

Comment sortir d'un labyrinthe?

Damien Garreau et Floriane Dardard

Table des matières

1	Introduction	3
2	Marches aléatoires	4
3	Graphes expandeurs	7
3.1	Expansion des arêtes et définition combinatoire des expandeurs	7
3.2	Exemples	8
3.3	Spectre et définition algébrique de l'expansion	9
3.4	Le lemme de mélange	9
3.5	Quelle taille peut atteindre le trou spectral?	12
4	Marches aléatoires sur les graphes expandeurs	13
4.1	Mélange rapide de marches aléatoires	13
4.1.1	Convergence en norme l_1 et l_2	15
5	Produit zig-zag	17
5.1	Représentation	17
5.2	Élévation à la puissance	18
5.3	Principe du produit zig-zag	19
5.4	Théorème zig-zag	22
6	USTCON en espace logarithmique	23
6.1	Problème et objectifs	23
6.1.1	USTCON	23
6.1.2	Objectif concret	23
6.2	L'algorithme de Reingold	24
6.2.1	Les étapes	24
6.2.2	Étape 1	25

6.2.3	Etape 2	25
6.2.4	Etape 3	29
6.2.5	Résultats obtenus	30
6.3	Suites de traversée déterministes	31
7	Conclusion générale	33

Chapitre 1

Introduction

Le but de cet exposé est de répondre à une question précise : comment sortir d'un labyrinthe ? L'objet mathématique de prédilection de cet exposé sera donc le graphe, à chaque intersection du labyrinthe correspondant un sommet du graphe et à chaque chemin direct entre deux intersections correspondant une arête. Sortir du labyrinthe reviendra alors à résoudre un problème de connectivité entre deux sommets ou arêtes, et bien sûr à exhiber un tel chemin reliant les deux, s'il existe. On montrera alors qu'une marche aléatoire sur le graphe sort en temps polynomial.

Par la suite, on essaiera de dérandomiser la méthode, c'est-à-dire de trouver un algorithme ne dépendant pas du hasard. Pour cela, on introduira les notions de graphe expasseur et de produit zig-zag grâce auxquelles on produira un tel algorithme. Algorithme qui vérifiera de bonnes propriétés de complexité.

Chapitre 2

Marches aléatoires

Dans cette partie, on traite l'approche marche aléatoire du problème, sur un graphe quelconque.

Soit G un graphe connexe non-orienté, n son nombre de sommets et e son nombre d'arêtes. On pose $d(i)$ le nombre d'arêtes incidentes en i et $\Delta(i, j)$ la distance entre les sommets i et j .

Une marche aléatoire sur ce graphe est équivalente à une chaîne de Markov dont les états sont les sommets de G et dont la matrice est celle des $P_{ij} = \begin{cases} 0 & \text{si } i, j \text{ n'est pas une arête} \\ \frac{1}{d(i)} & \text{sinon .} \end{cases}$

On pose aussi $T(i, j)$ le nombre moyen de transitions pour aller de i à j .

On commence par une série de lemmes concernant les temps de transition moyens :

Lemme 2.0.1 $T(i, i) = \frac{2e}{d(i)}$.

Preuve : Soit T la matrice carrée de taille n telle que $T_{ii} = 0$ et sinon T_{ij} est l'espérance du nombre de pas nécessaires pour atteindre j quand on part de i . Soit D la matrice diagonale dont les coefficients diagonaux sont les T_{ii} , égaux à l'espérance du nombre de pas nécessaire pour arriver en i quand on part de i et que le premier pas a conduit ailleurs qu'en i . Soit P la matrice des probabilités de transition ($P_{ij} = \frac{1}{d(i)}$ si j est voisin de i , 0 sinon). Soit I la matrice identité. Soit E la matrice carrée de taille n telle que $E_{ij} = 1$ pour tout i et j . Alors

$$(I - P)T + D = E.$$

En effet, par exemple, si i est différent de j , $T_{ij} = 1 + \sum_k P_{ik} T_{kj}$.

D'autre part, la mesure de probabilité stationnaire π (vue comme un vecteur ligne) satisfait par définition $\pi = \pi P$, i.e. $\pi(I - P) = 0$. Comme $\pi E = L$ est le vecteur ligne avec que des 1, cela nous donne $\pi D = L$ et donc pour tout i , $\pi_i T_{ii} = 1$.

Lemme 2.0.2 *Si i et j sont adjacents dans G , alors $T(i, j) + T(j, i) \leq 2e$ avec égalité si et seulement si i, j est une arête qui sépare le graphe en deux composantes connexes si on la supprime.*

Preuve : admis.

Lemme 2.0.3 *Pour tout i, j , $T(i, j) + T(j, i) \leq 2e$*

Preuve : Récurrence sur $\Delta(i, j)$ et lemme précédent.

On note $T(i, \cdot)$ le temps de traversée moyen (nombre moyen de transitions pour visiter tout G en partant de i).

Théorème 2.0.4 *Pour tout i , $T(i, \cdot) \leq 2e(n - 1)$.*

Démonstration : Soit H un arbre couvrant de G (sous-graphe connexe sans boucles contenant tous les sommets). Alors il existe une marche de i à i passant par chaque arête de H exactement une fois dans chaque direction. Soit $i_0, i_1, \dots, i_{2n-2}$ les sommets visités avec $i = i_0 = i_{2n-2}$. Alors :

$$\begin{aligned} T(i, \cdot) &\leq T(i_0, i_1) + T(i_1, i_2) + \dots + T(i_{2n-3}, i_{2n-2}) \\ &= \sum_{i, j \in H} T(i, j) + T(j, i) \\ &\leq 2e(n - 1) \end{aligned}$$

la dernière égalité découlant du lemme 2.

On suppose maintenant G d -régulier (tout sommet est de degré d), ce qui est toujours rendu possible par une transformation simple. On appelle $\sigma \in \{0, 1, \dots, (d - 1)\}^N$ *séquence universelle* si en partant de tout sommet v et en suivant les directions de σ , on visite tous les sommets de G . Comme le graphe est connexe, l'existence d'une telle séquence garantit la sortie effective du labyrinthe. Le théorème suivant donne même une estimation de sa longueur :

Théorème 2.0.5 *Il existe une séquence universelle de longueur $O(n^3 \log n)$ (la constante ne dépend que du degré d , fixé).*

Démonstration : Soit $\tilde{\sigma}$ une séquence aléatoire de longueur $2dn(n-1)(dn+2)\lceil \log_2 n \rceil$ ($\lceil \cdot \rceil$ = partie entière). Soit G un graphe d -régulier non-orienté, et $v \in G$. On définit une variable aléatoire $\tilde{\chi}_{G,v}$ à valeurs dans $\{0, 1\}$ comme suit. $\tilde{\chi}_{G,v} = 1$ si et seulement si en partant de V et en suivant $\tilde{\sigma}$ on n'arrive pas à visiter tous les sommets de G . Soit $\tilde{Y} = \sum_{G,v} \tilde{\chi}_{G,v}$. Par construction, si $\tilde{Y} = 0$ alors $\tilde{\sigma}$ est une séquence universelle. On va montrer que $E[\tilde{Y}] < 1$ ce qui implique clairement l'existence d'une telle séquence. Par linéarité, $E[\tilde{Y}] = \sum_{G,v} E[\tilde{\chi}_{G,v}]$. Montrons maintenant que

$$\forall G, \forall v, \quad E[\tilde{\chi}_{G,v}] \leq 2^{-(dn+2)\lceil \log_2 n \rceil} \leq n^{-dn+2}.$$

Considérons $\tilde{\sigma}$ comme la concaténation des séquences aléatoires $s_1, s_2, \dots, s_{(dn+2)\lceil \log_2 n \rceil}$, chacune de longueur $2dn(n-1)$. D'après le théorème précédent, $T(v, \cdot) \leq dn(n-1)$ ($2e = dn$ car G est de degré constant). L'inégalité de Markov nous donne :

$$P(s_1 \text{ traverse tous les sommets de } G \text{ à partir de } v) \geq \frac{1}{2}$$

et de même pour s_2 en partant de w , sommet atteint après s_1 et pour tous les s_i . On en déduit que la proba qu'aucun des s_i ne visite G est inférieure à

$$\left(\frac{1}{2}\right)^{(dn+2)\lceil \log_2 n \rceil} \leq n^{-(dn+2)}.$$

Donc $E[\tilde{\chi}_{G,v}] \leq n^{-dn+2}$. Comme le nombre de graphes considérés est inférieur à n^{dn+1} , on a :

$$E[\tilde{Y}] \leq n^{dn+1} \times n^{-dn+2} < 1$$

Notre problème de départ est donc partiellement résolu. On veut maintenant une solution ne faisant pas intervenir le hasard. On commence par introduire une classe de graphes ayant de bonnes propriétés : les graphes *expandeurs*.

Chapitre 3

Graphes expandeurs

Dans cette partie, on introduit le concept de graphe expandeur.

3.1 Expansion des arêtes et définition combinatoire des expandeurs

Introduisons quelques conventions. Sauf mention contraire, un graphe $G = (V, E)$ est non-orienté et d -régulier (tous les sommets ont le même degré d). Les boucles et arêtes multiples sont autorisées. Le nombre de sommets $|V|$ est noté n . Pour $S, T \subset V$, on pose $E(S, T) = \{(u, v), u \in S, v \in T, (u, v) \in E\}$ l'ensemble des arêtes de S à T . Ici, nous considérons chaque arête non-orientée comme une paire d'arêtes orientées, ce qui fait de $E(S, T)$ un ensemble d'arêtes **orientées**. On posera aussi $E(S) = E(S, S)$.

Définition 1 1. La frontière d'un ensemble S , notée ∂S , est l'ensemble $E(S, \bar{S})$ des arêtes partant de S dans son complémentaire. (Rappelons qu'on considère E comme un ensemble d'arêtes orientées).

2. Le facteur d'expansion de G , noté $h(G)$, est défini par :

$$h(G) = \min_{\{S, |S| \leq \frac{n}{2}\}} \frac{|\partial S|}{|S|}$$

Définition 2 Une suite de graphes d -réguliers $(G_i)_{i \in \mathbb{N}}$ de taille croissante avec i est une famille de graphes expandeurs s'il existe $\epsilon > 0$ tel que $h(G_i) \geq \epsilon$ pour tout i .

Les problèmes concernant la construction explicite d'objets mathématiques sont fondamentaux en informatique, et les graphes expandeurs n'échappent pas à la règle. Il y a deux niveaux d'efficacité naturels à atteindre dans cette construction. Le premier est la construction *ex-nihilo* d'un graphe expandeur à n sommets en temps polynomial en n . Le second, plus exigeant, demande la détermination du voisinage de chaque sommet en temps polynomial en la longueur de l'étiquette des sommets (usuellement polynomial en $\log n$). La technicité de ces définitions peut paraître étrange au lecteur non initié, mais elles ne sont que le reflet d'un besoin naturel. Les graphes expandeurs sont utilisés par divers algorithmes. Les performances de ces algorithmes dépendront de notre capacité à obtenir rapidement des informations significatives sur les graphes expandeurs utilisés.

Définition 3 Soit $(G_i)_i$ une famille de graphes expandeurs d -réguliers, G_i ayant n_i sommets et (n_i) croissante (mais pas trop vite : $n_{i+1} \leq n_i^2$ convient par exemple).

1. La famille est dite **faiblement explicite** s'il existe un algorithme qui génère le j -ème graphe de la famille en temps polynomial en j .
2. La famille est dite **fortement explicite** s'il existe un algorithme qui, à la donnée de $i \in \mathbb{N}$, $v \in V(G_i)$ et $k \in \{1, \dots, d\}$, renvoie le k -ème voisin de v dans le graphe G_i en temps polynomial en la longueur de l'entrée.

3.2 Exemples

1. *Une famille de graphes 8-réguliers* L'ensemble des sommets est $V_m = \mathbb{Z}_m \times \mathbb{Z}_m$ pour tout m et les voisins de (x, y) sont $(x + y, y)$, $(x - y, y)$, $(x, x + y)$, $(x, y - x)$, $(x + y + 1, y)$, $(x - y + 1, y)$, $(x, y + x + 1)$ et $(x, y - x + 1)$ (toutes les opérations sont modulo m). La preuve de l'expansion due à Margulis repose sur la théorie des représentations. C'est une famille fortement explicite.
2. *Une famille de graphes 3-réguliers* Pour tout nombre premier p , on pose $V_p = \mathbb{Z}_p$ l'ensemble des sommets, et si x est un sommet, ses voisins sont par définition x^{-1} , $x + 1$ et $x - 1$. Les opérations sont modulo p et on pose 0 comme inverse de 0. La preuve de l'expansion repose sur un résultat de théorie des nombres, le théorème 3/16 de Selberg.

Cette famille est faiblement explicite car on est incapable de générer de grands nombres premiers de manière déterministe.

3.3 Spectre et définition algébrique de l'expansion

La matrice d'adjacence d'un graphe à n sommets G , notée $A(G)$, est une matrice carrée de taille n telle que le coefficient (u, v) soit le nombre d'arêtes de G reliant u à v . La matrice A est symétrique réelle (le graphe est non-orienté), et a donc n valeurs propres que nous désignerons par $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$. Nous pouvons aussi y associer un système orthonormal de vecteurs propres v_1, \dots, v_n vérifiant $Av_i = \lambda_i v_i$. Nous ferons souvent référence aux valeurs propres de $A(G)$ comme le **spectre** du graphe G . Le spectre d'un graphe contient beaucoup d'informations sur celui-ci. Voici quelques illustrations de ce fait pour un graphe d -régulier :

- $\lambda_1 = d$, et le vecteur propre correspondant est $v_1 = (\frac{1}{\sqrt{n}}, \dots, \frac{1}{\sqrt{n}}) = \frac{\mathbf{1}}{\sqrt{n}}$;
- le graphe est connexe ssi $\lambda_1 > \lambda_2$;
- le graphe est bipartit ssi $\lambda_1 = -\lambda_n$.

Comme le montre le théorème suivant, la deuxième valeur propre du graphe est reliée de façon intime à son paramètre d'expansion.

Théorème 3.3.1 *Soit G un graphe d -régulier de spectre $\lambda_1 \geq \dots \geq \lambda_n$. Alors :*

$$\frac{d - \lambda_2}{2} \leq h(G) \leq \sqrt{2d(d - \lambda_2)}.$$

Plus concrètement, nous verrons que $d - \lambda_2$, aussi appelé **trou spectral**, fournit une estimation sur l'expansion d'un graphe. En particulier, un graphe d -régulier a un facteur d'expansion $h(G)$ strictement positif ssi son trou spectral l'est aussi. Le lemme suivant montre que si le trou spectral d'un graphe est important, ses arêtes sont « déployées », une caractéristique des graphes aléatoires.

3.4 Le lemme de mélange

Soit G un graphe d -régulier à n sommets. Nous noterons $\lambda(G) = \max(|\lambda_2|, |\lambda_n|)$.

Lemme 3.4.1 *Soit G un graphe d -régulier à n sommets et soit $\lambda = \lambda(G)$. Alors pour tout $S, T \subseteq V$:*

$$\left| |E(S, T)| - \frac{d|S||T|}{n} \right| \leq \lambda \sqrt{|S||T|}.$$

Quelques commentaires sur ce résultat : le membre de gauche mesure la déviation entre deux quantités, le nombre d'arêtes entre deux ensembles et le nombre attendu d'arêtes entre ces deux ensembles dans un graphe aléatoire de densité d'arêtes $\frac{d}{n}$. Un petit λ (ou plutôt un large trou spectral) implique que cette déviation est petite, et donc que le graphe est presque aléatoire en ce sens.

Quand le trou spectral de G est beaucoup plus petit que d , les bornes supérieures et inférieures diffèrent sensiblement dans le théorème précédent. Cela nous amène à nous demander si le trou spectral ne peut être encadré plus précisément par des invariants combinatoires du graphe. Une réponse positive, et une variante du lemme de mélange a été prouvée récemment :

Lemme 3.4.2 *Soit G un graphe d -régulier. On suppose que*

$$\left| |E(S, T)| - \frac{d|S||T|}{n} \right| \leq \rho \sqrt{|S||T|}$$

pour tous sous-ensembles disjoints S et T et pour un ρ positif ou nul. Alors $\lambda \leq O(\rho(1 + \log(\frac{d}{\rho})))$. La borne est optimale.

Preuve du lemme de mélange : Soient 1_S et 1_T les vecteurs caractéristiques de S et T (la v -ième coordonnée du vecteur vaut 1 si $v \in S$ et 0 sinon). Exprimons ces vecteurs dans la base orthonormale de vecteurs propres : $1_S = \sum_i \alpha_i v_i$ et $1_T = \sum_j \beta_j v_j$. Rappelons que $v_1 = \frac{1}{\sqrt{n}}$. Alors :

$$|E(S, T)| = 1_S A 1_T = \left(\sum_i \alpha_i v_i \right) A \left(\sum_j \beta_j v_j \right).$$

Comme les v_i sont des vecteurs propres orthonormaux, cela vaut $\sum_i \lambda_i \alpha_i \beta_i$. De plus, $\alpha_1 = \langle 1_S, \frac{1}{\sqrt{n}} \rangle = \frac{|S|}{\sqrt{n}}$, $\beta_1 = \frac{|T|}{\sqrt{n}}$ et $\lambda_1 = d$. Donc :

$$|E(S, T)| = d \frac{|S||T|}{n} + \sum_{i=2}^n \lambda_i \alpha_i \beta_i.$$

Par définition de λ :

$$\left| |E(S, T)| - \frac{d|S||T|}{n} \right| = \left| \sum_{i=2}^n \lambda_i \alpha_i \beta_i \right| \leq \lambda \sum_{i=2}^n |\alpha_i \beta_i|.$$

Pour finir, par Cauchy-Schwartz,

$$\left| |E(S, T)| - \frac{d|S||T|}{n} \right| \leq \lambda \|\alpha\|_2 \|\beta\|_2 = \lambda \sqrt{|S||T|}.$$

Dans ce qui suit, il est parfois pratique de considérer la seconde valeur propre normalisée $\frac{\lambda(G)}{d}$. Un graphe d -régulier G à n sommets est appelé un (n, d) -graphe. Si $\lambda(G) \leq \lambda d$, c'est un (n, d, λ) -graphe. Les graphes réguliers avec un petit α ont un certain nombre de propriétés intéressantes, par exemple :

- un **ensemble indépendant** d'un graphe $G = (V, E)$ est un ensemble de sommets tel que deux d'entre eux ne soient jamais adjacents, i.e. $|E(S)| = 0$. Une conséquence immédiate du lemme d'expansion est qu'un ensemble indépendant d'un (n, d, α) -graphe est de cardinal au plus αn .
- un k -**coloriage** d'un graphe est une application $c : V \rightarrow \{1, \dots, k\}$ telle que $c(x) \neq c(y)$ pour tout sommets adjacents x et y . Le **nombre chromatique** de G , noté $\chi(G)$, est le plus petit k pour lequel G ait un k -coloriage. Comme $c^{-1}(j)$ est un ensemble indépendant de G pour tout $j \leq k$, on a $\chi(G) \geq \frac{1}{\alpha}$ pour tout (n, d, α) -graphe.
- la distance $d_G(x, y)$ entre les sommets x et y dans un graphe $G = (V, E)$ est la longueur du plus court chemin entre les deux. Le **diamètre** de G est défini par $D = \max_{x, y} d_G(x, y)$. On définit aussi $B(x, r) = \{y, d_G(x, y) \leq r\}$ la boule de centre x et de rayon r . Alors tout (n, d, α) -graphe a un diamètre en $O(\log n)$. Cela vient du fait que $|B(x, r)| > \frac{n}{2}$ pour tout sommet x et un $r \leq O(\log n)$, ce qu'on montre grâce aux propriétés d'expansion de G . En fait, nous montrons que $|B(x, r+1)| \geq (1+\epsilon)|B(x, r)|$ pour un $\epsilon > 0$ fixé, tant que $|B(x, r)| \leq \frac{n}{2}$. Le lemme de mélange montre que $\frac{|E(S, S)|}{|S|} \leq d(\frac{|S|}{n} + \alpha)$ pour tout ensemble S . Donc $\frac{|E(S, \bar{S})|}{|S|} \geq d((1-\alpha) - \frac{|S|}{n})$. Mais S a au moins $\frac{|E(S, \bar{S})|}{d}$ voisins en dehors de lui-même, donc en prenant $\epsilon = \frac{1}{2} - \alpha$, le résultat en découle. Un graphe expenseur a un diamètre logarithmique.

3.5 Quelle taille peut atteindre le trou spectral ?

Dans ce paragraphe, on fixe d et on fait tendre n vers l'infini. Les choses sont plus compliquées quand d varie avec n .

Théorème 3.5.1 (Alon-Boppana) *Pour tout (n, d) -graphe :*

$$\lambda \geq 2\sqrt{d-1} - o_n(1),$$

où $o_n \rightarrow 0$ pour tout d fixé.

On admet ce théorème. Voici un énoncé plus faible et beaucoup plus simple à montrer :

Proposition 1 *Pour tout (n, d) -graphe :*

$$\lambda \geq \sqrt{d}(1 - o_n(1)).$$

Preuve : Soit A la matrice d'adjacence de G . Il est facile de voir que $\text{trace}(A^k)$ est le nombre de marches de longueur k dans G qui démarrent et s'arrêtent sur le même sommet. En particulier, tous les coefficients diagonaux de A^2 sont supérieurs à d . (Il suffit de faire un aller-retour sur chaque arête incidente au sommet considéré). On a donc $\text{trace}(A^2) \geq nd$. D'autre part,

$$\text{trace}(A^2) = \sum_i \lambda_i^2 \leq d^2 + (n-1)\lambda^2,$$

et on a donc $\lambda^2 \geq d \frac{n-d}{n-1}$, ce qui conclut.

Chapitre 4

Marches aléatoires sur les graphes expandeurs

Dans cette partie, on met en évidence certaines propriétés des graphes expandeurs.

Une propriété clé des marches aléatoires sur un graphe expandeur est qu'elles convergent rapidement vers leurs distributions limites. Ce résultat a de nombreuses implications que nous ne pouvons que suggérer. Dans de nombreux problèmes d'informatique, pratiques ou théoriques, il est nécessaire de faire un échantillonnage sur une distribution F d'un ensemble V (généralement fini mais gigantesque). De tels problèmes sont souvent résolus par des algorithmes du type Monte-Carlo. Considérons un graphe G sur un ensemble de sommets V tel que la distribution limite d'une marche aléatoire sur G soit \mathcal{F} . Un choix intelligent pour G garantit que (i) il est possible de simuler efficacement la marche aléatoire et (ii) la distribution induite par la marche aléatoire sur le graphe converge rapidement vers \mathcal{F} . Plusieurs domaines utilisent grandement cette méthodologie : la physique statistique, l'étude combinatoire des groupes et l'optimisation combinatoire.

4.1 Mélange rapide de marches aléatoires

Une marche sur un graphe $G = (V, E)$ est une suite de sommets $v_1, v_2 \dots \in V$ telle que pour tout i , v_{i+1} soit un voisin de v_i . Quand v_{i+1} est sélectionné uniformément parmi les voisins de v_i et cela indépendamment de i , la marche

est dite aléatoire. On démarre habituellement ce processus en choisissant v_1 selon une distribution initiale π_1 sur V . Cela induit naturellement une suite de distributions de probabilité π_i sur V telle que la probabilité que $v_i = x$ soit $\pi_i(x)$ pour tout i et tout x . Il est bien connu que pour tout graphe connexe fini G , v_i converge vers une distribution limite (ou stationnaire). De plus, il est facile de voir que si G est régulier, cette distribution est la distribution uniforme.

Cette sous-section traite de la rapidité de la convergence. Il y a plusieurs manières intéressantes de voir l'écart entre π_i et la distribution limite, et nous considérerons donc plusieurs normes et mesures de l'entropie. Le point important est que pour les expanseurs, la distance à la limite se réduit notablement à chaque étape de la marche aléatoire, et que cela caractérise les expanseurs. Quantifions cette affirmation, et introduisons quelques définitions.

Rappelons qu'un (n, d) -graphe est un graphe d -régulier sur n sommets. C'est un (n, d, α) -graphe si α majore en valeur absolue $\lambda(G)$.

Un vecteur \mathbf{p} est dit vecteur distribution de probabilité si ses coordonnées sont positives et si $\sum_1^n \mathbf{p}_i = 1$. Le vecteur de probabilité qui correspond à la distribution uniforme sur $1, \dots, n$ est $\mathbf{u} = (1, \dots, 1)/n$.

Une marche aléatoire sur un graphe fini $G = (V, E)$ est un processus aléatoire en temps discret (X_0, X_1, \dots) à valeurs dans V . Le sommet X_0 est choisi selon une certaine distribution sur V et X_{i+1} uniformément parmi les voisins de X_i .

Si G est un graphe d -régulier de matrice d'adjacence A , alors sa matrice d'adjacence normalisée est $\hat{A} = \frac{1}{d}A$. Voici quelques remarques simples sur les marches aléatoires :

- Une marche aléatoire sur G est une chaîne de Markov d'ensemble d'états V et de matrice de transition \hat{A} comme on l'a remarqué dans la première partie.
- \hat{A} est réelle, symétrique et doublement stochastique (la somme des composantes selon chaque colonne ou chaque ligne est égale à 1).
- Si $\hat{\lambda}_1 \leq \hat{\lambda}_2 \leq \dots \leq \hat{\lambda}_n$ sont les valeurs propres de \hat{A} , alors $\hat{\lambda}_1 = 1$ et $\max\{|\hat{\lambda}_2|, |\hat{\lambda}_n|\} \geq \alpha$.
- Les vecteurs propres correspondants sont les mêmes que pour A .
- Considérons une expérience au cours de laquelle on tire un sommet x au hasard sur V avec le vecteur probabilité p , puis où l'on se déplace aléatoirement sur un voisin de x . Cela est équivalent à tirer au hasard un sommet sur V avec le vecteur probabilité $\hat{A}p$.

- La matrice \hat{A}^t est la matrice de transition de la chaîne de Markov définie par des marches aléatoires de longueur t sur G . En d'autres termes, $(\hat{A}^t)_{ij}$ est la probabilité qu'une marche aléatoire démarrant en i soit en j après t étapes.
- La distribution stationnaire de la marche aléatoire sur G est la distribution uniforme $\mathbf{u} = \hat{A}\mathbf{u} = \mathbf{u}\hat{A}$.

4.1.1 Convergence en norme l_1 et l_2

On définit le produit scalaire dans \mathbb{R}^n par $\langle x, y \rangle = \sum_{i=1}^n x_i y_i$. Les normes l_1, l_2 et l_∞ sont comme d'habitude :

- $\|x\|_1 = \sum_{i=1}^n |x_i|$;
- $\|x\|_2 = \sqrt{\sum_{i=1}^n x_i^2}$
- $\|x\|_\infty = \max |x_i|$.

Nous allons maintenant prouver que si G est un (n, d, α) -graphe et $\alpha < 1$, alors quelle que soit la distribution initiale \mathbf{p} , la marche aléatoire converge exponentiellement vite en norme l_1 vers sa distribution limite (la distribution uniforme). Cela découle (*via* Cauchy-Schwartz) d'une borne similaire sur la norme l_2 , qui vient du fait que la distance à la distribution uniforme se réduit d'un facteur α à chaque étape en norme l_2 .

Théorème 4.1.1 *Soit G un (n, d, α) -graphe de matrice d'adjacence normalisée \hat{A} . Alors pour tout vecteur distribution \mathbf{p} et pour tout entier t ,*

$$\|\hat{A}^t \mathbf{p} - \mathbf{u}\|_1 \leq \sqrt{n} \alpha^t.$$

Ce théorème découle de ce qui va suivre et de l'inégalité de Cauchy-Schwarz. Pourquoi utiliser la norme l_1 pour mesurer la distance entre deux distributions de probabilités p et q ? Une métrique naturelle et communément utilisée est la **distance variation totale** $\max_B |P_p(B) - P_q(B)|$, et il n'est pas très difficile de vérifier que cela vaut exactement $\frac{1}{2} \|p - q\|_1$. En d'autres mots, si la distance l_1 est petite, alors les deux distributions p et q donnent presque le même poids à **tous** les événements de l'espace de probabilité.

Théorème 4.1.2 *Soit G un (n, d, α) -graphe de matrice d'adjacence normalisée \hat{A} . Alors pour tout vecteur distribution \mathbf{p} et pour tout entier t :*

$$\|\hat{A}^t \mathbf{p} - \mathbf{u}\|_2 \leq \|\mathbf{p} - \mathbf{u}\|_2 \alpha^t \leq \sqrt{n} \alpha^t.$$

Il suffit de prouver la borne pour $t = 1$ (le rétrécissement à chaque étape et une récurrence concluent).

Lemme 4.1.3 *Pour tout vecteur de probabilité p ,*

$$\|\hat{A}\mathbf{p} - \mathbf{u}\|_2 \leq \|\mathbf{p} - \mathbf{u}\|_2 \alpha \leq \sqrt{n}\alpha.$$

Preuve : La distribution uniforme est invariante sous l'action de \hat{A} . De plus, $\mathbf{p} - \mathbf{u}$ est orthogonal à \mathbf{u} et est donc réduit d'un facteur α à chaque étape en norme l_2 . Donc

$$\|\hat{A}\mathbf{p} - \mathbf{u}\|_2 = \|\hat{A}(\mathbf{p} - \mathbf{u})\|_2 \leq \alpha\|\mathbf{p} - \mathbf{u}\|_2 \leq \alpha.$$

Chapitre 5

Produit zig-zag

On va décrire une transformation sur les graphes qui permet d'augmenter la constante d'expansion d'un graphe tout en gardant son degré constant.

5.1 Représentation

On considère la matrice d'adjacence A du graphe.

Soit $n = |V|$, on note $V = [n] = \{1, 2, \dots, n\}$. On suppose que, pour tout sommet $v \in G$, la liste des d arêtes incidentes à v est donnée. Ces arêtes sont numérotées de 1 à d .

Définition 4 Soit G un graphe d -régulier non-orienté avec n sommets. On définit la **fonction de rotation de G** $Rot_G : [n] \times [d] \rightarrow [n] \times [d]$ par :

$$Rot_G(v, i) = (w, j)$$

si la $i^{\text{ème}}$ arête incident à v mène à w , et si cette arête est la $j^{\text{ème}}$ incidente à w . (voir figure 5.1 page 18)

Définition 5 On rappelle que la matrice d'adjacence normalisée M de G est la matrice $\frac{1}{d}G$, avec G la matrice d'adjacence du graphe. On a alors la relation :

$$M_{u,v} = \frac{1}{d} |\{(i, j) \in [d]^2 : Rot_G(u, i) = (v, j)\}|$$

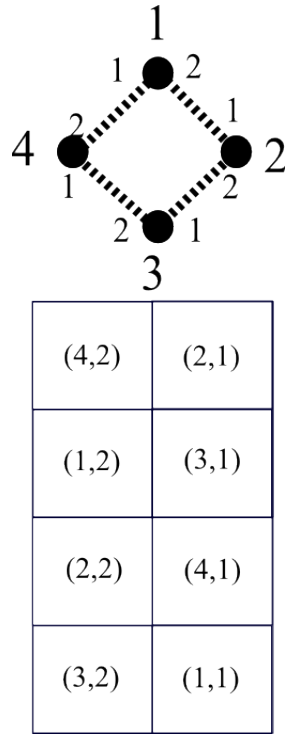


FIGURE 5.1 – Fonction de rotation d'un 4-cycle

5.2 Élévation à la puissance

Définition 6 Soit G un graphe d -régulier sur $[n]$ de fonction de rotation Rot_G . La puissance $k^{\text{ème}}$ de G est le graphe d^k -régulier sur $[n]$ dont la fonction de rotation est donnée par

$$Rot_{G^k}(v_0, (a_1, a_2, \dots, a_k)) = (v_k, (b_k, b_{k-1}, \dots, b_1)),$$

où les b_i et v_i sont calculés avec la règle : $(v_i, b_i) = Rot_G(v_{i-1}, a_i), i = 1, 2, \dots, k$.

On retrouve bien l'élévation à la puissance habituelle : la matrice d'adjacence est aussi élevée à la puissance et transcrit les chemins de longueur k . D'autre part, on sait que le caractère expenseur ne dépend que de la 2^{ème} valeur propre de la matrice d'adjacence du graphe. Il est donc évident que l'élévation à la puissance augmente la constante d'expansion du graphe. On peut voir formellement que :

Lemme 5.2.1 *Si G est un (n, d, λ) -graphe, alors G^k est un (n, d^k, λ^k) -graphe.*

5.3 Principe du produit zig-zag

Comme élever à la puissance un graphe réduit sa constante d'expansion en augmentant son degré, on a besoin d'une opération complémentaire qui réduit le degré du graphe sans trop modifier sa constante d'expansion.

On définit alors formellement le produit zig-zag comme suit :

Définition 7 *Si G est un graphe d -régulier sur $[n]$ de rotation map Rot_G et H est un graphe d -régulier sur $[d]$ de rotation map Rot_H , alors leur produit zig-zag $G \otimes H$ est défini par le graphe d^2 -régulier sur $[n] \times [d]$ dont la fonction de rotation $Rot_{G \otimes H}$ est comme suit : $Rot_{G \otimes H}((v, a), (i, j))$:*

- 1 Soit $(a', i') = Rot_H(a, i)$.
- 2 Soit $(w, b') = Rot_G(v, a')$.
- 3 Soit $(b, j') = Rot_H(b', j)$.
- 4 Renvoyer $((w, b), (j', i'))$

Une définition intuitive et plus visuelle de $G \otimes H$ est de le voir comme le graphe résultant de :

- 1) **Produit de remplacement de G et H** : Pour chaque sommet v de G , remplacer v par une copie H_v de H .
- 2) **Zig-zag** : Tracer des arêtes partout où il y a des chemins de longueur 3, la première et la troisième restant dans la même copie de H , et la seconde changeant de copie, comme en figure 2 ci-dessous. On peut le voir clairement sur la figure 5.2 page 20.

Remarque 1 *On remarque qu'intuitivement le produit zig-zag définit une correspondance entre une arête du graphe de départ, et plusieurs chemins dans le graphe d'arrivée. Un chemin dans le graphe d'arrivée peut toujours s'écrire comme l'image d'un chemin dans le graphe de départ.*

Exemple 1 *Produit zig-zag de la grille Z^2 et d'un 4-cycle :*

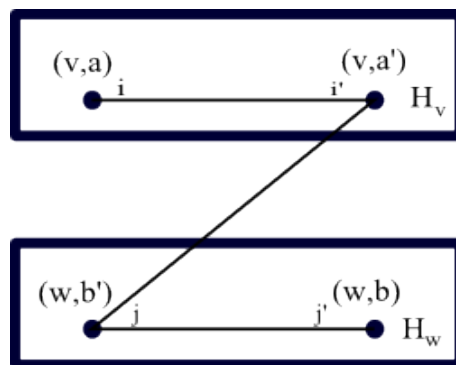
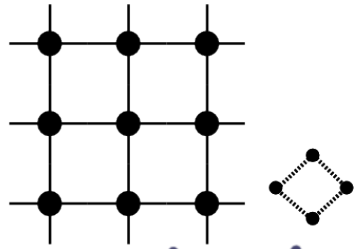
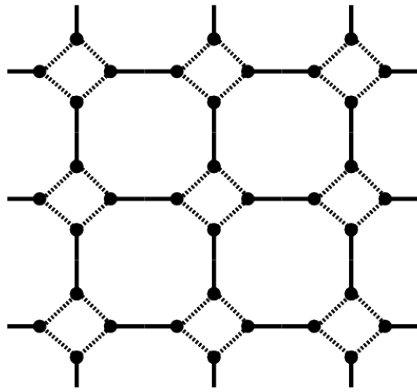


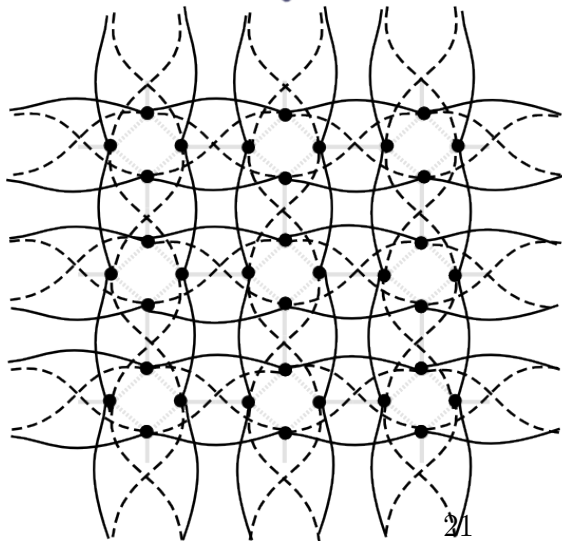
FIGURE 5.2 – Représentation d'un zig-zag



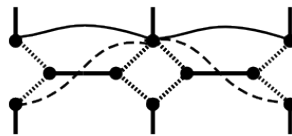
PRODUIT DE REMPLACEMENT



ZIG-ZAG



Zoom sur un point:



Le degré de $G \otimes H$ est alors d^2 , que l'on prend petit devant d , et son ensemble de sommets est $[n] \times [d]$. On réduit alors considérablement le degré de G , en multipliant le nombre de sommets par une constante. Qu'advient-il de la constante d'expansion ?

5.4 Théorème zig-zag

Théorème 5.4.1 *Soit G un (n, m, α) -graphe et H un (m, d, β) -graphe. Alors $G \otimes H$ est un $(nm, d^2, \phi(\alpha, \beta))$ -graphe où la fonction ϕ satisfait les conditions suivantes :*

- (1) *Si $\alpha < 1$ et $\beta < 1$, alors $\phi(\alpha, \beta) < 1$.*
- (2) *$\phi(\alpha, \beta) \leq \alpha + \beta$.*
- (3) *$\phi(\alpha, \beta) \leq 1 - \frac{1}{2}(1 - \beta^2) \cdot (1 - \alpha)$*

La première condition retranscrit le fait que le produit zig-zag transforme deux expandeurs en un expandeur. Les deux suivantes sont utiles pour les applications : la deuxième pour des constructions explicites d'expandeurs, comme dans l'ouvrage [4], et la troisième pour la majoration lors de l'algorithme de Reingold.

Chapitre 6

USTCON en espace logarithmique

6.1 Problème et objectifs

6.1.1 USTCON

- *Données* : $G = (V, E)$ un graphe, où V est l'ensemble des sommets et E est l'ensemble des arêtes ; $s, t \in V$.
- *Sortie* : L'existence d'un chemin entre s et t , et la description de ce chemin s'il existe.

On écrira la complexité d'un algorithme résolvant ce problème en fonction de $|V| + |E|$.

6.1.2 Objectif concret

La complexité en temps de USTCON est linéaire, en utilisant des algorithmes de parcours basique de graphe. Cependant, l'espace demandé est aussi linéaire (en $|V| + |E|$, comme on vient de le dire), ce qui n'est pas avantageux. L'objectif est de réduire l'espace utilisé à un espace logarithmique, c'est à dire, concrètement : trouver une machine de Turing

- déterministe,
- fonctionnant en temps polynomial,
- dont l'entrée (qui est bien sûr linéaire) ne peut en aucun cas être modifiée mais est libre d'accès,

- (iv) l'espace utilisé pour les opérations internes est logarithmique,
- (v) la bande de sortie est écrite sans modifications possibles.

Remarquons que la sortie, qui est le chemin entre s et t , peut être de taille linéaire!

Il est déjà possible de le faire en utilisant des algorithmes randomisés, en utilisant les résultats précédents, et l'objectif de cette partie est de dérandomiser le processus.

6.2 L'algorithme de Reingold

6.2.1 Les étapes

L'approche de Reingold pour résoudre ce problème de connectivité est d'augmenter d'abord la connectivité du graphe, puis de résoudre ce problème sur le graphe transformé. On pense naturellement aux graphes expandeurs, qui sont des graphes plutôt éparpillés, mais néanmoins hautement connectés. Il est aussi plus facile que le graphe d'arrivée soit de degré constant.

- **Étape 1** : Transformer le graphe G donné en un graphe régulier de degré constant et dont chaque composante connexe est non-bipartie (cette étape est facile).
- **Étape 2** : Transformer chaque composante connexe du graphe en un graphe expandeur en un nombre logarithmique de cycles. Nous préciserons le sens de cycle par la suite.
- **Étape 3** : Chaque composante connexe a maintenant un diamètre logarithmique. On peut alors résoudre USTCON sur le graphe transformé en énumérant les chemins de longueur logarithmique partant de s .

La difficulté de cet algorithme réside dans le fait qu'à aucun moment on ne calcule explicitement le graphe expandeur, pour des contraintes d'espace, et que l'on se contente d'énumérer des chemins de taille logarithmique.

Il existe plusieurs représentations usuelles de graphes, et comme il existe des transformations en espace logarithmique entre celles-ci, la complexité spatiale de USTCON est indépendante de la représentation choisie. On choisit ici la représentation en fonction de rotation.

6.2.2 Étape 1

Il existe plusieurs manières d'effectuer cette étape, une des plus intuitives est celle-ci : on va transformer G en un graphe régulier de degré constant de degré 5. Pour cela, on remplace chaque sommet par un nombre de sommets égal à son arité, ces sommets étant reliés en un cycle comme à la figure 6.1 . On ajoute alors un cycle autour de chacun des sommets obtenus pour éviter que le graphe ne soit biparti.

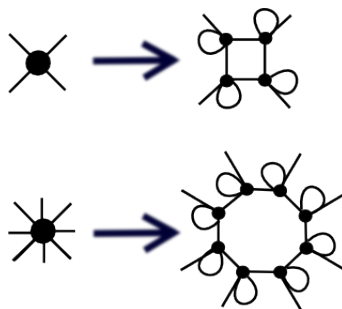


FIGURE 6.1 – Deux exemples de transformation de sommets par l'étape 1

Cette transformation multiplie les sommets de G par un facteur constant, et il reste facile d'obtenir en espace logarithmique la valeur de $Rot_{G'}(w, i)$, pour tout sommet w de G' et toute arête i partant de ce sommet, à partir de la connaissance de Rot_G . Cette transformation conserve les composantes connexes, donc ne modifie pas la réponse au problème.

6.2.3 Étape 2

Comment transformer chaque graphe en expenseur ? Il ne faut pas oublier que tout graphe peut être vu comme un graphe expenseur avec une petite constante d'expansion. Une opération naturelle pour augmenter la constante d'expansion d'un graphe est l'élévation à la puissance. La $k^{\text{ème}}$ puissance de G contient une arête entre s et t pour chaque chemin de longueur k entre s et t .

On va combiner l'élévation à la puissance avec le produit zig-zag avec H , un graphe *fixé* d -régulier tel que $\lambda(H) \leq \frac{1}{2}$. L'existence d'un tel H est admise, mais des ouvrages tels que [4] en fournissent des constructions explicites.

Algorithme de transformation

– *Données* : G graphe non biparti, d^{16} -régulier sur l'ensemble de sommets $[n]$, H graphe d -régulier sur $[d^{16}]$, tous les deux définis par leur fonction de rotation.

– *Sortie* : un graphe expandeur avec un nombre polynomial en n le nombre de sommets

On définit G_ℓ selon ce qui suit :

– On pose $\ell = 2\lceil \log dn^2 \rceil$

– On définit la suite G_n par $G_0 = G$, et pour $i > 0$,

$$G_i = (G_{i-1} \otimes H)^8.$$

On note $T_i(G, H)$ le graphe G_i , et $T(G, H) = G_\ell$.

Remarque 2 *Par les propriétés du produit zig-zag et de l'élévation à la puissance, on obtient que chaque G_i est un graphe d^{16} -régulier sur $[n] \times ([d^{16}])^i$, et donc le produit zig-zag entre G_i et H est bien défini.*

Adéquation au problème

On va montrer que cet algorithme augmente la constante d'expansion, sans augmenter le degré du graphe. On pourrait opposer à cela que l'algorithme multiplie la taille du graphe à chaque étape. Cependant, comme d est une constante, alors $\ell = O(\log n)$ et G_ℓ a $\text{poly}(n)$ sommets, ce qui est acceptable.

Montrons que T transforme G en un graphe expandeur.

Lemme 6.2.1 *Soient G et H les entrées de T . Si $\lambda(H) \leq 1/2$ et que G est connexe et non biparti, alors $\lambda(T(G, H)) \leq 1/2$.*

Preuve 1 : *Comme $G = G_0$ est connexe et non biparti, alors on a*

$\lambda(G) \leq 1 - \frac{1}{d \cdot n^2}$. *D'autre part, le choix de ℓ implique*

$$\left(1 - \frac{1}{d \cdot n^2}\right)^{2^\ell} < \frac{1}{2}.$$

Il suffit alors de montrer que pour tout $i > 0$, on a

$$\lambda(G_i) \leq \max\{\lambda(G_{i-1}), \frac{1}{2}\}.$$

Comme $\lambda(H) \leq \frac{1}{2}$, d'après le théorème zig-zag, on a $\lambda(G_{i-1} \otimes H) \leq 1 - \frac{3}{8}(1 - \lambda(G_{i-1})) < 1 - \frac{1}{3}(1 - \lambda(G_{i-1}))$.

Par définition de G_i , et par le lemme 3.2.1, on a $\lambda(G_i) < [1 - \frac{1}{3}(1 - \lambda(G_{i-1}))]^8$. On a alors

$$\lambda(G_i) < \frac{1}{2}.$$

Montrons maintenant que T conserve les composantes connexes de G . On va montrer pour cela que T agit sur chaque composante connexe, indépendamment des autres.

En effet, en notant $G|_S$, pour $S \subseteq [n]$, le sous-graphe de G induit par S , composé des sommets de S et de toutes les arêtes de G les reliant, on a le résultat :

Lemme 6.2.2 *Soit G et H les entrées de T . Si $S \subseteq [n]$ est une composante connexe de G , alors*

$$T(G|_S, H) = T(G, H)|_{S \times ([d^{16}])^\ell}.$$

Remarque 3 *Si le graphe G est non biparti connexe, alors G^8 est connexe. Autrement dit, l'élévation à la puissance conserve les composantes connexes si et seulement si G est non biparti, ce qui est le cas à la fin de l'étape 1. Le produit zig-zag préserve les composantes connexes, car il ne fait qu'augmenter la connectivité en rajoutant des chemins.*

Le problème de savoir si s et t connectés dans G est donc équivalent à savoir si s et t sont connectés dans $T(G, H)$ ¹.

Il reste à montrer que T est une transformation qui ne prend qu'un espace logarithmique :

Théorème 6.2.3 *Pour toute constante d la transformation T peut être calculée en espace $O(\log n)$ en les entrées G et H , où G est un graphe d^{16} -régulier sur $[n]$ et H est un graphe d -régulier sur $[d^{16}]$.*

1. s et t correspondent chacun à plusieurs sommets, qui sont dans la même composante connexe. On cherche à savoir, par exemple, si $(s, 1^{d^{16\ell}})$ et $(t, 1^{d^{16\ell}})$ sont connectés.

Le théorème est équivalent au lemme suivant, plus intuitif :

Lemme 6.2.4 *Pour tout v sommet de $G_\ell (\in [n] \times ([d^{16}])^\ell)$, pour toute arête $\alpha (\in [d^{16}])$, on sait calculer $Rot_{G_\ell}(v, \alpha)$ en espace logarithmique.*

En effet, si on a un algorithme A qui trouve chacune des valeurs de $Rot_{G_\ell}(v, \alpha)$, il suffit alors de les énumérer pour tout couple (v, α) . Comme $\ell = O(\log n)$ et que d est une constante, la longueur de chaque (v, α) est $O(\log n)$, alors les énumérer tous prend un espace $O(\log n)$.

Il reste donc à montrer que pour tout couple (v, α) , l'évaluation de $Rot_{G_\ell}(v, \alpha)$ ne prend que l'espace requis. Décrivons l'idée de l'algorithme qui renvoie l'évaluation de $Rot_{G_\ell}(v, \alpha)$.

Si $v \in [n]$ représente un sommet de G , et $\alpha = (a_0, a_1, \dots, a_\ell) \in [d^{16}]$, les a_n représentant alternativement des sommets de H , a_0 étant l'étiquette d'une arête issue de v , ou des suites de 16 arêtes de H , dont chacune est notée $k_{n,h}$. On aura alors $a_i = k_{i,1}, k_{i,2}, \dots, k_{i,16}$.

On veut lire les étiquettes a_0, a_1, \dots, a_ℓ à partir de v et trouver le sommet dans lequel on arrive. On va parcourir les arêtes au fur et à mesure, et modifier les valeurs des a_i en même temps (on fera attention aux effets de bord) pour obtenir directement $Rot_{G_\ell}(v, \alpha)$ sans avoir à le réécrire.

On va évaluer le résultat récursivement. A l'étape i , l'algorithme va évaluer $Rot_{G_i}(v, a_0, a_1, \dots, a_i)$.

Pour $G_0 = G$, c'est immédiat. Il suffit de lire le résultat sur Rot_G , et on peut le faire en espace logarithmique (et même constant).

Pour $i \leq 1$, on va évaluer Rot_{G_i} ainsi : Pour $j = 1$ à 16

- Remplacer $a_{i-1}, k_{i,j}$ par $Rot_H(a_{i-1}, k_{i,j})$
- Si j est impair, remplacer (étape récursive) $v, a_0, a_1, \dots, a_{i-1}$ par $Rot_{G_{i-1}}((v, a_0, a_1, \dots, a_{i-2}), a_{i-1})$
- Si $j = 16$, renverser l'ordre des arêtes dans a_i : remplacer $k_{i,1}, \dots, k_{i,16}$ par $k_{i,16}, \dots, k_{i,1}$

Ces étapes retranscrivent exactement le parcours d'une arête sur G_i , qui correspond à 8 parcours (puissance 8) de 3 arêtes zig-zag dans le graphe G_{i-1} .

Plus précisément, on peut suivre le parcours de deux boucles de j sur la figure 4, avec les flèches correspondant aux arêtes gardées en mémoire :

1. Lorsque j est impair : Pendant la première étape, on passe du point 1 au point 2 par une arête de H , et dans la deuxième étape on passe au point 3 (par une arête de G_{i-1}).
2. Lorsque j est pair : Pendant la première étape, on passe du point 3 au point 4 par une arête de H .

Lorsque $j = 16$, on a alors fait 8 fois ce cycle, on inverse les arêtes à l'intérieur de a_i pour que le parcours soit dans le bon sens.

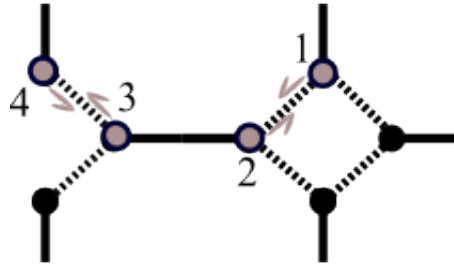


FIGURE 6.2 – Parcours pendant un cycle dans le graphe G_i

Il reste à montrer que T est une transformation qui ne prend qu'un espace logarithmique : En effet, l'évaluation de $Rot_{G_{i+1}}$ pour n'importe quel i ne requiert qu'un espace constant de plus que celui utilisé pour l'évaluation de Rot_{G_i} . L'évaluation de G_{i+1} se fait à l'aide de plusieurs étapes qui effectuent :

- un nombre constant d'opérations telles que l'appel de Rot_H ou Rot_G , ou l'inversion des étiquettes, qui se font en espace constant,
- un nombre constant d'appels récursifs de l'évaluation de Rot_{G_i} .

Donc la mémoire utilisée pour l'évaluation de Rot_{G_i} , disons M_i , suit l'équation suivante :

$$M_{i+1} = M_i + \text{constante}$$

Comme la récursion est de profondeur $\ell = O(\log n)$, on a :

$$M_\ell = O(\log n).$$

6.2.4 Etape 3

Cela repose sur le fait que le diamètre d'un graphe expandeur est logarithmique :

Théorème 6.2.5 *Soit $\lambda < 1$ une constante. Alors, pour tout (n, d, λ) -graphe G , il existe un chemin de longueur $O(\log n)$ qui relie s à t .*

On peut alors en déduire que *USTCON* dans un graphe expenseur de degré constant peut être résolu en espace logarithmique :

En effet, grâce à la connaissance de Rot_{G_ℓ} , on peut énumérer tous les chemins de taille logarithmique possibles partant d'un sommet correspondant à s (n'importe lequel) et en voyant si au moins l'un d'entre eux passe par un sommet correspondant à t (on regarde simplement si la bande de sortie renvoie ce sommet à un moment).

On arrive alors à une complexité en $O(\log n \cdot \log d)$.

Théorème 6.2.6 *Soit $\lambda < 1$ une constante. Alors il existe un algorithme A_{exp} déterministe en espace $O(\log n \cdot \log d)$ et en temps polynomial tel que si on lui donne en entrée un graphe non orienté d -régulier sur n sommets, on a :*

- (1) *Si s et t sont dans la même composante connexe, et que cette composante est un (n', d, λ) -graphe, alors A_{exp} renvoie “connectés”.*
- (2) *Si A_{exp} renvoie “connectés”, alors s et t sont effectivement dans la même composante connexe.*

6.2.5 Résultats obtenus

On a alors construit un algorithme permettant de donner une réponse à *USTCON* en temps polynomial et en espace logarithmique.

Donc $USTCON \in L$, où L est l'ensemble des problèmes décidables par une machine de Turing déterministe en espace logarithmique. Donc on obtient la propriété fondamentale qui est :

$$SL = L$$

Or, *STCON*, c'est-à-dire la résolution du même problème dans un graphe orienté est le problème universel de la classe NL , qui est l'ensemble des problèmes décidables par une machine de Turing non déterministe en espace logarithmique, tandis que *USTCON* est le problème qui correspond à la classe SL , qui est la classe des langages décidables par une machine de Turing *symétrique* en espace logarithmique.

Comme on a $SL = L \subset NL$, et que l'on ne sait pas si l'inclusion est stricte, on a le résultat concret suivant :

Il n'est pas plus facile de trouver son chemin dans un labyrinthe pour quelqu'un qui ne peut pas reculer.

Cela suit bien l'intuition courante.

6.3 Suites de traversée déterministes

On donne ici une idée de dérandomisation de suites de traversées grâce à cet algorithme. Les preuves font essentiellement appel à l'intuition.

On peut aller encore plus loin dans cet algorithme et trouver effectivement le chemin qui relie s à t en espace logarithmique. En effet, lorsqu'on a opéré des transformations sur le graphe, on pouvait à tout moment trouver le chemin dans G qui correspondait à une arête dans le graphe transformé. C'est ce qu'on a implicitement fait lors du calcul de $T(G, H)$. On peut alors construire en espace logarithmique le *chemin qui relie s à t* .

Ce chemin est donné sous forme de sommets traversés, c'est-à-dire une suite d'éléments de $[n]$, ou sous forme d'arêtes empruntées, c'est-à-dire une suite d'éléments de $[d]$. Cette deuxième forme permet de tracer un chemin équivalent depuis n'importe quel sommet, et dans n'importe quel graphe d -régulier.

En ayant comme outil un algorithme trouvant le chemin d'un sommet particulier à un autre dans un graphe particulier, est-il possible de trouver une suite de traversée déterministe qui marche pour le plus de graphes possible ?

Définition 8 *Soit π une permutation de $[d]$ et Rot_G la fonction de rotation d'un graphe d -régulier G . Alors Rot_G est π -cohérent si pour tout v, i, w et j tels que $Rot_G(v, i) = (w, j)$, alors on a $j = \pi(i)$. On dit aussi que l'étiquetage de G est π -cohérent.*

Si G a un étiquetage π -cohérent, alors l'algorithme de Reingold n'a plus besoin de la connaissance de Rot_G . En effet, dans le cycle décrit pour l'évaluation de Rot_{G_ℓ} , la seule fois où l'algorithme y fait appel est lors du dernier appel récursif, lorsqu'on cherche à remplacer (v, a_0) par $Rot_G(v, a_0)$. Comme G est π -cohérent, on peut complètement ignorer v et remplacer seulement a_0 par $\pi(a_0)$ pour retranscrire le chemin suivi.

On note B l'ensemble des chemins que l'on a tracés entre s et t , s et t décrivant l'ensemble des sommets de G . (Pour chaque couple (s, t) on garde seulement le chemin de s à t et on enlève son inverse, celui de t à s .)

Alors, en concaténant chaque chemin de B avec son inverse, et en les concaténant tous, on voit que :

Quel que soit le sommet de départ, chaque sommet de G est visité.

Comme cette suite ne dépend pas de Rot_G mais seulement de π , on a :

Lemme 6.3.1 *Il existe un algorithme utilisant un espace logarithmique qui prend en entrée 1^n et une permutation π de $[d]$ et qui renvoie une suite de traversée valable pour tous les graphes d -réguliers sur $[n]$ dont l'étiquetage est π -cohérent.*

On n'est pas totalement satisfait car la suite de traversée ne l'est pas pour tous les graphes.

Cependant, M. Koucky a montré dans [6] comment transformer une suite universelle de traversée qui marche pour les graphes avec un étiquetage π -cohérent pour une permutation π , en une suite universelle d'exploration. Une suite d'exploration étant une suite universelle de traversée avec la propriété de retour en arrière, ce qui permet une plus grande flexibilité. On peut alors déduire du théorème précédent le résultat :

Théorème 6.3.2 *Il existe un algorithme fonctionnant en espace logarithmique qui prend en entrée $(1^n, 1^d)$ et qui produit une (n, d) -suite d'exploration universelle.*

Chapitre 7

Conclusion générale

Les expandeurs et leurs propriétés remarquables avaient déjà permis de grandes avancées avec leurs applications diverses comme la construction de codes correcteurs ou de graphes de Cayley, et l'invention du produit zig-zag vient enrichir ces horizons. Il a déjà été démontré dans [7] que $RL = L$, où le R représente bien sûr le fait que la machine de Turing soit randomisée. On discute de la longueur optimale des suites d'exploration des graphes d -réguliers dans [6], et bien d'autres domaines avancent en ce moment-même grâce à ces découvertes.

Bibliographie

- [1] R. ALELIUNAS, R.M. KARP, R.J. LIPTON, L. LOVASZ, C. RACKOFF, *Random walks, universal traversal sequences, and the complexity of maze problems*. 20th Annual Symposium on Foundations of Computer Science, pp.218-223), 1979
- [2] O. REINGOLD, *Undirected Connectivity in Log-Space*. J. ACM 55,4, Article 17, September 2008
- [3] O. REINGOLD, *On Expander Graphs and Connectivity in Small Space*. Proceedings of the International Congress of Mathematicians : Madrid, August 22-30,2006
- [4] S. HOORY, N. LINIAL, A. WIDGERSON, *Expander Graphs and their Applications*. Bull. AMS 43, pp. 439-561, 2006
- [5] O. REINGOLD, S. VADHAN, A. WIDGERSON, *Entropy Waves, the Zig-Zag Graph Product and New Constant-Degree Expanders* . The Annals of Mathematics, Second Series, Vol. 155, No. 1 , pp. 157-187, January 2002
- [6] M. KOUCKY *Universal traversal sequences with backtracking*. J. Comput. System Sci. 65 (2002), 717–726
- [7] O. REINGOLD, L. TREVISAN, S. VADHAN, *Pseudorandom walks in biregular graphs and the RL vs. L problem*. Electronic Colloquium on Computational Complexity Technical Report TR05-022, 2005.

Travail effectué sous la direction de Pierre Pansu et Eric Colin de Verdière.