

Chapitre 2

Optimisation et homologie

2.1 Calcul d'une base optimale de H_1

2.1.1 Cas des graphes

Si G est connexe, la proposition 1.1.9 montre que l'on peut construire une base de $H_1(G)$ à partir d'un arbre couvrant T de G , en reliant chaque corde de T par l'unique chemin simple dans T qui joint ses extrémités. Une telle base est parfois appelée *base fondamentale des cycles* ou *base de Kirchhoff*.

Si G n'est pas connexe, il suffit de travailler indépendamment sur chacune de ses composantes connexes, puisque l'homologie est la somme directe des homologies de chaque composante. On peut d'ailleurs raffiner ce découpage pour travailler sur chaque "composante" 2-connexe (un bloc au sens de la théorie des graphes).

Exercice 2.1.1 *Montrer que l'homologie d'un graphe combinatoire est la somme directe des homologies de chacune de ses composantes 2-connexes. On pourra étudier l'application qui associe à tout cycle du graphe sa trace sur chacune des composantes 2-connexes.*

Si les arêtes de G portent des poids, ou longueurs, on peut chercher à calculer une famille de cycles génératrice de l'homologie qui soit de longueur totale minimale. On conviendra, par un léger abus de langage, d'appeler *base minimale* une telle famille. On peut également se demander s'il existe une base minimale qui soit fondamentale (i.e. associée à un arbre couvrant). Contrairement au cas de l'homotopie, la réponse à cette dernière question est généralement négative comme le montre l'exemple de la figure 2.1 tirée de Hartvigsen et Mardon [HM93]. Dans ce même article, Hartvigsen et Mardon caractérisent les graphes pour lesquels on peut toujours trouver une base minimale qui soit fondamentale, et ce quelque soient les poids associés aux arêtes. Par ailleurs, Deo, Prabhu et Krishnamoorthy [DPeK82] ont montré que le calcul d'une base fondamentale minimale est NP-complet. Cependant, le calcul d'une base minimale de *l'homologie à coefficients dans \mathbb{Z}_2* peut être effectué en temps polynomial. Nous décrivons l'algorithme proposé par Horton [Hor87]. Nous supposons donné par la suite un graphe G fini et connexe dont les arêtes sont munies de poids non-négatifs. Notons que $H_1(G, \mathbb{Z}_2)$ est fini et isomorphe

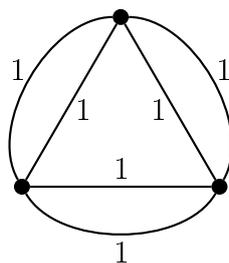


FIG. 2.1 – Tous les arbres couvrants de ce graphe sont isomorphes à une chaîne de deux arêtes. Les bases fondamentales associées se composent de deux 2-cycles et deux 3-cycles, cependant que la base minimale est formée de trois 2-cycles et un 3-cycle.

à l'ensemble des sous-graphes eulériens de G munis de la différence symétrique. Si c est un cycle de $H_1(G, \mathbb{Z}_2)$ on note $|c|$ la somme des poids de ses arêtes, appelée également *longueur* de c .

Si B est une famille de cycle de G , on note $\ell(B)$ la liste croissante des longueurs des cycles de B . Dans ce qui suit on notera toujours une base $B = (b_1, \dots, b_r)$ dans l'ordre croissant des longueurs de ses cycles : $|b_1| \leq |b_2| \leq \dots \leq |b_r|$. Le calcul d'une base minimale repose sur l'observation suivante :

Lemme 2.1.2 *Une base B de $H_1(G, \mathbb{Z}_2)$ est de longueur minimale si et seulement si $\ell(B)$ est minimale pour l'ordre lexicographique.*

Preuve : On considère une base $B = (b_1, \dots, b_r)$ qui minimise $\ell(B)$ pour l'ordre lexicographique, où r est le rang de $H_1(G, \mathbb{Z}_2)$ (c.f. définition 1.1.10). Soit $C = (c_1, \dots, c_r)$ une base quelconque de $H_1(G, \mathbb{Z}_2)$. On considère l'ensemble d'indices $I = \{i \in [1, r], |b_i| > |c_i|\}$. Montrons que I est vide et donc que C est au moins aussi longue que B , i.e. que B est minimale. Supposons par l'absurde que ce n'est pas le cas. Soit k le plus petit indice dans I . Il existe un cycle c parmi (c_1, \dots, c_k) tel que $\{b_1, \dots, b_{k-1}\} \cup \{c\}$ constitue une famille libre de rang¹ k . On peut alors compléter cette famille libre avec des éléments de $(b_k, b_{k+1}, \dots, b_r)$ pour former une base B' . Mais $|c| \leq |c_k| < |b_k|$ implique $\ell(B') < \ell(B)$ pour l'ordre lexicographique, une contradiction.

Réciproquement, soit $B = (b_1, \dots, b_r)$ une base minimale de $H_1(G, \mathbb{Z}_2)$. Montrons que $\ell(B)$ est minimal. Soit $C = (c_1, \dots, c_r)$ une base qui minimise $\ell(C)$. Soit k le plus petit indice tel que $|b_k| > |c_k|$. On forme comme précédemment une famille libre $\{b_1, \dots, b_{k-1}\} \cup \{c\}$ avec $c \in \{c_1, \dots, c_k\}$ que l'on complète en une base B' avec des éléments de $(b_k, b_{k+1}, \dots, b_r)$. Mais $|c| \leq |c_k| < |b_k|$ implique $|B'| < |B|$, une contradiction. \square

Ce lemme assure que l'algorithme "glouton" suivant renvoie une base minimale :

1. Initialiser une liste B à vide.
2. Parcourir $H_1(G, \mathbb{Z}_2)$ dans un ordre croissant des longueurs des cycles, et pour chaque cycle c parcouru, l'ajouter à B si $B \cup \{c\}$ constitue une famille libre.
3. retourner B .

¹C'est la propriété d'échange matroïdale.

$\ell(B)$ est effectivement minimale pour l'ordre lexicographique (le vérifier !). Il y a en général bien trop de cycles dans $H_1(G, \mathbb{Z}_2)$ pour que cet algorithme soit efficace. L'idée de Horton est de caractériser les cycles de toute base minimale afin de restreindre le parcours (2) à ces cycles.

Lemme 2.1.3 *Soit b un cycle d'une base B de $H_1(G, \mathbb{Z}_2)$ égal à la somme (modulo 2) de deux cycles non nuls c et d . Alors l'une des deux familles $B \setminus \{b\} \cup \{c\}$ et $B \setminus \{b\} \cup \{d\}$ est une base de $H_1(G, \mathbb{Z}_2)$.*

Preuve : On ne peut avoir à la fois c et d liés à $B \setminus \{b\}$, car b le serait. □

Corollaire 2.1.4 *Tout cycle d'une base de longueur minimale est simple.*

Preuve : Tout cycle b non-simple est la somme de deux cycles disjoints (par arêtes), i.e. on peut écrire $b = c + d$ avec $|b| = |c| + |d|$. Le lemme précédent montre que b ne peut appartenir à une base minimale. □

Lemme 2.1.5 *Soit b un cycle d'une base des cycles B de longueur minimale. Soient x et y deux sommets de b et soient p et q les deux chemins joignant x et y dans b . Alors p ou q est un plus court chemin dans G .*

Preuve : Soit t un plus court chemin joignant y et x . Appliquer le lemme 2.1.3 à la somme $b = p.t + q.t$ (avec une petite confusion abusive entre chemins et chaînes), en comparant les longueurs des chemins $p.t$ et $q.t$ à $|b|$. □

Corollaire 2.1.6 *Soit b un cycle d'une base B de longueur minimale. Pour tout sommet x de b , on a $b = p.a.q$ où a est un arc de G , et p et q sont deux plus courts chemins joignant x aux extrémités de a .*

Preuve : Le résultat est trivial si b contient un unique arc (donc une boucle). Supposons que ce n'est pas le cas et écrivons $b = (a_1, a_2, \dots, a_k)$ avec $x = o(a_1) = o(a_k^{-1})$. Soit i l'indice maximal tel que (a_1, a_2, \dots, a_i) est un plus court chemin. Alors b s'écrit sous la forme $(a_1, a_2, \dots, a_i).a_{i+1}.(a_{i+2}, \dots, a_k)$ et le lemme précédent indique que (a_{i+2}, \dots, a_k) est un plus court chemin. □

S'il y a unicité des plus courts chemins entre chaque paire de sommets, alors ce corollaire permet de restreindre le parcours (2) de l'algorithme glouton à $O(|S||A|)$ cycles – un par paire (sommet, arête). Dans le cas général, cependant, le nombre de cycles ayant la forme $p.a.q$ du corollaire peut être trop grand.

Pour chaque sommet x , on choisit un arbre de plus court chemin T_x de racine x et on définit $p(x, y)$ comme le chemin de x à y dans T_x . En particulier, si z est un sommet de $p(x, y)$ alors $p(x, z)$ est un sous-chemin de $p(x, y)$. À tout couple $(x, a) \in S \times A$ on associe le cycle $c(x, a) = p(x, o(a)).a.p(x, o(a^{-1}))^{-1}$. Le cycle $c(x, a)$ est simple si et seulement si x est l'ancêtre commun des extrémités de a dans T_x .

On suppose dans ce qui suit que les poids des arêtes sont strictement positifs. Dans le cas contraire, un prétraitement permet de s'y ramener [MM09].

Lemme 2.1.7 *Le parcours (2) de l'algorithme glouton peut être restreint aux cycles $c(x, a)$, $(x, a) \in S \times A$ qui sont simples, de sorte que la base retournée est de longueur minimale.*

Preuve : Il suffit de montrer (cf. exercice 2.1.8 suivant) qu'il existe une base minimale constituée uniquement de cycles de la forme $c(x, a)$. Rappelons que ces cycles sont nécessairement simples par le corollaire 2.1.4. Soit $B = (b_1, b_2, \dots, b_r)$ une base minimale. On associe à tout cycle b une valeur de défaut $d(b)$ obtenue comme suit : pour chaque sommet x de b on sait par le corollaire 2.1.6 que $b = p.a.q$ où a est un arc et p, q sont des plus courts chemins entre x et les extrémités de a . On compte le nombre d'arêtes de b qui ne sont pas dans $c(x, a)$ et on définit $d(b)$ comme le minimum de ce nombre sur tous les x et a définis comme ci-dessus. En particulier, b est de la forme $c(x, a)$ si et seulement si $d(b) = 0$.

Supposons que pour un certain $i \in [0, r - 1]$ les i premiers cycles de B sont de la forme $c(x, a)$. Si $d(b_{i+1}) = 0$ alors i peut être incrémenté. Sinon, on considère le couple (x, a) qui réalise $d(b_{i+1})$ et on écrit $b_{i+1} = p.p(x, y)^{-1} + c(x, a) + p(x, z).q$ où y et z sont les extrémités de a . On peut supposer sans perte de généralité que $|p| < |a.q|$, d'où $|p.p(x, y)^{-1}| < |b_{i+1}|$ et $|c(x, a)| = |p(x, z).q| = |b_{i+1}|$. On déduit du lemme 2.1.3, et du fait que B est minimale, que $B \setminus b_{i+1} \cup c(x, a)$ ou $B \setminus b_{i+1} \cup p(x, z).q$ est une base minimale. Dans le premier cas on peut incrémenter i . Dans le second, on remarque que $d(p(x, z).q) < d(b_{i+1})$, ce qui permet par récurrence sur $d(b_{i+1})$ de supposer $d(b_{i+1}) = 0$ et on peut à nouveau incrémenter i . Par récurrence sur i on peut finalement supposer que B est composée de cycles de la forme $c(x, a)$. \square

Exercice 2.1.8 *Vérifier que si l'on restreint le parcours (2) de l'algorithme glouton ci-dessus à une partie $A \subset H_1(G, \mathbb{Z}_2)$ contenant une base B de longueur minimale, alors la base retournée par l'algorithme est bien de longueur minimale.*

Exercice 2.1.9 *La preuve du lemme 2.1.7 suppose implicitement que le poids des arêtes est strictement positif. Adapter la preuve au cas où des poids peuvent être nuls.*

Exercice 2.1.10 *En toute généralité, il n'est pas nécessaire que $p(x, y)$ soit défini à partir d'un arbre de plus court chemin, i.e. que pour $z \in p(x, y)$ on ait $p(x, z) \subset p(x, y)$. Pour chaque couple (x, y) de sommets on choisit un plus court chemin quelconque $p(x, y)$ entre x et y . À tout couple $(x, a) \in S \times A$ on associe le cycle $c(x, a) = p(x, o(a)).a.p(x, o(a^{-1}))^{-1}$. Montrer à nouveau que l'algorithme glouton peut être restreint aux cycles $c(x, a)$, $(x, a) \in S \times A$ qui sont simples (cf. [Hor87]).*

Proposition 2.1.11 *Soit $G = (S, A, o, ^{-1})$ un graphe connexe fini dont les arêtes sont munies de poids (longueurs) non-négatifs. Alors on peut calculer une base de longueur minimale en temps $O(|S|^2 \log |S| + r^2 |S| |A|) = O(|S| |A|^3)$, où $r = |A| - |S| + 1$.*

Preuve : Suivant le lemme 2.1.7 On restreint l'algorithme glouton aux cycles de la forme $c(x, a)$. Pour chaque sommet x , le calcul des $c(x, a)$ peut se faire à l'aide d'un arbre de plus

courts chemins de source x en calculant pour chaque corde a de cet arbre la concaténation de a avec les chemins joignant dans l'arbre les extrémités de a à la source x . Il y a r cycles de ce type, où r est le rang de $H_1(G)$. (En fait seules les cordes dont le chemin simple associé passe par x sont à considérer). Il y a donc en tout $O(r|S|)$ cycles de taille $O(|S|)$ à considérer. Ils peuvent être calculés et stockés en temps $O(|S|(|S| \log |S| + |A|) + r|S|^2)$ en utilisant l'algorithme de Dijkstra pour le calcul de l'arbre. Leur tri nécessite un temps $O(r|S| \log(r|S|))$. Pour tester si un cycle est indépendant d'une famille donnée, on code les cycles par des vecteurs de $(\mathbb{Z}_2)^{A^+}$. Par l'algorithme du pivot de Gauss on maintient une version triangulée de la famille libre courante. Celle-ci est composée d'au plus r vecteurs. L'ajout d'un vecteur à cette famille par l'algorithme du pivot de Gauss nécessite $O(r|A|)$ opérations. Le temps consacré aux tests d'indépendance est donc un $O(r^2|S||A|)$. On obtient au total un calcul en temps

$$O(|S|(|S| \log |S| + |A|) + r^2|S| + r|S| \log(r|S|) + r^2|S||A|) = O(|S|^2 \log |S| + r^2|S||A|)$$

□

Remarquons que l'on peut encore restreindre le parcours dans la preuve ci-dessus. Par exemple, on peut éliminer $c(x, a)$ lorsque $p(x, o(a)).a.p(o(a^{-1}), x)$ n'est pas un cycle simple. On peut également décomposer les cycles sur une base de Kirchhoff fixée associée aux cordes d'un arbre couvrant. Un cycle simple s'exprime dans une telle base à l'aide de sa trace sur les cordes. On remplace ainsi une matrice de taille rA par une matrice de taille r^2 .

Le calcul d'une base minimale de cycles portent le nom de MCB (Minimum Cycle Basis problem) dans la littérature. On trouve de nombreuses notions et propriétés de bases minimales et de cycles "courts" dans cette littérature. La thèse de Gleiss [Gle01], fort justement intitulée "Short Cycles", contient une impressionnante bibliographie à ce propos.

Ces notions peuvent être replacées dans le cadre plus général de l'espace des cycles d'un matroïde. Golinski et Horton [GH02] ont étudiés en particulier le calcul de base minimale pour certains types de matroïdes. Notons que l'algorithme glouton présenté plus haut n'est autre que l'algorithme du même nom pour les matroïdes (où l'on considère cette fois l'espace des cycles comme un matroïde et non comme les cycles d'un matroïde). Le cours de DEA de Victor Chepoi [Che] donne une introduction très claire et concise sur les matroïdes et présente l'algorithme glouton. Pour une vue plus générale sur le sujet on pourra consulter l'article des *Notices of the American Mathematical Society* sur W. Tutte [HO04]. Par ailleurs, la complexité de l'algorithme de la proposition 2.1.11 n'est pas optimale et des améliorations ont été apportées [KMMP04, MM09, KMMP08].

Un problème ouvert est de calculer une base minimale pour l'homologie à coefficients dans \mathbb{Z} . Cette fois-ci $H_1(G, \mathbb{Z})$ ne constitue plus un matroïde. (C'est le cas pour $H_1(G, \mathbb{Z}_2)$ car \mathbb{Z}_2 étant un corps, $H_1(G, \mathbb{Z}_2)$ se trouve munie d'une structure d'espace vectoriel).

2.1.2 Cas des surfaces

Soit \mathcal{M}_g une surface triangulée sans bord de genre g dont les arêtes sont munies de poids (ou longueur). Comme pour les graphes, on s'intéresse au calcul d'une base minimale de

$H_1(\mathcal{M}_g, \mathbb{Z}_2)$, c'est-à-dire à une famille de cycles $\{c_1, \dots, c_{2g}\}$ dont les classes d'homologie forment une base de $H_1(\mathcal{M}_g, \mathbb{Z}_2)$ et qui minimise $\sum_i |c_i|$. La méthode qui suit, proposée par Erickson et Whittlesey [EW05], est une adaptation aux surfaces du calcul sur les graphes. On notera n le nombre de sommets de \mathcal{M}_g .

Le lemme 2.1.2 pour les graphes reste valide à condition de redéfinir le rang par $r = 2g$. L'algorithme glouton peut donc s'appliquer en remplaçant le test d'indépendance des cycles par celui de leur classe d'homologie. Les corollaires 2.1.4 et 2.1.6 qui caractérisent les cycles candidats à une base minimale restent également valides. Il en est de même du lemme 2.1.7, utile lorsque l'unicité du plus court chemin entre deux sommets n'est pas assurée. Notons que de manière générale cette condition peut être induite par perturbation des poids, ou plus précisément en considérant un ordre total sur les parties de 2^A . Un lemme probabiliste, dit lemme d'isolation de Mulmuley, Vazirani et Vazirani [MVV87], permet d'obtenir cette condition avec une bonne probabilité en associant à chaque arête un entier aléatoire borné par un nombre suffisamment grand (dépendant du nombre d'arêtes de \mathcal{M}_g).

Par la suite, pour tout sommet x de \mathcal{M}_g , on note T_x un arbre de plus courts chemins de source x dans le graphe (1-squelette) de \mathcal{M}_g . Pour toute arête a on pose, comme dans le cas des graphes, $c(x, a) = p(x, o(a)).a.p(x, o(a^{-1}))^{-1}$ où $p(x, y)$ est le chemin simple de x à y dans T_x .

Comme pour le cas des graphes, l'algorithme de calcul d'une base minimale consiste à appliquer l'algorithme glouton à l'union des cycles $c(x, a)$ lorsque x parcourt les sommets de \mathcal{M}_g et a les cordes de T_x . On peut cependant faire deux modifications, l'une portant sur le nombre de cycles à prendre en considération, l'autre portant sur le test d'indépendance pour l'ajout d'un cycle à la famille libre courante. La première modification repose sur le lemme suivant :

Lemme 2.1.12 *Soit x un sommet de \mathcal{M}_g . L'ensemble des classes d'homologie des cycles $c(x, a)$, lorsque a parcourt les arêtes de \mathcal{M}_g , contient au plus $6g - 3$ éléments.*

Ce lemme permet de restreindre l'algorithme glouton à $O(gn)$ cycles ($O(g)$ pour chaque sommet). Sa preuve utilise une propriété générale des graphes cubiques (i.e. 3-régulier)

Lemme 2.1.13 *Le nombre d'arêtes d'un graphe cubique est égal à -3 fois sa caractéristique d'Euler.*

Preuve : La relation d'incidence sommet/arête donne $3s = 2a$; d'où $3\chi = 3s - 2a = -a$.
□

Preuve du lemme 2.1.12 : On note $[c]$ la classe d'homologie d'un cycle c . Soit G^* le graphe dual associé au complémentaire de T_x dans \mathcal{M}_g : Les sommets de G^* correspondent aux triangles de \mathcal{M}_g et les arêtes de G^* correspondent aux arêtes dans $\mathcal{M}_g^1 \setminus T_x$. On note a^* l'arête duale d'une arête a de \mathcal{M}_g . Remarquons que $[c(x, a)] = 0$ si $a \in T_x$. Remarquons également que si trois arêtes a, b, d de \mathcal{M}_g bordent un triangle t , alors $c(x, a) + c(x, b) + c(x, d) = \partial t$ et donc $[c(x, a)] + [c(x, b)] + [c(x, d)] = 0$.

Soient deux arêtes a et b n'appartenant pas à T_x (i.e. $a^*, b^* \in G^*$).

1. si a^* a une extrémité de degré un dans G^* , alors $[c(x, a)] = 0$,
2. si a^* et b^* ont une extrémité en commun de degré deux dans G^* , alors $[c(x, a)] = [c(x, b)]$.

Pour la première propriété, on considère le triangle t correspondant au sommet de degré un de a^* . Puisque les deux autres arêtes de t sont dans T_x , les deux remarques ci-dessus impliquent $[c(x, a)] = 0$.

Pour la seconde propriété, on considère le triangle t correspondant au sommet de degré deux commun à a^* et b^* . Les arêtes de ce triangles sont a , b et une troisième arête, dans T_x . Les deux remarques ci-dessus permettent à nouveau de conclure.

Ces deux propriétés montrent que le nombre de classes d'homologie des cycles $c(x, a)$, lorsque a parcourt les arêtes de \mathcal{M}_g , est au plus égal au nombre d'arcs du graphe H^* obtenu à partir de G^* en contractant toutes les arêtes incidentes à un sommet de degré un ou deux. Notons que H^* est cubique puisque tout sommet de G^* est de degré au plus trois. Par le lemme 2.1.13, le nombre de classes est encore $-3\chi(H^*)$. Mais H^* et G^* ont le même type d'homotopie, d'où $\chi(H^*) = \chi(G^*)$.

On a par ailleurs

$$\chi(G^*) = S(G^*) - A(G^*) = F(\mathcal{M}_g) - A(G^*).$$

Or

$$A(G^*) = A(\mathcal{M}_g) - A(T_x) = A(\mathcal{M}_g) - S(T_x) + 1 = A(\mathcal{M}_g) - S(\mathcal{M}_g) + 1.$$

D'où

$$\chi(G^*) = F(\mathcal{M}_g) - A(\mathcal{M}_g) + S(\mathcal{M}_g) - 1 = \chi(\mathcal{M}_g) - 1 = 1 - 2g.$$

□

Pour un sommet x de \mathcal{M}_g on peut construire les $O(g)$ cycles pertinents de la forme $c(x, a)$ en temps $O(n \log n + gn)$. Il suffit pour cela de construire T_x par l'algorithme de Dijkstra puis de parcourir G^* et de retenir le plus court cycle de la forme $c(x, a)$ dans chaque branche de G^* , i.e. lorsque a^* parcourt une chaîne maximale de G^* .

La seconde modification porte sur le test d'indépendance – pour les classes d'homologie – d'un cycle $c(x, a)$ avec une famille libre de cycles déjà calculée. Dans le cas de l'homologie des graphes, la notion de cycle se confond avec celle de classe d'homologie. Les cycles formant un sous-espace des chaînes, on a pu travailler avec une base des chaînes, c.-à-d. les arêtes, pour exprimer matriciellement le test d'indépendance. Dans le cas des surfaces cette confusion n'est plus possible et il faut commencer par calculer une base de référence pour l'homologie afin d'exprimer les cycles dans cette base. On peut par exemple choisir une base associée à un graphe de coupe (cf. remarque 1.3.5). Une telle base peut être obtenue en temps $O(gn)$ suivant la construction de la proposition 1.3.3. Rappelons que les éléments de cette base sont en bijection avec les $2g$ cordes d'un arbre couvrant du graphe de coupe et que l'expression de la classe d'homologie d'un cycle quelconque se réduit à la trace sur les cordes de tout homologue de ce cycle contenu dans le graphe de coupe. Le calcul des $2g$ composantes (de la classe d'homologie) de chaque cycle candidat $c(x, a)$ dans la base de référence prend donc a priori un temps $O(gn)$. Pour un sommet x fixé, on peut cependant calculer dans le même temps les composantes de tous les $O(g)$ cycles candidats $c(x, a)$. Pour cela on commence par calculer les *composantes* (i.e. les

traces sur les cordes de la base de référence) de tous les chemins $p(x, y)$ en parcourant l'arbre de plus courts chemins T_x . Ceci prend un temps (et une place) $O(gn)$. Ensuite, les composantes de chacun des $O(g)$ cycles candidats $c(x, a)$ s'obtiennent en ajoutant en temps $O(g)$ celles de $p(x, o(a))$, a , et $p(x, o(a^{-1}))$. Le temps amorti pour le calcul des composantes d'un cycle candidat est donc $O(n)$ au lieu de $O(gn)$.

Par l'algorithme du pivot de Gauss on maintient une version triangulée de la famille libre courante. Celle-ci est composée d'au plus $2g$ cycles. L'ajout d'un cycle à cette famille par l'algorithme du pivot de Gauss nécessite donc $O(g^2)$ opérations.

Théorème 2.1.14 *On peut construire une base minimale de l'homologie de \mathcal{M}_g en temps $O(n^2 \log n + gn^2 + g^3n)$*

Preuve : On applique l'algorithme glouton aux $O(gn)$ cycles pertinents. Ces derniers, ainsi que leurs composantes sur une base de référence, se calculent en temps $O(n^2 \log n + gn^2)$ d'après la discussion précédente. Le tri de ces cycles suivant leur longueur prend un temps $O(gn \log n)$ auquel il faut ajouter le $O(g^3n)$ pour les tests de dépendance linéaire. \square