d}h} \mathrm{d}s$.

The first term $T_t^1$ is always easy to deal with and one can check that under the theorem's assumptions we always have $T_t^1 = o(1/t^{\delta_1 - \delta_2})$ when $t$ goes to infinity. As for the second term, using a substitution gives:

$$\begin{aligned}
T_t^2 &\leq \int_1^{1+td} \frac{B}{s^{\delta_2}} e^{\int_s^{1+td} -\frac{A}{h^{\delta_1}} + \frac{B}{h^{\delta_2}} \mathrm{d}h} \mathrm{d}s \\
&\leq e^{-\frac{A(1+td)^{1-\delta_1}}{1-\delta_1} + \frac{B(1+td)^{1-\delta_2}}{1-\delta_2}} \int_1^{1+td} \frac{B}{s^{\delta_2}} e^{\frac{As^{1-\delta_1}}{1-\delta_1} - \frac{Bs^{1-\delta_2}}{1-\delta_2}} \mathrm{d}s
\end{aligned} \tag{A.2}$$

For the last inequality we assume $\delta_2 \neq 1$ which corresponds to the case $\alpha \leq 2$.

Let $I_t = \int_1^{1+td} \frac{B}{s^{\delta_2}} e^{\frac{As^{1-\delta_1}}{1-\delta_1} - \frac{Bs^{1-\delta_2}}{1-\delta_2}} \mathrm{d}s$ and $f_s = \frac{As^{1-\delta_1}}{1-\delta_1} - \frac{Bs^{1-\delta_2}}{1-\delta_2}$. Let $T_0$ be such that for all $s \geq T_0$, $s^{\delta_2 - \delta_1} \geq \frac{B+1}{A}$ (for instance $T_0 = (\frac{B}{A} + 1)^{1/(\delta_2 - \delta_1)}$). We can write $I_t$ as follows:

$$\begin{aligned}
I_t &= \int_1^{T_0} \frac{B}{s^{\delta_2}} e^{f_s} \mathrm{d}s + \int_{T_0}^{1+td} \frac{B}{s^{\delta_2}} e^{f_s} \mathrm{d}s \\
&= K_{T_0} + \int_{T_0}^{1+td} \frac{B}{s^{\delta_2}(As^{-\delta_1} - Bs^{-\delta_2})} e^{f_s} f_s' \mathrm{d}s \\
&= K_{T_0} + \left[\frac{B}{As^{\delta_2 - \delta_1} - B} e^{f_s}\right]_{T_0}^{1+td} + \int_{T_0}^{1+td} \frac{AB(\delta_2 - \delta_1)s^{\delta_2 - \delta_1 - 1}}{(As^{\delta_2 - \delta_1} - B)^2} e^{f_s} \mathrm{d}s
\end{aligned}$$

Since $\delta_1 < \delta_2$, $\frac{A(\delta_2 - \delta_1)s^{-\delta_1 - 1}}{(As^{\delta_2 - \delta_1} - B)^2}$ goes to 0 when $s$ goes to infinity. Moreover one can check that for all $s \geq T_0$ this quantity is smaller than 1/2. Using this we get:

$$I_t \leq K_{T_0} + \left[\frac{B}{As^{\delta_2 - \delta_1} - B} e^{f_s}\right]_{T_0}^{1+td} + \frac{1}{2} I_t$$

and therefore for all $t \geq T_0$:

$$I_t \leq 2\left[\frac{B}{As^{\delta_2 - \delta_1} - B} e^{f_s}\right]_1^{1+td} + 2K_{T_0}$$

This gives:

$$\begin{aligned}
T_t^2 &\leq 2e^{-f_{1+td}} \left(\left[\frac{B}{As^{\delta_2 - \delta_1} - B} e^{f_s}\right]_1^{1+td} + K_{T_0}\right) \\
&\leq 2\left[\frac{B}{A(1+td)^{\delta_2 - \delta_1} - B}\right] + 2\left(K_{T_0} - \frac{B}{A - B} e^{f_1}\right) e^{-f_{1+td}}
\end{aligned}$$

Regrouping the terms we obtain for all $t \geq T_0$:

$$u_t \leq \left[u_0 + 2\left(K_{T_0} - \frac{B}{A - B} e^{f_1}\right)\right] e^{-f_{1+td}} + \frac{2B}{A(1+td)^{\delta_2 - \delta_1} - B} \tag{A.3}$$

Since the first term is a $\mathcal{O}(e^{-f_t})$ and therefore a $o(t^{\delta_1-\delta_2})$ it is obvious that:

$$u_t = \mathcal{O}\left(\frac{1}{t^{\delta_2-\delta_1}}\right) \text{ when } t \to \infty \tag{A.4}$$

This means that for all $\gamma \in (0, \alpha/2 - m^\star b)$ there exists $\Gamma_\gamma > 0$ such that:

$$u_t \leq \Gamma_\gamma t^{-\alpha/2+m^\star b+\gamma} \text{ for all } t \geq T_0. \tag{A.5}$$

Since $f_t = \mathcal{O}(t^{1-\delta_1})$, $e^{-f_t}t^{\delta_1-\delta_2}$ goes very fast to zero and therefore the size of $\Gamma_\gamma$ is mainly driven by $\dfrac{2B}{A}$. Using Equation (A.1) one can see that:

$$\Gamma_\gamma \simeq \frac{1}{\gamma}.$$

If $\alpha > 2$, $\delta_2 = 1$ and Equation (A.2) becomes of the form:

$$T_t^2 \leq e^{-\frac{A(1+td)^{1-\delta_1}}{1-\delta_1}+B\log(1+td)} \int_1^{1+td} \frac{B}{s^{1+B}} e^{\frac{As^{1-\delta_1}}{1-\delta_1}} \mathrm{d}s.$$

Using a similar procedure one can check that in this case we also have an inequality similar to Equation (A.3) , $T_2^t$ remains a $\mathcal{O}\left(\dfrac{1}{t^{1-\delta_1}}\right)$, and Equation (A.4) and Equation (A.5) still hold.

**Remark 5.** *In this case one can choose* $\gamma = \alpha/2 - 1$*, this way* $\min(1, \frac{\alpha}{2} - \gamma) = 1$ *and* $\Gamma_\gamma$ *is minimal.*

■

## A.2 Definition of $m^\star$

In this section we prove that the definition of $m^\star$, *i.e.* Equation (5.4), is equivalent to the definition provided by [77].

**Lemma 4.** *Let,* $m^\star_{HS} := \max\limits_{x,y\in E}\left\{\min\limits_{p\in P_{xy}}\left\{\max\limits_{z\in p} J(z)\right\} - J(y) - J(x) + \min_u J(u)\right\}$*.*
*Then*

$$m^\star = m^\star_{HS}$$

■

**Proof.** Let $x, y \in E$ and denote $H_{xy} := \min\limits_{p\in P_{xy}}\left\{\max\limits_{z\in p} J(z)\right\}$.
First it can be noticed that if $x$ is a global minimum of $J$ then we have

$$H_{x,y} - J(y) - J(x) + \min_u J(u) = H_{x,y} - J(y) \tag{A.6}$$

Thus $m^\star_{HS} \geq H_{x^\star,y} - J(y)$ for any $y$ in $E$, where $x^\star$ is a global minimum of $J$.

Recall that $m^\star = \max\limits_{x,y\in E}\{H_{xy} - \max(J(y), J(x))\}$. As the set of paths going from $x$ to $y$ containing a global minimum $x^\star$ is a subset of the paths going from $x$ to $y$, we have:

$$H_{xy} \leq \max(H_{x^\star x}, H_{x^\star y})$$

Let $x, y \in E$ such that $m^\star = H_{xy} - \max\left(J(y), J(x)\right)$,

$$\begin{aligned} m^\star &\leq \max\left(H_{x^\star x}, H_{x^\star y}\right) - \max\left(J(y), J(x)\right) \\ &\leq \max\left(H_{x^\star x} - J(x), H_{x^\star y} - J(y)\right) \\ &\leq m^\star_{HS} \end{aligned}$$

On the other hand, as $\forall\ x, y\ \in\ E,$ we have $-\min\left(J(y), J(x)\right) + \min_u J(u) \leq 0$, so

$$H_{xy} - J(y) - J(x) + \min_u J(u) \leq H_{xy} - \max\left(J(y), J(x)\right)$$

This implies $m^\star_{HS} \leq m^\star$, which completes the proof. ■

# Perspectives

Les travaux effectués pendant cette thèse ouvrent plusieurs perspectives, la plupart d'entre elles étant déjà mentionnées. Commençons par les plus évidentes, liées aux travaux en cours, notamment les chapitres 3 et 4.

Dans le chapitre 3, une heuristique est proposée pour estimer le barycentre d'un très grand graphe en utilisant un clustering préliminaire. Un travail, en cours, est de tester les performances numériques de notre méthode sur plusieurs graphes. Une perspective immédiate est d'étudier l'influence du clustering sur la précision des résultats en fonction du type de graphe, et éventuellement de proposer des méthodes alternatives pour les graphes ayant une structure très différente de celle d'un graphe provenant d'un réseau routier.

Par rapport au chapitre 4, une perspective évidente est de prouver les conjectures énoncées concernant l'existence et la convergence du processus de Markov. Par rapport à l'existence on peut imaginer utiliser des méthodes à la Freidlin et Wentzell [64]. En ce qui concerne la convergence, le seul point difficile dans l'adaptation de la preuve utilisée dans le chapitre 2 pour les graphes quantiques, est de montrer une inégalité de log-Sobolev pour une mesure de Gibbs. Plus précisément, l'enjeu est de montrer qu'il existe $p > 2$ pour lequel l'inégalité de Sobolev classique pour la mesure de Lebesgue normalisée sur notre espace d'intérêt $\mathcal{G}_1$ est vraie. Comme $\mathcal{G}_1$ peut être couvert par un nombre fini d'ensembles homéomorphes à des compacts de $\mathbb{R}^2$, la théorie classique des équations aux dérivées partielles semble un bon point de départ dans la recherche d'une solution.

Une fois cette difficulté résolue, il existe plusieurs pistes de généralisation. D'abord on peut s'intéresser à l'optimisation d'autres fonctions dans le même cadre. Dans un second temps, on peut aussi étendre la méthode pour estimer des composantes principales d'ordre plus grand.

Parmi les perspectives, nous n'oublions pas l'aspect pratique de ce chapitre, notamment le fait que les études numériques sont réalisées dans un cadre restrictif, et doivent être à la fois améliorées et généralisées.

Une autre perspective intéressante est d'étudier du point de vue théorique les algorithmes proposés aux chapitres 2 et 4, en leur associant à chaque fois un processus qui évolue comme un mouvement brownien entre des temps aléatoires auxquels il saute dans la direction d'un $Y_n$ (à la place d'un processus dirigé par un mouvement brownien et un terme de rappel qui dépend du processus auxiliaire $Y$). On peut envisager de montrer la convergence d'un tel processus en adaptant la méthode utilisée par Arnaudon et Miclo dans [15] pour estimer les $p$-moyennes sur le cercle.

On peut observer dans les simulations, comme on l'a mentionné dans le chapitre 2, que les performance des algorithmes de type Metropolis Hastings sont moins bonnes sur des graphes pour lesquels la distribution des degrés des nœuds est très variable (voir les expériences sur les sous-graphes Facebook présentées en section 2.4). On peut donc supposer que l'efficacité des algorithmes de ce type peut être influencée par le choix de voisinage, surtout quand le système de voisinage n'est pas imposé par la structure de l'espace. Une question naturelle qui se pose est à quelle point le voisinage influence la rapidité avec laquelle la chaine de Markov explore l'espace et comment le choisir pour améliorer les performances algorithmiques.

Enfin, on présente une piste de recherche qui, tout en étant motivée par les thématiques étudiées dans la thèse, est un peu plus éloignée des travaux présentés ici, surtout par rapport aux outils nécessaires. Cette piste est liée à la constante $c^\star(U_\nu)$, définie dans le chapitre 2. On a vu que cette constante représente une limite dans le choix du schéma de température et donc la connaitre peut représenter un intérêt pratique. La calculer de façon explicite n'est pas envisageable sans avoir accès aux valeurs de $U_\nu$, donc il semble plus propice d'essayer de l'estimer.

Une façon de l'estimer peut être au travers de l'étude de la fonction $U_\nu$ sur des graphes aléatoires. Idéalement, on souhaite estimer $c^\star(U_\nu)$ sur un modèle de graphes à blocs stochastiques, car ceux-ci ont une structure similaire aux réseaux sociaux et sont souvent utilisés pour les modéliser. Étant toujours souhaitable, au moment de se lancer dans un nouveaux domaine, de commencer par des choses plus élémentaires, on choisit d'abord de simplifier le problème. On considère donc un graphe complet avec des arêtes de longueurs aléatoires distribuées suivant des variables exponentielles indépendantes. De plus, on considère $\nu$ comme étant uniforme et on s'intéresse à la function $U$, correspondant à la médiane. Si la taille du graphe est $n$, on utilise la notation $U_n$ :

$$U_n(x) = \frac{1}{n} \sum_{y \in V} d(x, y).$$

En utilisant une approche inspirée de l'étude du premier temps de passage dans un modèle de percolation, adoptée aussi dans [84], on montre que pour un sommet $x$ pris au hasard, à une mise à échelle près, on peut calculer la distribution asymptotique de $U_n(x)$ :

$$\lim_{n \to \infty} U_n(x) - \log n \xrightarrow{\mathcal{L}} U_x$$

où $U_x$ est une variable aléatoire de loi Gumbel, $U_x \sim \text{Gumbel}(0, 1)$, qui admet comme fonction de répartition $F(y) = \exp(-e^{-y})$.

Après ce (petit) premier pas, l'étape suivante est d'obtenir des résultats quantifiant la loi jointe de $(U_n(x), U_n(y))$ pour toute paire de sommets $(x, y)$ et d'utiliser des outils déjà développés dans la théorie des matrices aléatoires pour étudier asymptotiquement le trou spectral de $U_n$.

# Bibliography

[1] E. Aarts and J. Korst. Simulated annealing and Boltzmann machines. New York, NY; John Wiley and Sons Inc., 1988.

[2] E. Aarts and J. Korst. Boltzmann machines for travelling salesman problems. European Journal of Operational Research, 39(1):79–95, 1989.

[3] H. Abdi and L. J. Williams. Principal component analysis. Wiley Interdisciplinary Reviews: Computational Statistics, 2(4):433–459, 2010.

[4] D. Acemoğlu, G. Como, F. Fagnani, and Ozdaglar A. Opinion fluctuations and disagreement in social networks. Math. Oper. Res., 38(1):1–27, February 2013.

[5] B. Afsari. Riemannian $L^p$ center of mass: existence, uniqueness, and convexity. Proc. Amer. Math. Soc., 139(2):655–673, 2011.

[6] M. Agueh and G. Carlier. Barycenters in the Wasserstein space. SIAM J. Math. Anal., 43(2):904–924, 2011.

[7] R. Albert and A-L. Barabási. Statistical mechanics of complex networks. Rev. Modern Phys., 74(1):47–97, 2002.

[8] T.M. Alkhamis, M.A. Ahmed, and V.K. Tuan. Simulated annealing for discrete optimization with estimation. European Journal of Operational Research, 116(3):530–544, 1999.

[9] S. Allassonnière, E. Kuhn, and A. Trouvé. Construction of bayesian deformable models via stochastic approximation algorithm: A convergence study. Bernoulli, 16:641–678, 2010.

[10] N. Alon and J. H. Spencer. The probabilistic method. Wiley Series in Discrete Mathematics and Optimization. John Wiley & Sons, Inc., Hoboken, NJ, fourth edition, 2016.

[11] P. Alquier, N. Friel, R. Everitt, and A. Boland. Noisy Monte Carlo: convergence of Markov chains with approximate transition kernels. Stat. Comput., 26(1-2):29–47, 2016.

[12] M. Arnaudon, C. Dombry, A. Phan, and L. Yang. Stochastic algorithms for computing means of probability measures. Stochastic Process. Appl., 122(4):1437–1455, 2012.

[13] M. Arnaudon and L. Miclo. Means in complete manifolds: uniqueness and approximation. ESAIM: Probability and Statistics, 18:185–206, 2014.

[14] M. Arnaudon and L. Miclo. A stochastic algorithm finding generalized means on compact manifolds. Stochastic Processes and their Applications, 124:3463–3479, 2014.

[15] M. Arnaudon and L. Miclo. A stochastic algorithm finding p-means on the circle. Bernoulli, in press, 2016.

[16] J-Y. Audibert and S. Bubeck. Best arm identification in multi-armed bandits. In COLT-23th Conference on Learning Theory-2010, pages 13–p, 2010.

[17] B. F. Auer and R. Bisseling. Graph coarsening and clustering on the gpu. Graph Partitioning and Graph Clustering, 588, 2012.

[18] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. Machine learning, 47(2-3):235–256, 2002.

[19] F. Bach and M. Jordan. Learning spectral clustering. Advances in Neural Information Processing Systems, pages 305–312, 2004.

[20] M. Baker and X. Faber. Metrized graphs, laplacian operators, and electrical networks. American Mathematical Society, 2005.

[21] D. Bakry, I. Gentil, and M. Ledoux. Analysis and geometry of Markov diffusion operators, volume 348. Springer Science & Business Media, 2013.

[22] D. Barden, H. Le, and M. Owen. Central limit theorems for Fréchet means in the space of phylogenetic trees. Electronic Journal of Probability, 18:1–25, 2013.

[23] R. Bhattacharya and L. Lin. A central limit theorem for fréchet means. arXiv preprint arXiv:1306.5806, 2013.

[24] R. Bhattacharya and V. Patrangenaru. Large sample theory of intrinsic and extrinsic sample means on manifolds. Ann. Statist., 31:1–29, 2003.

[25] J. Bierkens, P. Fearnhead, and G. Roberts. The zig-zag process and super-efficient sampling for bayesian analysis of big data. arXiv preprint arXiv:1607.03188, 2016.

[26] J. Bigot. Fréchet means of curves for signal averaging and application to ECG data analysis. Annals of Applied Statistics, 7:1837–2457, 2013.

[27] J. Bigot, E. Cazelles, and N. Papadakis. Regularization of barycenters in the wasserstein space. arXiv preprint arXiv:1606.01025, 2016.

[28] J. Bigot and B. Charlier. On the consistency of Fréchet means in deformable models for curve and image analysis. Electron. J. Stat., 5:1054–1089, 2011.

[29] J. Bigot and S. Gadat. A deconvolution approach to estimation of a common shape in a shifted curves model. Annals of Statistics, 38:224–243, 2010.

[30] J. Bigot and X. Gendre. Minimax properties of Fréchet means of discretely sampled curves. Annals of Statistics, 41:923–956, 2013.

[31] J. Bigot, R. Gouet, and Lopez A. Geometric PCA of images. Siam, J. Imaging Sci., 6:1851–1879, 2013.

[32] J. Bigot, R. Gouet, T. Klein, and A. Lopez. Geodesic PCA in the wasserstein space by convex PCA. Annales de l'Institut Henri Poincaré B: Probability and Statistics, to appear, 2015.

[33] M. Blazere. Inférence statistique en grande dimension pour des modèles structurels. Modèles linéaires généralisés parcimonieux, méthode PLS et polynômes orthogonaux et détection de communautés dans des graphes. 2015. Thèse de doctorat, Mathématiques Appliquées Toulouse, INSA 2015.

[34] B. Bollobás. Random graphs, volume 73 of Cambridge Studies in Advanced Mathematics. Cambridge University Press, Cambridge, second edition, 2001.

[35] D. Bontemps and S. Gadat. Bayesian methods for the shape invariant model. Electronic Journal of Statistics, 8:1522–1568, 2014.

[36] S. Boucheron, G. Lugosi, and P. Massart. Concentration inequalities: A nonasymptotic theory of independence. OUP Oxford, 2013.

[37] O. Brandiere and M. Duflo. Les algorithmes stochastiques contournent-ils les pieges. 1996.

[38] J. Branke, S. Meisel, and C. Schmidt. Simulated annealing in the presence of noise. Journal of Heuristics, 14(6):627–654, 2008.

[39] H. Brezis. Analyse fonctionelle. Masson, Paris, 1987.

[40] S. Bubeck, R. Munos, G. Stoltz, and C. Szepesvári. X-armed bandits. Journal of Machine Learning Research, 12:1587–1627, 2011.

[41] A. Bull. Convergence rates of efficient global optimization algorithms. The Journal of Machine Learning Research, 12:2879–2904, 2011.

[42] O. Catoni. Rough large deviation estimates for simulated annealing : application to exponential schedules. Ann. Probab., 20:1109–1146, 1992.

[43] B. Charlier. Étude des propriétés statistiques des moyennes de Fréchet dans des modèles de déformations pour l'analyse de courbes et d'images en grande dimension. 2011. Thèse de doctorat, Mathématiques appliquées Toulouse 3, 2011.

[44] B. Charlier. Necessary and sufficient condition for the existence of a Fréchet mean on the circle. ESAIM Probab. Stat., 17:635–649, 2013.

[45] F. Chung and L. Lu. The average distance in a random graph with given expected degrees. Internet Mathematics, 1(1):91–113, 2003.

[46] I. Csiszár. Information-type measures of difference of probability distributions and indirect observations. Stud. Sci. Math. Hung., 2:299–318, 1967.

[47] M. Cuturi and A. Doucet. Fast computation of wasserstein barycenters. In International Conference on Machine Learning, pages 685–693, 2014.

[48] B. C. Davis. Medical image analysis via Frechet means of diffeomorphisms. ProQuest LLC, Ann Arbor, MI, 2008. Thesis (Ph.D.)–The University of North Carolina at Chapel Hill.

[49] P. Delicado. Dimensionality reduction when data are density functions. Comput. Statist. Data Anal., 55(1):401–420, 2011.

[50] L. Devilliers, S. Allassonnière, A. Trouvé, and X. Pennec. Template estimation in computational anatomy: Fréchet means in top and quotient spaces are not consistent. working paper or preprint, August 2016.

[51] E. W. Dijkstra. A note on two problems in connexion with graphs. Numer. Math., 1:269–271, 1959.

[52] L. C. W. Dixon and G. P. Szegö. Towards global optimisation 2. North-Holland Amsterdam, 1978.

[53] A. Doucet, M. K. Pitt, G. Deligiannidis, and R. Kohn. Efficient implementation of Markov chain Monte Carlo when using an unbiased likelihood estimator. Biometrika, 102(2):295–313, 2015.

[54] I. L. Dryden and K. V. Mardia. Statistical Shape Analysis. Wiley, New York, 1998.

[55] P. Erdös and A. Rényi. The evolution of random graphs. Magyar Tud. Akad. Mat. Kutato Int. Kozl., 5:17–61, 1960.

[56] P. Erdos and A. Renyi. On random graphs. I. Publ. Math. Debrecen, 6:290-297, 1967.

[57] E. Estrada. Introduction to Complex Networks. Structure and Dynamics, chapter of Evolutionary Equations with Applications to Natural Sciences. Lecture Notes in Mathematics. Springer, 2015.

[58] S. N. Ethier and T. Kurtz. Markov processes. Characterization and convergence. Wiley Series in Probability and Statistics. John Wiley & Sons, 2005.

[59] L. Euler. Solutio problematis ad geometriam situs pertinentis. Commentarii Academiae Scientiarum Imperialis Petropolitanae 8: 128–140., 1752.

[60] T.MA. Fink. Inverse protein folding, hierarchical optimisation and tie knots. PhD thesis, University of Cambridge, 1998.

[61] M. Fréchet. Les éléments aléatoires de nature quelconque dans un espace distancié. Annales de l'Institut Henri Poincaré (B), 10:215–310, 1948.

[62] M. Freidlin and S.J. Sheu. Diffusion processes on graphs: stochastic differential equations, large deviation principle. Probab. Theory Relat. Fields, 116:181–220, 2000.

[63] M. I. Freidlin and A. D. Wentzell. Random perturbations of dynamical systems, volume 260 of Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]. Springer-Verlag, New York, second edition, 1979. Translated from the 1979 Russian original by Joseph Szücs.

[64] M. I. Freidlin and A. D. Wentzell. Diffusion processes on graphs and the averaging principle. Ann. Probab., 21(4):2215–2245, 1993.

[65] M. I. Freidlin and A. D. Wentzell. Random perturbations of Hamiltonian systems, volume 523 of Memoirs of the American Mathematical Society. A.M.S., New York, 1995.

[66] S. Gadat, I. Gavra, and L. Risser. How to calculate the barycenter of a weighted graph. arXiv preprint arXiv:1605.04148, 2016.

[67] S. B. Gelfand and S. K. Mitter. Simulated annealing with noisy or imprecise energy measurements. J. Optim. Theory Appl., 62(1):49–62, 1989.

[68] C.E. Ginestet. Strong consistency of set-valued fréchet sample means in metric spaces. Preprint, 2013.

[69] A. Goldenberg, A. X. Zheng, S.E. Fienberg, and E. M. Airoldi. A survey of statistical network models. Found. Trends Mach. Learn., 2(2):129–233, February 2010.

[70] D. Groisser. Newton's method, zeroes of vector fields, and the Riemannian center of mass. Adv. in Appl. Math., 33(1):95–135, 2004.

[71] R. Guimera and L. N. Amaral. Functional cartography of complex metabolic networks. Nature, 433(7028):895–900, 2005.

[72] W.J. Gutjahr and G.Ch. Pflug. Simulated annealing for noisy cost functions. Journal of Global Optimization, 8(1):1–13, 1996.

[73] B. Hajek. Cooling schedules for optimal annealing. Mathematics of operations research, 13(2):311–329, 1988.

[74] S.L. Hakimi. On locating new facilities in a competitive environment. European J. Oper. Res., 12(1):29–35, 1983.

[75] M. Handcock and K.Gile. Modeling social networks from sampled data. The Annals of Applied Statistics, 2010.

[76] R. Holley, D. Kusuoka, and D. Stroock. Asymptotics of the spectral gap with applications to the theory of simulated annealing. J. Funct. Anal., 83(2):333–347, 1989.

[77] R. Holley and D. Stroock. Simulated annealing via Sobolev inequalities. Comm. Math. Phys., 115(4):553–569, 1988.

[78] T. Homem-de Mello. Variable-sample methods and simulated annealing for discrete stochastic optimization. 2000.

[79] R. Horst and P. Pardalos. Handbook of global optimization, volume 2. Springer Science & Business Media, 2013.

[80] H. Hotelling. Analysis of a Complex of Statistical Variables Into Principal Components. Warwick & York, 1933.

[81] S. Huckemann, T. Hotz, and A. Munk. Intrinsic shape analysis: geodesic PCA for Riemannian manifolds modulo isometric Lie group actions. Statist. Sinica, 20(1):1–58, 2010.

[82] N. Ikeda and S. Watanabe. Stochastic Differential Equations and Diffusion Processes. North-Holland, 1981.

[83] M. O. Jackson. Social and Economic Networks. Princeton Univ. Press, Princeton, NJ, 2008.

[84] S. Janson. One, two and three times log n/n for paths in a complete graph with random weights. Combinatorics, Probability and Computing, 8(4):347–361, 1999.

[85] I. T. Jolliffe. Principal component analysis. Springer Series in Statistics. Springer-Verlag, New York, second edition, 2002.

[86] D. Jones, C. Perttunen, and B. Stuckman. Lipschitzian optimization without the lipschitz constant. Journal of Optimization Theory and Applications, 79(1):157–181, 1993.

[87] D. Jones, M. Schonlau, and W. Welch. Efficient global optimization of expensive black-box functions. Journal of Global optimization, 13(4):455–492, 1998.

[88] S. Kirkpatrick and M.P. et al. Vecchi. Optimization by simmulated annealing. science, 220(4598):671–680, 1983.

[89] O. Klopp, A.B. Tsybakov, and N. Verzelen. Oracle inequalities for network models and sparse graphon estimation. Annals of Statistics, in press, 2016.

[90] E. D. Kolaczyk. Statistical analysis of network data: methods and models. Springer Series in Statistics. Springer New-York, 2009.

[91] S. Kullback. A lower bound for discrimination informat ion in terms of variation. IEEE Trans. Inform. Theory, 4:126–127, 1967.

[92] H. Le. Locating Fréchet means with application to shape spaces. Adv. Appl. Probab., 33:324–338, 2001.

[93] Huiling Le. Estimation of Riemannian barycentres. LMS J. Comput. Math., 7:193–200, 2004.

[94] J. Little, K. Murty, D. Sweeney, and C. Karel. An algorithm for the traveling salesman problem. Operations research, 11(6):972–989, 1963.

[95] L. Lovasz. Large networks and graph limits, volume 60. American Mathematical Society, 2012.

[96] L. Miclo. Recuit simulé sur $\mathbf{R}^n$. Étude de l'évolution de l'énergie libre. Ann. Inst. H. Poincaré Probab. Statist., 28(2):235–266, 1992.

[97] S. Milgram. The small world problem. Psychology Today, 1967.

[98] E. Miller, M. Owen, and J. Provan. Polyhedral computational geometry for averaging metric phylogenetic trees. Advances in Applied Mathematics, 68:51–91, 2015.

[99] P. Monmarché. Piecewise deterministic simulated annealing. ALEA, 13(1):357–398, 2016.

[100] J. Moreno. Who Shall Survive? Nervous and Mental Disease Publishing Company, 1934.

[101] E. Munch, K. Turner, P. Bendich, S. Mukherjee, J. Mattingly, and J. Harer. Probabilistic Fréchet means for time varying persistence diagrams. Electronic Journal of Statistics, 9:1173–1204, 2015.

[102] R. Munos. From bandits to monte-carlo tree search: The optimistic principle applied to optimization and planning. Foundations and Trends® in Machine Learning, 2014.

[103] A. Nemirovski and D-B Yudin. Problem complexity and method efficiency in optimization. John Wiley & Sons, Inc., Panstwowe Wydawnictwo Naukowe (PWN), 1982.

[104] M. Newman. Networks, An Introduction. Oxford University Press, 2010.

[105] M. E. Newman. Fast algorithm for detecting community structure in networks. Physical Review, E 69(066133), 2004.

[106] M E Newman. Modularity and community structure in networks. Proc Natl Acad Sci U S A, 103(23):8577–8582, June 2006.

[107] M. E. Newman and M. Girvan. Finding and evaluating community structure in networks. Physical Review, E 69(026113), 2004.

[108] P.M. Pardalos and S.A. Vavasis. Quadratic programming with one negative eigenvalue is NP-hard. J. Global Optim., 1(1):15–22, 1991.

[109] R. Pastor-Satorras and A. Vespignani. Evolution and structure of the internet : A statistical physics approach. Cambridge University Press, 2007.

[110] K. Pearson. On lines and planes of closest fit to systems of points in space. Philosophical Magazine, 2(6):559–572, 1901.

[111] X. Pennec. Intrinsic statistics on Riemannian manifolds: basic tools for geometric measurements. J. Math. Imaging Vision, 25:127–154, 2006.

[112] M.S. Pinsker. Information and information stability of random variables and processes. San Francisco, Holden-Day, originally published in Russian in 1960, 1964.

[113] H. Robbins and S. Monro. A stochastic approximation method. Ann. Math. Statistics, 22:400–407, 1951.

[114] S. Sahni. Computationally related problems. SIAM J. Comput., 3:262–279, 1974.

[115] F. S Samaria and A. C Harter. Parameterisation of a stochastic model for human face identification. In Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on, pages 138–142. IEEE, 1994.

[116] S. E. Schaeffer. Graph clustering. Computer science review, 1(1):27–64, 2007.

[117] B. Shneiderman and A. Aris. Network visualization by semantic substrates. IEEE Transactions on Visualization and Computer Graphics, 12:733–740, 2006.

[118] J. Spoerhase and H.-C. Wirth. $(r, p)$-centroid problems on paths and trees. Theoret. Comput. Sci., 410(47-49):5128–5137, 2009.

[119] A. Trouvé. Parallélisation massive du recuit simulé. PhD Thesis, Université d'Orsay, 1993.

[120] Remco van der Hofstad. Random Graphs and Complex Networks Volume One. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2016.

[121] Stijn van Dongen. Graph Clustering by Flow Simulation. Utrecht, May 2000. http://www.library.uu.nl/digiarchief/dip/diss/1895620/inhoud.htm.

[122] S. Wold, K. Esbensen, and P. Geladi. Principal component analysis. Chemometrics and intelligent laboratory systems, 2(1-3):37–52, 1987.

[123] M. Woodroofe. Normal approximation and large deviations for the robbins-monro process. Probability Theory and Related Fields, 21(4):329–338, 1972.

[124] H. Ziezold. On expected figures and a strong law of large numbers for random elements in quasi-metric spaces. In Transactions of the Seventh Prague Conference on Information Theory, Statistical Decision Functions, Random Processes and of the 1974 European Meeting of Statisticians, pages 591–602. Springer, 1977.

Stochastic algorithms for optimization under uncertainty on complex structures.
Convergence and applications.

The main topics of this thesis involve the development of **stochastic algorithms** for **optimization under uncertainty**, the study of their theoretical properties and applications. The proposed algorithms are modified versions of **simulated annealing** that use only unbiased estimators of the cost function. We study their convergence using the tools developed in the theory of **Markov processes**: we use properties of infinitesimal generators and functional inequalities to measure the distance between their probability law and a target one.

The first part is concerned with **quantum graphs** endowed with a probability measure on their vertex set. Quantum graphs are continuous versions of undirected weighted graphs. The starting point of the present work was the question of finding **Fréchet means** on such a graph. The Fréchet mean is an extension of the Euclidean mean to general metric spaces and is defined as an element that minimizes the sum of weighted square distances to all vertices. Our method relies on a Langevin formulation of a noisy simulated annealing dealt with using **homogenization**. In order to establish the convergence in probability of the process, we study the evolution of the **relative entropy** of its law with respect to a convenient Gibbs measure. Using **functional inequalities** (Poincare and Sobolev) and Gronwall's Lemma, we then show that the relative entropy goes to zero. We test our method on some real data sets and propose an heuristic method to adapt the algorithm to huge graphs, using a preliminary clustering.
In the same framework, we introduce a definition of principal component analysis for quantum graphs. This implies, once more, a stochastic optimization problem, this time on the space of the graph's geodesics. We suggest an algorithm for finding the **first principal component** and conjecture the convergence of the associated Markov process to the wanted set.

On the second part, we propose a modified version of the simulated annealing algorithm for solving a stochastic global optimization problem on a finite space. Our approach is inspired by the general field of **Monte Carlo methods** and relies on a Markov chain whose probability transition at each step is defined with the help of mini batches of increasing (random) size. We prove the algorithm's **convergence in probability** towards the optimal set, provide **convergence rate** and its optimized parametrization to ensure a minimal number of evaluations for a given accuracy and a confidence level close to 1. This work is completed with a set of **numerical experiments** and the assessment of the practical performance both on benchmark test cases and on real world examples.

# Algorithmes stochastiques d'optimisation sous incertitude sur des structures complexes. Convergence et applications.

Les principaux sujets étudiés dans cette thèse concernent le développement **d'algorithmes stochastiques** d'**optimisation sous incertitude**, l'étude de leurs propriétés théoriques et leurs applications. Les algorithmes proposés sont des variantes du **recuit simulé** qui n'utilisent que des estimations sans biais de la fonction de coût. On étudie leur convergence en utilisant des outils développés dans la théorie des **processus de Markov** : on utilise les propriétés du générateur infinitésimal et des inégalités fonctionnelles pour mesurer la distance entre leur distribution et une distribution cible.

La première partie est dédiée aux **graphes quantiques**, munis d'une mesure de probabilité sur l'ensemble des sommets. Les graphes quantiques sont des versions continues de graphes pondérés non-orientés. Le point de départ de cette thèse a été de trouver la **moyenne de Fréchet** de tels graphes. La moyenne de Fréchet est une extension aux espaces métriques de la moyenne euclidienne et est définie comme étant le point qui minimise la somme des carrés des distances pondérées à tous les sommets. Notre méthode est basée sur une formulation de Langevin d'un recuit simulé bruité et utilise une technique d'**homogénéisation**. Dans le but d'établir la convergence en probabilité du processus, on étudie l'évolution de l'**entropie relative** de sa loi par rapport a une mesure de Gibbs bien choisie. En utilisant **des inégalités fonctionnelles** (Poincaré et Sobolev) et le lemme de Gronwall, on montre ensuite que l'entropie relative tend vers zéro. Notre méthode est testée sur des données réelles et nous proposons une méthode heuristique pour adapter l'algorithme à de très grands graphes, en utilisant un clustering préliminaire. Dans le même cadre, on introduit une définition d'analyse en composantes principales pour un graphe quantique. Ceci implique, une fois de plus, un problème d'optimisation stochastique, cette fois-ci sur l'espace des géodésiques du graphe. Nous présentons un algorithme pour trouver la **première composante principale** et conjecturons la convergence du processus de Markov associé vers l'ensemble voulu.

Dans une deuxième partie, on propose une version modifiée de l'algorithme du recuit simulé pour résoudre un problème d'optimisation stochastique global sur un espace d'états fini. Notre approche est inspirée du domaine général des **méthodes Monte-Carlo** et repose sur une chaine de Markov dont la probabilité de transition à chaque étape est définie à l'aide de « mini-lots » de taille croissante (aléatoire). On montre la **convergence en probabilité** de l'algorithme vers l'ensemble optimal, on donne la **vitesse de convergence** et un choix de paramètres optimisés pour assurer un nombre minimal d'évaluations pour une précision donnée et un intervalle de confiance proche de 1. Ce travail est complété par un ensemble de **simulations numériques** qui illustrent la performance pratique de notre algorithme à la fois sur des fonctions tests et sur des données réelles issues de cas concrets.