

# Approche graphique des groupes pleins topologiques

Vivien Ducros

encadré par  
Emmanuel Jeandel et Léo Gayral

Mars – juillet 2025

## 1 Introduction

### Contexte général

La *dynamique symbolique* s'inscrit dans le domaine plus général des systèmes dynamiques, qui s'intéresse à l'évolution d'un système dans le temps [Kür03]. Les systèmes dynamiques peuvent être à temps continu, c'est d'ailleurs le cas de la mécanique Newtonienne à l'origine de ce domaine, ou à temps discret. Dans ce cas, l'état du système est un point  $x$  de l'*espace des phases*  $X$  qui évolue selon la loi  $F$ . La *trajectoire* d'un point est la suite des itérées de  $F$  à  $x$  :  $(x, F(x), F^2(x) = F(F(x)), F^3(x), \dots)$ .

La dynamique symbolique étudie les trajectoires des systèmes après discrétisation de l'espace des phases. On effectue une partition finie de  $X$ , puis une trajectoire est encodée par la suite des *symboles* associés à la partie où se trouve chacun des éléments de la trajectoire (figure 1). En effectuant la translation des éléments de la trajectoire d'un point  $x$  on obtient celle de son image  $F(x)$ .

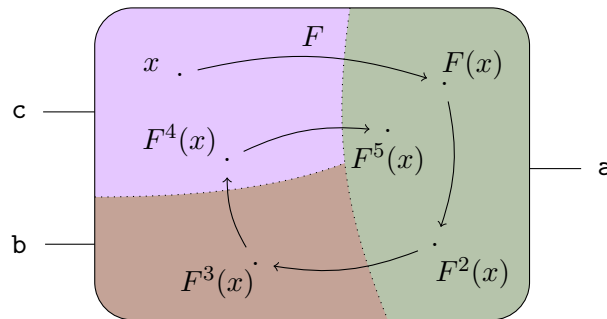


FIGURE 1 – Exemple de système dynamique  $(X, F)$ .

Le mot  $caabca \dots$  encode la trajectoire  $(x, F(x), F^2(x), F^3(x), F^4(x), F^5(x), \dots)$ .

Ces suites de symboles sont regroupées dans des ensembles appelés *sous-décalages* (*subshifts*), les objets mathématiques étudiés par la dynamique symbolique [LM95]. La *conjugaison* des subshifts est une relation d'équivalence usuelle qui correspond à l'existence d'un *automate cellulaire* (homéomorphisme qui commute avec la translation) réversible entre deux subshifts.

### Problème de recherche

Le *problème de la conjugaison*, qui consiste à déterminer si deux subshifts de type fini sont conjugués ou non, est décidable dans le cas simplement infini défini ci-dessus [Wil73]. C'est un problème très étudié [Boy08] et toujours ouvert dans le cas des systèmes *réversibles* (bijectifs), où l'on travaille avec des suites bi-infinies qui représentent la trajectoire future et passée d'un point. Le problème de la

conjugaison sur les mots bi-infinis a été montré décidable pour certaines classes de subshifts mais n'est toujours pas résolu dans le cas général.

Un *invariant* de conjugaison est une qualité mathématique associée à un subshift, partagée par des subshifts conjugués. C'est un outil pour l'étude de la conjugaison : en effet, si on parvient à montrer que cette qualité est différente pour deux subshifts, alors ils ne sont pas conjugués.

## Votre contribution

À partir de la deuxième moitié du 20<sup>e</sup> siècle, une grande variété de notations graphiques est apparue pour décrire des théories mathématiques ou physiques sous forme de diagrammes. On peut citer en particulier les diagrammes de Feynman en physique des particules, les réseaux de tenseurs [JS91], ou plus récemment, les circuits quantiques [CK17]. En informatique fondamentale, des diagrammes similaires sont créés pour représenter les morphismes de catégories monoïdales [Sel11]. Cette représentation visuelle permet de faciliter la compréhension et la manipulation des concepts théoriques, on parle de langages graphiques.

Le sujet de mon stage est de créer une représentation graphique d'un invariant de conjugaison : le groupe plein topologique [GPS99 ; Kat19 ; Mat15].

Nous définissons une PROP colorée<sup>1</sup> dont les morphismes généralisent les éléments du groupe plein topologique. Une PROP colorée est naturellement représentée par un langage graphique dont les diagrammes ont des fils colorés [CP20]. Cela nous permet de représenter graphiquement les éléments du groupe plein topologique et de les construire aisément.

## Arguments confirmant sa validité

Les catégories ont déjà été utilisées en dynamique symbolique [Bri22] et pour étudier le problème de la conjugaison [Jea21 ; Mat25]. La méthode développée est similaire à celle de la représentation des éléments du groupes des machines de Turing réversibles [Jea24].

Le groupe plein topologique associé à un subshift  $X$  est l'ensemble des homéomorphismes qui échangent les préfixes des mots de  $X$ . Les groupes pleins topologiques généralisent le groupe  $V$  de Thompson.

La représentation graphique du groupe plein topologique dans le cas du full shift et celle des groupes de Thompson sont très proches [BM14 ; CFP96]. En effet, on sait déjà que le groupe plein topologique généralise les groupes de Thompson.

## Résumé et travail futur

Le résultat principal de ce stage est le formalisme de diagrammes qui représente une généralisation des éléments du groupe plein topologique. Nous montrons que les diagrammes sont générés à partir de générateurs et relations.

Nous nous sommes focalisés sur une classe particulière de subshifts induits par des graphes, il serait intéressant de compléter ce travail avec des classes plus générales de subshifts (SFT, sofiques...).

Un autre axe de généralisation du sujet est le cas réversible, où les mots d'un subshift sont bi-infinis. Cela pourrait apporter une représentation visuelle à l'invariant de conjugaison qu'est le groupe plein topologique et un nouvel outil pour la résolution du problème de la conjugaison réversible.

---

1. Une catégorie monoïdale stricte dont les objets sont générés librement par un ensemble de couleurs.

## 2 Dynamique symbolique

Cette partie fournit les définitions nécessaires pour le travail réalisé pendant le stage. Les trois premières sous-parties peuvent être complétées par la lecture des livres de dynamique symbolique [LM95] et [Kit98]. La partie 2.4 définit le groupe plein topologique puis rappelle des résultats connus.

### 2.1 Définitions générales

Soit  $\mathcal{A}$  un *alphabet* fini,  $\mathcal{A}^*$  est l'ensemble des mots finis sur  $\mathcal{A}$ . On note  $|\mathbf{u}|$  la longueur du mot  $\mathbf{u} \in \mathcal{A}^*$  et  $\mathbf{u}_k$  sa  $k$ -ème lettre,  $k \in \llbracket 1, |\mathbf{u}| \rrbracket$ . La concaténation des mots  $\mathbf{u}$  et  $\mathbf{v}$  est notée  $\mathbf{u} \cdot \mathbf{v}$  ou  $\mathbf{uv}$ . Le mot vide  $\varepsilon$  est l'unique mot de longueur 0.

$\mathcal{A}^\omega := \mathcal{A}^\mathbb{N}$  est l'ensemble des mots simplement infinis (ou semi-infinis) sur  $\mathcal{A}$ . Le mot  $\mathbf{u} \in \mathcal{A}^*$  est un facteur de  $x \in \mathcal{A}^\omega$  lorsqu'il existe  $i \in \mathbb{N}$  tel que  $x_{\llbracket i, i+|\mathbf{u}|-1 \rrbracket} := x_i \cdots x_{i+|\mathbf{u}|-1} = \mathbf{u}$ .  $\mathcal{L}_n(x)$  est l'ensemble de ses facteurs de  $x$  de longueur  $n$ , et  $\mathcal{L}(x) := \bigcup_{n \in \mathbb{N}} \mathcal{L}_n(x)$  est son langage. Le *cylindre* induit par un mot  $\mathbf{u} \in \mathcal{A}^*$  est l'ensemble des mots semi-infinis dont  $\mathbf{u}$  est préfixe :  $[\mathbf{u}] := \{x \in \mathcal{A}^\omega : x_{\llbracket 0, |\mathbf{u}|-1 \rrbracket} = \mathbf{u}\}$ .

On définit l'opérateur de *translation (shift)*  $\sigma : \mathcal{A}^\omega \rightarrow \mathcal{A}^\omega$  tel que :

$$\forall x \in \mathcal{A}^\omega, \forall i \in \mathbb{N}, \sigma(x)_i = x_{i+1}.$$

*Exemple 1.* Avec  $\mathcal{A} = \{0, 1\}$ ,  $\sigma(01001010 \cdots) = 1001010 \cdots$ . ◁

On munit  $\mathcal{A}^\omega$  de la topologie où les cylindres  $\mathcal{B} := \{[\mathbf{u}] \mid \mathbf{u} \in \mathcal{A}^*\}$  forment une base des ouverts. Cette topologie est aussi induite par la distance de Cantor sur  $\mathcal{A}^\omega$ , qui indique que deux mots sont proches s'ils ont un long préfixe commun :

$$d(x, y) := \begin{cases} 0 & \text{si } x = y, \\ 2^{-\min\{i: x_i \neq y_i\}} & \text{sinon.} \end{cases}$$

*Exemple 2.* Par exemple,  $d(10010 \cdots, 10110 \cdots) = 2^{-2} = 1/4$ . ◁

Un ensemble  $X \subseteq \mathcal{A}^\omega$  de mots semi-infinis est un *sous-décalage (subshift)* lorsque  $X$  est fermé (pour la topologie de Cantor) et shift-invariant ( $\sigma(X) \subseteq X$ ). On lui associe  $\sigma_X$ , l'action de translation restreinte à  $X$ . Son langage est  $\mathcal{L}(X) := \bigcup_{x \in X} \mathcal{L}(x)$ . La topologie sur  $X$  est engendrée par l'ensemble  $\mathcal{B}_X$  des  $[\mathbf{u}]_X := [\mathbf{u}] \cap X$ , pour tout  $\mathbf{u} \in \mathcal{L}(X)$ .

Muni de l'action de translation, un subshift  $(X, \sigma_X)$  est un système dynamique. On écrira souvent  $\sigma$  à la place de  $\sigma_X$  lorsqu'on manipule un seul subshift  $X$ .

On peut définir de façon équivalente les subshifts comme les ensembles de mots évitant certains facteurs :  $X_F := \{x \in \mathcal{A}^\omega : \forall \mathbf{u} \in \mathcal{L}(x), \mathbf{u} \notin F\}$ , où  $F \subseteq \mathcal{A}^*$  est appelé *ensemble de motifs interdits*.

Un subshift  $X$  est *sofique* lorsqu'il existe un langage régulier  $F$  tel que  $X_F = X$ . De plus, si  $F$  est fini,  $X$  est un *subshift de type fini (SFT)*.

*Exemple 3.* Exemples de subshifts :

- $\mathcal{A}^\omega = X_\emptyset$  est le *full shift*.
- Avec  $F = \{11\}$ ,  $X_F \subseteq \{0, 1\}^\omega$  est le *golden mean shift*. C'est un SFT qui contient tous les mots qui ne possèdent pas 11 comme facteur. C'est le cas du mot de Fibonacci [MH40] :  $01001010 \cdots \in X_{\{11\}}$ . ◁

Pour  $\mathbf{u} \in \mathcal{A}^*$  et un subshift  $X$ , on note  $\mathbf{u}^{-1}X := \{x \in \mathcal{A}^\omega : \mathbf{u}x \in X\} \subseteq X$ . C'est l'ensemble des mots semi-infinis qui appartiennent à  $X$  après concaténation avec le préfixe  $\mathbf{u}$ . En général, ce n'est pas un subshift.

## 2.2 Graphes et edge shifts

Dans ce stage, on considère une classe particulière de subshifts générés par des graphes.

Un graphe orienté  $G$  est un quadruplet  $(Q, A, s, t)$  où  $Q$  est l'ensemble des sommets et  $A$  l'ensemble des arêtes. Les fonctions *source* et *target* :  $s, t : A \rightarrow Q$  associent à chaque arête ses sommets initial et final. Pour  $a \in A$ , on notera  $p \xrightarrow{a} q$  lorsque  $s(a) = p$  et  $t(a) = q$ . Cette définition autorise les boucles ( $t(a) = s(a)$ ) et les arêtes multiples ( $a \neq b$  telles que  $s(a) = s(b)$  et  $t(a) = t(b)$ ).

Le *edge shift* généré par  $G$  est donné par  $X_G := \{x \in A^\omega : \forall n \in \mathbb{N}, t(x_n) = s(x_{n+1})\}$ . C'est l'ensemble des mots simplement infinis lus sur les arêtes du graphe. En particulier, c'est un SFT :  $X_G = X_{\{ab \in A^2 : t(a) \neq s(b)\}}$ .

Soient  $G = (Q, A, s, t)$ ,  $X = X_G$ . L'ensemble des mots lus sur  $G$  qui commencent au sommet  $q \in Q$  est  $X_q := \{x \in A^\omega : \forall n \in \mathbb{N}, t(x_n) = s(x_{n+1}) \wedge s(x_0) = q\}$ . Ainsi,  $X = \biguplus_{q \in Q} X_q$ .

*Exemple 4.* Avec le graphe  $G$  de la figure 2,  $X = X_G$  est une variante du golden mean shift. On verra par la suite qu'ils sont conjugués.

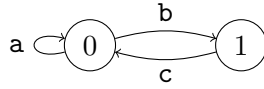


FIGURE 2 – Graphe  $G$ .

On a :  $abcaabc \dots \in X$ ,  $X_0 = (a + bc)^\omega$  et  $X_1 = c(a + bc)^\omega$ . ◁

Ces graphes rappellent les automates de la théorie des langages formels, où toutes les transitions sont étiquetées par des lettres différentes. Si on utilise plusieurs fois la même lettre, on définit les subshifts sofiques.

## 2.3 Conjugaison

Afin de comparer les subshifts, on utilise la notion de conjugaison. C'est la définition la plus naturelle d'isomorphisme entre les subshifts vus comme des systèmes dynamiques.

Le problème de déterminer si deux SFTs sont conjugués est décidable dans le cas semi-infini [Wil73].

**Définition 1.** Un *homéomorphisme* est une application continue bijective dont la réciproque est continue. On note  $\text{Homeo}(X, Y)$  l'ensemble des homéomorphismes de  $X$  vers  $Y$  et  $\text{Homeo}(X)$  ceux de  $X$  dans lui-même.

Rappelons que la fonction réciproque d'une fonction continue et bijective entre espaces compacts est elle-même continue. Puisque les subshifts sont des compacts, il suffit de montrer qu'une fonction entre subshifts est continue et bijective pour qu'elle soit un homéomorphisme.

**Définition 2.** Deux subshifts  $X$  et  $Y$  sont *conjugués* (noté  $X \cong Y$ ) s'il existe un homéomorphisme  $\varphi \in \text{Homeo}(X, Y)$  qui commute avec le shift :  $\varphi \circ \sigma_X = \sigma_Y \circ \varphi$ . C'est une relation d'équivalence appelée *conjugaison*.

*Exemple 5.* Le subshift  $X_G$  de l'exemple 4 est conjugué au golden mean shift  $X_{\{11\}}$ .

La substitution  $s : a, b \mapsto 0, c \mapsto 1$  induit l'application  $\varphi : X_G \rightarrow X_{\{11\}}$  tel que  $\varphi(x)_i = s(x_i)$ ,  $x \in X_G$ ,  $i \in \mathbb{N}$ . Soit  $x \in X_G$ ,  $\varphi(x)$  ne contient pas le facteur 11 car si c'était le cas, cc serait un facteur de  $x$ . Ainsi,  $\varphi(x) \in X_{\{11\}}$ .

La substitution  $s$  est lettre à lettre donc  $\varphi$  est continue et commute avec le shift.

De plus,  $\varphi$  est inversible. Il faut regarder des facteurs de deux lettres pour définir son inverse :  $\psi(x)_i = t(x_i x_{i+1})$  où  $t : 00 \mapsto \mathbf{a}, 01 \mapsto \mathbf{b}, 10 \mapsto \mathbf{c}$  ( $x \in X_{\{11\}}$ ).

$$\text{On a : } (\varphi \circ \psi(x))_i = s(\psi(x)_i) = s(t(x_i x_{i+1})) = \begin{cases} s(\mathbf{a}) = 0 = x_i & \text{si } x_i x_{i+1} = 00 \\ s(\mathbf{b}) = 0 = x_i & \text{si } x_i x_{i+1} = 01 \\ s(\mathbf{c}) = 1 = x_i & \text{si } x_i x_{i+1} = 10, \end{cases}$$

$$\text{et : } (\psi \circ \varphi(x))_i = t(\varphi(x)_i \varphi(x)_{i+1}) = t(s(x_i) s(x_{i+1})) = \begin{cases} t(00) = \mathbf{a} = x_i & \text{si } x_i x_{i+1} = \mathbf{aa} \\ t(00) = \mathbf{a} = x_i & \text{si } x_i x_{i+1} = \mathbf{ab} \\ t(01) = \mathbf{b} = x_i & \text{si } x_i x_{i+1} = \mathbf{bc} \\ t(10) = \mathbf{c} = x_i & \text{si } x_i x_{i+1} = \mathbf{ca} \\ t(10) = \mathbf{c} = x_i & \text{si } x_i x_{i+1} = \mathbf{cb}. \end{cases}$$

D'où  $\psi \circ \varphi(x) = x$ , pour tout  $x \in X_G$  et  $\psi \circ \varphi(x) = x$ , pour tout  $x \in X_{\{11\}}$ .

Ainsi,  $\varphi$  est un homéomorphisme,  $X_G \cong X_{\{11\}}$ . ◁

Des homéomorphismes qui commutent avec le shift peuvent être définis localement (c'est ce qu'on a fait dans l'exemple précédent). Ils sont appelés *block shift maps* ou *automates cellulaires*.

Pour étudier la conjugaison des subshifts, on peut utiliser les invariants de conjugaison : des objets associés aux subshifts qui sont communs pour les subshifts conjugués.

**Définition 3.** Un *invariant de conjugaison* est un objet  $P(X)$  associé à un subshift  $X$ , tel que pour tous subshifts  $X$  et  $Y$ ,  $X \cong Y \implies P(X) = P(Y)$ .

Lorsque la réciproque est aussi vraie,  $P$  est un invariant *complet*.

*Exemple 6.* Un mot  $x \in X$  est périodique de période  $p \in \mathbb{N}^*$  lorsque  $x_i = x_{i+p}$  pour tout  $i \in \mathbb{N}$ . Le nombre de points  $x \in X$  de période  $p$  fixée d'un subshift  $X$  est un invariant de conjugaison [LM95].

Ainsi, on peut déterminer que  $Y = \{0, 1\}^\omega$ , le full shift sur deux lettres, et  $X_G$  ne sont pas conjugués :  $0^\omega$  et  $1^\omega \in Y$  sont les mots de période 1 de  $Y$ , tandis que  $X_G$  n'a qu'un seul mot de période 1 :  $\mathbf{a}^\omega$ . ◁

## 2.4 Groupe plein topologique

Le groupe plein topologique d'un subshift [Mat15] est l'ensemble des applications cylindres sur le subshift. Les applications cylindres sont des homéomorphismes qui modifient les préfixes des éléments d'un subshift. Ainsi, un mot et son image sont asymptotiquement égaux (ils ont un suffixe infini commun). Les préfixes remplacés peuvent être de longueur différente.

On commence par donner une définition combinatoire des applications cylindres (définition 4). Une définition topologique équivalente sera donnée ensuite par la proposition 2.

**Définition 4.** Soit  $X$  un subshift. Une application  $\tau : X \rightarrow X$  est une *application cylindre* lorsqu'il existe deux familles finies  $(\mathbf{u}^{(l)})_{l \in L}, (\mathbf{v}^{(l)})_{l \in L} \in (\mathcal{A}^*)^L$  de facteurs de  $X$  telles que :

1.  $\{\mathbf{u}^{(l)}\}_X | l \in L$  est une partition de  $X$ ,
2.  $\{\mathbf{v}^{(l)}\}_X | l \in L$  est une partition de  $X$ ,
3.  $(\mathbf{u}^{(l)})^{-1}X = (\mathbf{v}^{(l)})^{-1}X$ , pour tout  $l \in L$ ,
4.  $\tau(\mathbf{u}^{(l)} \cdot z) = \mathbf{v}^{(l)} \cdot z$ , pour tous  $l \in L$  et  $z \in (\mathbf{u}^{(l)})^{-1}X$ .

On dira que  $\tau$  est définie par  $L, (\mathbf{u}^{(l)})_{l \in L}$  et  $(\mathbf{v}^{(l)})_{l \in L}$  lorsque les quatre points sont vérifiés.

Le point 4 définit  $\tau$  : elle remplace le préfixe  $\mathbf{u}^{(l)}$  de  $x = \mathbf{u}^{(l)} \cdot z \in X$  par le préfixe  $\mathbf{v}^{(l)}$ , comme indiqué par la figure 3.

Le point 1 permet de s'assurer que  $\tau$  définie par 4 est déterministe et totale  $X$  : tous les  $\mathbf{u}^{(l)}$  doivent être des préfixes de mots de  $X$ , et pour tout mot de  $X$ , il y a un unique  $\mathbf{u}^{(l)}$  qui est son préfixe. Le point 3 vérifie que l'image  $\mathbf{v}^{(l)} \cdot z$  est bien dans  $X$ . Le point 2 impose que  $\tau$  soit bijective : surjective car tout mot de  $X$  a un préfixe  $\mathbf{v}^{(l)}$ , et injective car il est unique.

$$\begin{array}{c}
 x = \quad | \text{---} \mathbf{u}^{(l)} \text{---} | \text{---} z \text{---} \text{---} \\
 \tau(x) = \quad | \text{---} \mathbf{v}^{(l)} \text{---} | \text{---} z \text{---} \text{---}
 \end{array}$$

FIGURE 3 –  $x$  et son image  $\tau(x)$  par une application cylindre.

*Exemple 7.* Soit  $X$  un subshift. L'identité  $\text{id}_X : x \in X \mapsto x$  est une application cylindre. On peut la définir par exemple avec deux familles de préfixes vides ( $L = \emptyset$ ).

Donnons deux autres exemples d'applications cylindres, accompagnés de schémas (informels) pour les visualiser.

Soit  $\tau_Y$  l'application cylindre sur le full shift  $Y = \{0, 1\}^\omega$  définie par les familles ( $\mathbf{u}^{(1)} = 0, \mathbf{u}^{(2)} = 10, \mathbf{u}^{(3)} = 11$ ) et ( $\mathbf{v}^{(1)} = 01, \mathbf{v}^{(2)} = 1, \mathbf{v}^{(3)} = 00$ ).

On peut représenter ces familles comme des arbres préfixes (figure 4). Ainsi, on peut lire sur les schémas que le préfixe 10 sera remplacé par le préfixe 1.



FIGURE 4 – Schémas des familles  $(\mathbf{u}^{(l)})_l$  et  $(\mathbf{v}^{(l)})_l$  pour l'application cylindre  $\tau_Y$ .

Ce sont exactement les représentations des éléments du groupe  $V$  de Thompson [CFP96].

Prenons  $X = X_G$  le edge shift généré par  $G$  (figure 2) de l'exemple 4.

Soit  $\tau_X : X \rightarrow X$  l'application cylindre définie par ( $\mathbf{u}^{(1)} = \mathbf{aa}, \mathbf{u}^{(2)} = \mathbf{ab}, \mathbf{u}^{(3)} = \mathbf{bc}, \mathbf{u}^{(4)} = \mathbf{c}$ ) et ( $\mathbf{v}^{(1)} = \mathbf{cbc}, \mathbf{v}^{(2)} = \mathbf{b}, \mathbf{v}^{(3)} = \mathbf{a}, \mathbf{v}^{(4)} = \mathbf{ca}$ ). En voici la représentation par arbres préfixes :

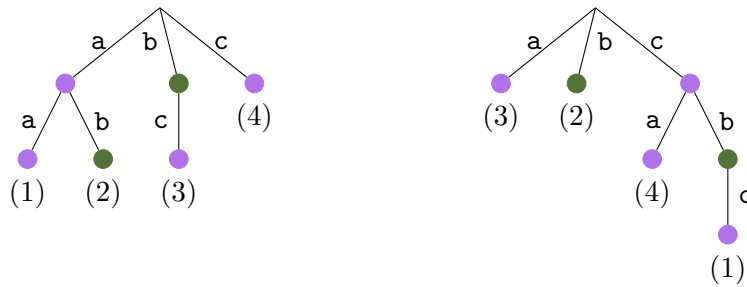


FIGURE 5 – Schéma des familles  $(\mathbf{u}^{(l)})_l$  et  $(\mathbf{v}^{(l)})_l$  pour l'application cylindre  $\tau_X$ .

◁

Les contraintes sur les familles  $(\mathbf{u}^{(l)})_l$  et  $(\mathbf{v}^{(l)})_l$  (points 1, 2 et 3) s'expriment dans les schémas grâce à la présence des perles colorées sur les nœuds. Après avoir lu un  $\mathbf{a}$  ou un  $\mathbf{c}$  on se trouve dans l'état 0, indiqué par la perle violette sur le nœud issu d'une branche  $\mathbf{a}$  ou un  $\mathbf{c}$ . Pour l'état 1, on utilise la couleur verte.

Pour assurer les points 1 et 2, un nœud violet, s'il a des fils, doit en avoir 2 (depuis l'état 0, on peut lire un **a** ou un **b**) alors qu'un nœud vert ne peut être suivi que d'une branche **b**. De plus, il faut que l'indice  $(l)$  des préfixes  $\mathbf{u}^{(l)}$  et  $\mathbf{v}^{(l)}$  soit attaché à deux nœuds de même couleur, cela correspond au point 3.

Ces schémas sont pratiques pour visualiser les applications cylindres et les contraintes de leur définition, mais on ne les a pas définis formellement. Le but de ce stage de définir une représentation graphique formelle des applications cylindres, qui permettra aussi de réaliser graphiquement des opérations (composition, inverse, ...) sur les applications cylindres.

**Proposition 1.** *Une application cylindre  $\tau : X \rightarrow X$  est un homéomorphisme.*

*Démonstration.* Soient  $L$ ,  $(\mathbf{u}^{(l)})_{l \in L}$  et  $(\mathbf{v}^{(l)})_{l \in L}$  qui vérifient les quatre points de la définition 4.

$\tau$  est inversible, son inverse  $\tau^{-1}$  est définie en inversant le rôle des familles  $(\mathbf{u}^{(l)})_{l \in L}$  et  $(\mathbf{v}^{(l)})_{l \in L}$  dans la définition. Ainsi,  $\tau^{-1}$  est aussi une application cylindre.

Montrons que  $\tau$  est continue : soient  $x \in X$ ,  $n \geq \max_{l \in L} |\mathbf{v}^{(l)}|$  et  $\varepsilon = 2^{-n}$ .

Soient  $l \in L$ ,  $z \in (\mathbf{u}^{(l)})^{-1}X$  tels que  $x = \mathbf{u}^{(l)} \cdot z$  et  $\tau(x) = \mathbf{v}^{(l)} \cdot z$ .

On pose  $\delta = 2^{-n+|\mathbf{v}^{(l)}|-|\mathbf{u}^{(l)}|}$ . Soit  $y \in X$  tel que  $d(x, y) < \delta$ .

On a :  $x_{\llbracket 0, n-|\mathbf{v}^{(l)}|+|\mathbf{u}^{(l)} \rrbracket} = y_{\llbracket 0, n-|\mathbf{v}^{(l)}|+|\mathbf{u}^{(l)} \rrbracket} = \mathbf{u}^{(l)} \cdot z_{\llbracket 0, n-|\mathbf{v}^{(l)} \rrbracket}$ .

Donc  $\tau(x)_{\llbracket 0, n \rrbracket} = \tau(y)_{\llbracket 0, n \rrbracket} = \mathbf{v}^{(l)} \cdot z_{\llbracket 0, n-|\mathbf{v}^{(l)} \rrbracket}$ . Ainsi,  $d(\tau(x), \tau(y)) < 2^{-n} = \varepsilon$ .

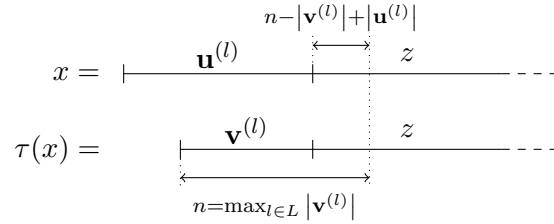


FIGURE 6 – Preuve de la continuité des applications cylindres.

□

*Remarque 1.* Cette représentation en terme de familles n'est pas unique. Dans les preuves, on utilisera souvent la notion de raffinement suivante. ◁

**Définition 5.** Les familles  $(\mathbf{u}^{(l,k)})_{l \in L, k \in K(l)}$ ,  $(\mathbf{v}^{(l,k)})_{l \in L, k \in K(l)}$  forment un *raffinement* de  $\tau$  (définie par  $(\mathbf{u}^{(l)})_{l \in L}$ ,  $(\mathbf{v}^{(l)})_{l \in L}$ ) lorsque :

- $\{[\mathbf{u}^{(l,k)}]_X \mid k \in K(l)\}$  est une partition de  $[\mathbf{u}^{(l)}]_X$ , pour tout  $l \in L$ ,
- $\{[\mathbf{v}^{(l,k)}]_X \mid k \in K(l)\}$  est une partition de  $[\mathbf{v}^{(l)}]_X$ , pour tout  $l \in L$ ,
- $(\mathbf{u}^{(l,k)})^{-1}X = (\mathbf{v}^{(l,k)})^{-1}X$ , pour tous  $l \in L$  et  $k \in K(l)$ .

Alors,  $\tau'$  définie par  $(\mathbf{u}^{(l,k)})_{l \in L, k \in K(l)}$ ,  $(\mathbf{v}^{(l,k)})_{l \in L, k \in K(l)}$  est une application cylindre telle que  $\tau' = \tau$ .

*Exemple 8.* Considérons de nouveau l'application  $\tau_X$  de l'exemple 7 (figure 5). On crée un raffinement  $(\mathbf{u}^{(l,k)})_{l \in L, k \in K(l)}$ ,  $(\mathbf{v}^{(l,k)})_{l \in L, k \in K(l)}$  pour  $\tau_X$  où  $K(1) = K(3) = \{\star\}$ ,  $K(2) = \{1\}$  et  $K(4) = \{1, 2\}$ .

$$\text{On pose : } \begin{cases} \mathbf{u}^{(1,\star)} = \mathbf{u}^{(1)} = \mathbf{aa} \\ \mathbf{u}^{(2,1)} = \mathbf{u}^{(2)} \cdot \mathbf{c} = \mathbf{abc} \\ \mathbf{u}^{(3,\star)} = \mathbf{u}^{(3)} = \mathbf{bc} \\ \mathbf{u}^{(4,1)} = \mathbf{u}^{(4)} \cdot \mathbf{a} = \mathbf{ca} \\ \mathbf{u}^{(4,2)} = \mathbf{u}^{(4)} \cdot \mathbf{b} = \mathbf{cb} \end{cases} \quad \text{et} \quad \begin{cases} \mathbf{v}^{(1,\star)} = \mathbf{v}^{(1)} = \mathbf{cbc} \\ \mathbf{v}^{(2,1)} = \mathbf{v}^{(2)} \cdot \mathbf{c} = \mathbf{bc} \\ \mathbf{v}^{(3,\star)} = \mathbf{v}^{(3)} = \mathbf{a} \\ \mathbf{v}^{(4,1)} = \mathbf{v}^{(4)} \cdot \mathbf{a} = \mathbf{caa} \\ \mathbf{v}^{(4,2)} = \mathbf{v}^{(4)} \cdot \mathbf{b} = \mathbf{cab} \end{cases} .$$

La visualisation du raffinement a fait grandir les arbres préfixes de la figure 5 :

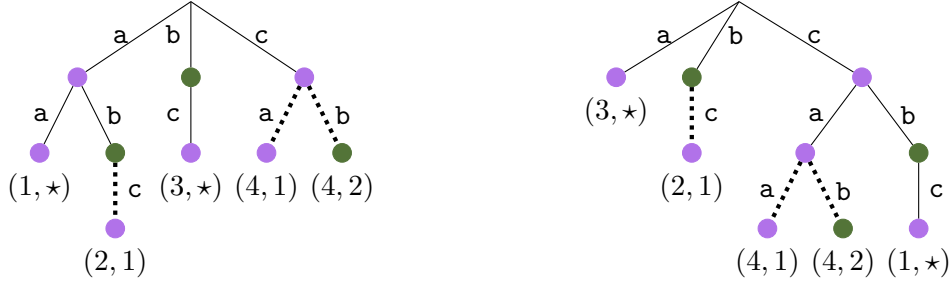


FIGURE 7 – Schémas du raffinement  $(\mathbf{u}^{(l,k)})$ ,  $(\mathbf{v}^{(l,k)})$  de  $\tau_X$ .

◁

La définition des applications cylindres (définition 4) est combinatoire. Elle permet de décrire leur action sur les mots d'un subshift. Parfois, une définition topologique est donnée : c'est l'objet de la proposition suivante.

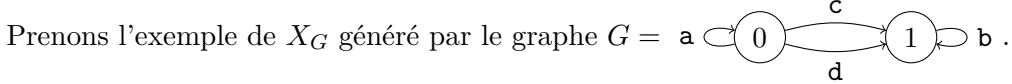
**Proposition 2** ([Mat25]). *Soit  $\tau \in \text{Homeo}(X)$ . Les énoncés suivants sont équivalents :*

1.  $\tau$  est une application cylindre.
2. Il existe deux fonctions continues  $k, l : X \rightarrow \mathbb{N}$  telles que  $\forall x \in X, \sigma^{k(x)}(\tau(x)) = \sigma^{l(x)}(x)$ .

*Démonstration.* Le sens 1  $\implies$  2 est facile : pour  $x \in X$ , prendre  $k(x) = |\mathbf{u}^{(\ell)}|$  et  $l(x) = |\mathbf{v}^{(\ell)}|$ , où  $\ell$  vérifie  $x \in [\mathbf{u}^{(\ell)}]$ .

La preuve de la réciproque est traitée dans [Mat25]. Elle repose sur le fait qu'une fonction continue  $k : X \rightarrow \mathbb{N}$  ne prend qu'un nombre fini de valeurs ( $k(X) \subseteq \mathbb{N}$  est fini).  $\square$

*Remarque 2.* Dans le sens 2  $\implies$  1, la condition de continuité de  $k$  et  $l$  est nécessaire.



On définit l'homéomorphisme  $g$  suivant qui, malgré l'existence de fonctions  $k$  et  $l$  non continues qui vérifient  $\forall x \in X, \sigma^{k(x)}(g(x)) = \sigma^{l(x)}(x)$ , n'est pas une application cylindre.

On pose  $g \in \text{Homeo}(X_G)$  tel que :

- $\forall n \in \mathbb{N}, g(\mathbf{a}^n \mathbf{c} \mathbf{b}^\omega) = \mathbf{a}^n \mathbf{d} \mathbf{b}^\omega$  et  $g(\mathbf{a}^n \mathbf{d} \mathbf{b}^\omega) = \mathbf{a}^n \mathbf{c} \mathbf{b}^\omega$
- $g(\mathbf{a}^\omega) = \mathbf{a}^\omega$  et  $g(\mathbf{b}^\omega) = \mathbf{b}^\omega$

Les fonctions  $k, l : \begin{cases} \mathbf{a}^n \mathbf{c} \mathbf{b}^\omega, \mathbf{a}^n \mathbf{d} \mathbf{b}^\omega & \mapsto n + 1 \\ \mathbf{a}^\omega, \mathbf{b}^\omega & \mapsto 0 \end{cases}$  vérifient  $\forall x \in X, \sigma^{k(x)}(g(x)) = \sigma^{l(x)}(x)$  et ne sont pas continues :  $\lim_{n \rightarrow \infty} \mathbf{a}^n \mathbf{c} \mathbf{b}^\omega = \mathbf{a}^\omega$ , mais  $(k(\mathbf{a}^n \mathbf{c} \mathbf{b}^\omega))_{n \in \mathbb{N}}$  diverge.

Pourtant,  $g$  n'est pas une application cylindre : supposons par contradiction qu'il existe deux familles finies  $(\mathbf{u}^{(l)})_{l \in L}$ ,  $(\mathbf{v}^{(l)})_{l \in L}$  telles que  $g(\mathbf{u}^{(l)} z) = \mathbf{v}^{(l)} z$  (pour tout  $l \in L, z \in (\mathbf{u}^{(l)})^{-1} X_G$ ). Soit  $N = \max_{l \in L} |\mathbf{u}^{(l)}|$ . Alors, puisque  $\mathbf{a}^N \mathbf{c} \mathbf{b}^\omega$  et  $\mathbf{a}^N \mathbf{d} \mathbf{b}^\omega$  ont les mêmes préfixes de taille inférieure à  $N$ ,  $g(\mathbf{a}^N \mathbf{c} \mathbf{b}^\omega) = g(\mathbf{a}^N \mathbf{d} \mathbf{b}^\omega)$ , absurde.  $\triangleleft$

Finalement, on définit l'invariant de conjugaison qui nous intéresse :

**Définition 6.** Le groupe plein topologique (topological full group)  $\Gamma_X$  d'un subshift  $X$  est l'ensemble des applications cylindres sur  $X$ .

On trouve aussi parfois le nom de groupe plein *continu* (*continuous full group*), particulièrement quand la définition combinatoire des applications cylindres est utilisée [Mat25 ; MM17]. Historiquement, c'est le terme *topologique* qui a été utilisé [GPS99 ; Kat19 ; Mat15].

*Remarque 3.* Comme son nom l'indique, le groupe plein topologique est un groupe. ◁

**Proposition 3.** *Le groupe plein topologique est un invariant de conjugaison.*

*Démonstration.* Soient  $X \subseteq \mathcal{A}^\omega$  et  $Y \subseteq \mathcal{B}^\omega$  deux subshifts conjugués. Soit  $\varphi \in \text{Homeo}(X, Y)$  une conjugaison de  $X$  vers  $Y$ . Montrons que  $\Gamma_X \simeq \Gamma_Y$  ( $\Gamma_X$  et  $\Gamma_Y$  sont isomorphes).

Soit  $\psi : \Gamma_X \rightarrow \Gamma_Y$  tel que  $\psi : \tau \mapsto \varphi \circ \tau \circ \varphi^{-1}$ .

On vérifie que  $\psi(\tau) \in \Gamma_Y$  :

- $\psi(\tau) = \varphi \circ \tau \circ \varphi^{-1} \in \text{Homeo}(Y)$  car  $\varphi$  et  $\tau$  sont des homéomorphismes.
- De plus, soit  $y \in Y$ ,  $x = \varphi^{-1}(y)$ . On pose  $k'(y) = k(\varphi^{-1}(y))$ ,  $l'(y) = l(\varphi^{-1}(y))$ .

$$\begin{aligned} \sigma^{k'(y)}(\psi(\tau)(y)) &= \sigma^{k'(y)}(\varphi \circ \tau \circ \varphi^{-1}(y)) = \sigma^{k(x)}(\varphi \circ \tau(x)) \\ &= \varphi(\sigma^{k(x)}(\tau(x))) = \varphi(\sigma^{l(x)}(x)) && \text{car } \tau \in \Gamma_X \\ &= \sigma^{l(x)}(\varphi(x)) = \sigma^{l'(y)}(\varphi \circ \varphi^{-1}(y)) \\ &= \sigma^{l'(y)}(y) \end{aligned}$$

Donc  $\psi(\tau)$  est une application cylindre sur  $Y$  ( $k'$  et  $l'$  sont continues).

Ainsi,  $\psi$  est un isomorphisme de groupes ( $\psi^{-1} : \tau' \mapsto \varphi^{-1} \circ \tau' \circ \varphi$ ),  $\Gamma_X$  et  $\Gamma_Y$  sont isomorphes. ◻

*Remarque 4.* Dans le cas général, le groupe plein topologique n'est pas un invariant complet.

Reprenons le graphe  $G = \mathbf{a} \begin{array}{c} \circlearrowleft \\ \text{0} \\ \circlearrowright \end{array} \begin{array}{c} \text{b} \\ \text{1} \\ \text{c} \end{array}$  (figure 2) de l'exemple 4.

Les subshifts  $X = X_G$  et  $Y = \{0, 1\}^\omega$  ne sont pas conjugués (voir exemple 6). Pourtant, ils ont le même groupe plein topologique (à isomorphisme près).

Nous allons exhiber un homéomorphisme  $h : X \rightarrow Y$  qui permettra de définir un isomorphisme  $\varphi$  entre les groupes  $\Gamma_X$  et  $\Gamma_Y$ .



FIGURE 8 – Transducteur  $\mathcal{T}_h$  et son inverse  $\mathcal{T}_{h'}$ .

L'application  $h$  induite par le transducteur  $\mathcal{T}_h$  (voir figure 8) est définie sur les mots de  $X$ , et prend ses valeurs dans  $Y$ . C'est une fonction continue. Elle est bijective : sa réciproque est la fonction  $h'$  induite par le transducteur inverse  $\mathcal{T}_{h'} = (\mathcal{T}_h)^{-1}$  (on remarque en effet que  $h'(Y) = X$ ).

Ainsi,  $h \in \text{Homeo}(X, Y)$ .

On pose  $\varphi : \tau \in \Gamma_X \mapsto h \circ \tau \circ h'$ .

Montrons que, pour tout  $\tau \in \Gamma_X$ ,  $\varphi(\tau) : Y \rightarrow Y$  est une application cylindre.

Soit  $\tau \in \Gamma_X$  définie combinatoirement par  $L$ ,  $(\mathbf{u}^{(l)})_{l \in L}$  et  $(\mathbf{v}^{(l)})_{l \in L}$  telle que :

- pour tout  $l \in L$ ,  $|\mathbf{u}^{(l)}| \geq 3$  et  $|\mathbf{v}^{(l)}| \geq 1$ . En effet, on peut toujours trouver de telles familles, si besoin on raffine suffisamment pour allonger les préfixes.
- pour tout  $l \in L$ , les dernières lettres de  $\mathbf{u}^{(l)}$  et  $\mathbf{v}^{(l)}$  ne sont pas des  $\mathbf{b}$ . Si cette condition n'est pas vérifiée, on peut trouver un raffinement qui convient :

Pour tout  $l \in L$  tel que  $\mathbf{u}^{(l)}_{|\mathbf{u}^{(l)}|} = \mathbf{b}$ , on pose  $K(l) = \{1, 2\}$  et  $\mathbf{u}^{(l,1)} = \mathbf{u}^{(l)}_{\mathbf{a}}$ ,  $\mathbf{v}^{(l,1)} = \mathbf{v}^{(l)}_{\mathbf{a}}$ ,  $\mathbf{u}^{(l,2)} = \mathbf{u}^{(l)}_{\mathbf{c}}$ ,  $\mathbf{v}^{(l,2)} = \mathbf{v}^{(l)}_{\mathbf{c}}$ .

Ça fonctionne car  $\mathbf{u}^{(l)}_{|\mathbf{u}^{(l)}|} = \mathbf{b} \iff (\mathbf{u}^{(l)})^{-1}X = X_1 \iff (\mathbf{v}^{(l)})^{-1}X = X_1 \iff \mathbf{v}^{(l)}_{|\mathbf{v}^{(l)}|} = \mathbf{b}$ .

Alors, l'application cylindre définie par  $L$ ,  $(h(\mathbf{u}^{(l)}))_{l \in L}$  et  $(h(\mathbf{v}^{(l)}))_{l \in L}$  est  $\varphi(\tau) = h \circ \tau \circ h'$ .

Ainsi,  $\varphi : \tau \in \Gamma_X \mapsto h \circ \tau \circ h' \in \Gamma_Y$  est un morphisme de groupes, qui est bijectif ( $\varphi^{-1} : \tau' \mapsto h' \circ \tau' \circ h$ ). On conclut que  $\Gamma_X$  et  $\Gamma_Y$  sont isomorphes.  $\triangleleft$

### 3 Représenter les applications cylindres par des diagrammes

Cette partie est destinée à donner l'intuition de la représentation formelle des applications cylindres développée pendant le stage. On s'appuiera sur des exemples pour donner les idées principales et motiver l'intérêt des diagrammes.

Soit  $G = (Q, A, s, t)$  le graphe de l'exemple 4 où l'on a ajouté des couleurs distinctes aux sommets (figure 9). Les arêtes sont de la couleur de leur sommet final.

Dans toute cette partie,  $X = X_G$ , le edge shift généré par  $G$ .

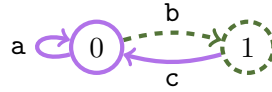


FIGURE 9 – Graphe  $G$ .

#### 3.1 Générateurs lecture et écriture : construction et interprétation

À l'aide du graphe, on crée des éléments de diagrammes appelés *générateurs*, qui sont de deux types : *lecture* et *écriture*. Un fil de la couleur de  $q \in Q$  représente l'ensemble  $X_q$  des mots commençant au sommet  $q$ .

Le générateur lecture associé au sommet  $q$  a un fil de la couleur de  $q$  à gauche, un nœud de la même couleur, puis un fil pour chaque lettre  $a \in s^{-1}(q)$  pouvant être lue à partir de  $q$ . La couleur de ces fils dépend de l'état  $t(a)$  obtenu après lecture de  $a$  dans le graphe ( $q \xrightarrow{a} t(a)$ ).



FIGURE 10 – Générateurs lecture associés aux sommets 0 et 1 du graphe  $G$ .

Par la suite, on ne notera plus le type des fils avec le nom des sommets, la couleur des fils est suffisante pour indiquer le sommet considéré.

Les générateurs écriture sont obtenus par symétrie à partir des générateurs lecture.

Donnons maintenant une interprétation mathématique aux générateurs.



FIGURE 11 – Générateurs écriture associés aux sommets 0 et 1 du graphe  $G$ .

Les diagrammes construits à partir des générateurs représenteront des fonctions sur les mots du subshift  $X_G$ . Ils se lisent de gauche à droite. Un mot est donné en entrée dans l'un des fils de gauche, puis l'image est renvoyée par l'un des fils de droite. Les mots que l'on peut trouver sur un fil de la couleur  $q$  sont les mots de  $X_q$  (ceux lus à partir de  $q$  dans  $G$ ).

Le générateur lecture  $q$  prend en entrée un mot  $x \in X_q$  et renvoie  $\sigma(x)$  par le fil de sortie qui correspond à la première lettre de  $x$ , voir figure 12.

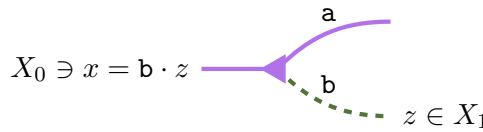


FIGURE 12 – Interprétation du générateur lecture associé au sommet 0. Avec l'entrée  $x = \mathbf{b} \cdot z$ , on obtient  $\sigma(x) = z$  sur le deuxième fil de sortie.

Inversement, les générateurs écriture ajoutent au début d'un mot la lettre correspondante au fil d'entrée (figure 13).



FIGURE 13 – Interprétation du générateur écriture associé au sommet 1. Avec  $z$  en entrée, on obtient  $\mathbf{c} \cdot z$  en sortie.

Cette interprétation s'étend ensuite à tous les diagrammes dessinés à partir des générateurs : on suit la position d'une entrée  $x$  et ses images par les générateurs jusqu'à arriver à un fil de sortie.

### 3.2 Exemples de diagrammes

Dessignons des diagrammes à partir des générateurs (figure 14). On peut relier les générateurs lorsque les fils ont la même couleur, et les fils peuvent se croiser. Essayons de les interpréter comme fonctions, à partir des interprétations données aux générateurs.

Le premier diagramme représente une fonction  $\tau : X \rightarrow X$  qui associe au mot

- $\mathbf{aa} \cdot z$  donné sur le premier fil d'entrée, le mot  $\mathbf{cbc} \cdot z$  par le deuxième fil de sortie,
- $\mathbf{ab} \cdot z$  donné sur le premier fil d'entrée, le mot  $\mathbf{b} \cdot z$  par le premier fil de sortie,
- $\mathbf{bc} \cdot z$  donné sur le premier fil d'entrée, le mot  $\mathbf{a} \cdot z$  par le premier fil de sortie,
- $\mathbf{c} \cdot z$  donné sur le deuxième fil d'entrée, le mot  $\mathbf{ca} \cdot z$  par le deuxième fil de sortie.

De même, le second diagramme représente une fonction  $\rho : X \rightarrow X$  qui associe au mot

- $\mathbf{a} \cdot z$  donné sur le premier fil d'entrée, le mot  $\mathbf{bc} \cdot z$  par le premier fil de sortie,
- $\mathbf{bc} \cdot z$  donné sur le premier fil d'entrée, le mot  $\mathbf{a} \cdot z$  par le premier fil de sortie,
- $x \in X_2$  donné sur le deuxième fil d'entrée, le mot  $x$  par le deuxième fil de sortie.

Si on oublie l'information des fils d'entrée et de sortie (en fait, elle est déterminée par la première lettre de chaque mot), la fonction  $\tau$  est l'application cylindre  $\tau_X$  de l'exemple 7, figure 5.

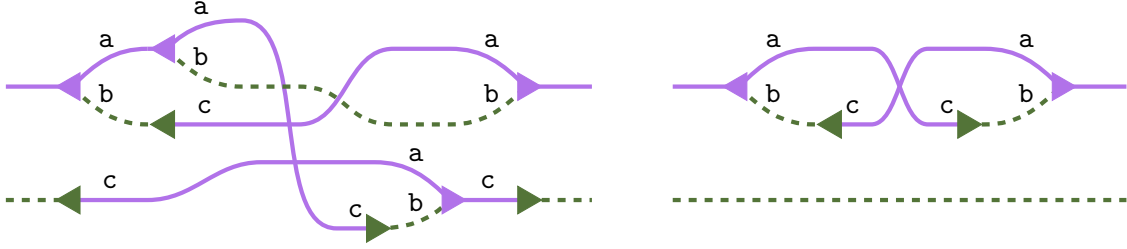


FIGURE 14 – Deux diagrammes construits à partir des générateurs.

On remarque une ressemblance entre le diagramme ci-dessus et le schéma utilisé pour visualiser les familles qui définissent  $\tau_X$  : on peut pivoter les arbres pour les mettre face à face, et relier les indices (comparer la figure 15 et le premier diagramme de la figure 14).

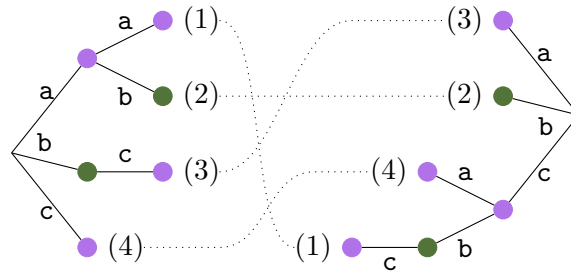


FIGURE 15 – Transformation du schéma de l'application  $\tau_X$  (figure 5).

La seconde fonction est aussi une application cylindre, définie par les familles  $(\mathbf{u}^{(1)} = \mathbf{a}, \mathbf{u}^{(2)} = \mathbf{bc}, \mathbf{u}^{(3)} = \mathbf{c})$  et  $(\mathbf{v}^{(1)} = \mathbf{bc}, \mathbf{v}^{(2)} = \mathbf{a}, \mathbf{v}^{(3)} = \mathbf{c})$ .

On verra par la suite que les fonctions représentées par des diagrammes construits à partir des générateurs, et qui ont un fil de chaque couleur en entrée et en sortie, sont des applications cylindres. Inversement, toute application cylindre peut être représentée par un diagramme.

On peut aussi composer ces applications en joignant les diagrammes comme dans la figure 16. Ce nouveau diagramme se simplifie (on fait une écriture puis une lecture des mêmes lettres), puis est réorganisé et de nouveau simplifié, avant d'obtenir une forme plus réduite.

On a écrit des symboles d'égalité entre les diagrammes : c'est le formalisme que nous avons développé qui nous permet de le faire. Tous ces diagrammes représentent en effet la même application cylindre  $\rho \circ \tau$ .

Désormais, on peut obtenir aisément une définition par familles de préfixes de l'application  $\rho \circ \tau$ , sans avoir manipulé explicitement les préfixes :

$$(\mathbf{u}^{(1)} = \mathbf{aa}, \mathbf{u}^{(2)} = \mathbf{abc}, \mathbf{u}^{(3)} = \mathbf{b}, \mathbf{u}^{(4)} = \mathbf{c}) \text{ et } (\mathbf{v}^{(1)} = \mathbf{cbc}, \mathbf{v}^{(2)} = \mathbf{a}, \mathbf{v}^{(3)} = \mathbf{b}, \mathbf{v}^{(4)} = \mathbf{ca}).$$

### 3.3 Remarques

La représentation de l'inverse d'une application s'obtient facilement par symétrie d'axe vertical. Cela vient du fait que les générateurs écriture ont été représentés symétriquement des générateurs lecture, et que les fonctions qu'ils représentent sont les inverses de celles représentées par les générateurs lecture. On remarque que  $\rho$  est une involution : le diagramme qui la représente est symétrique.

Puisque  $X_G = \biguplus_{q \in Q} X_q$ , un diagramme qui représente une application cylindre doit avoir un fil de chaque couleur en entrée et en sortie. En effet, s'il manque une couleur  $q$  en entrée (respectivement,

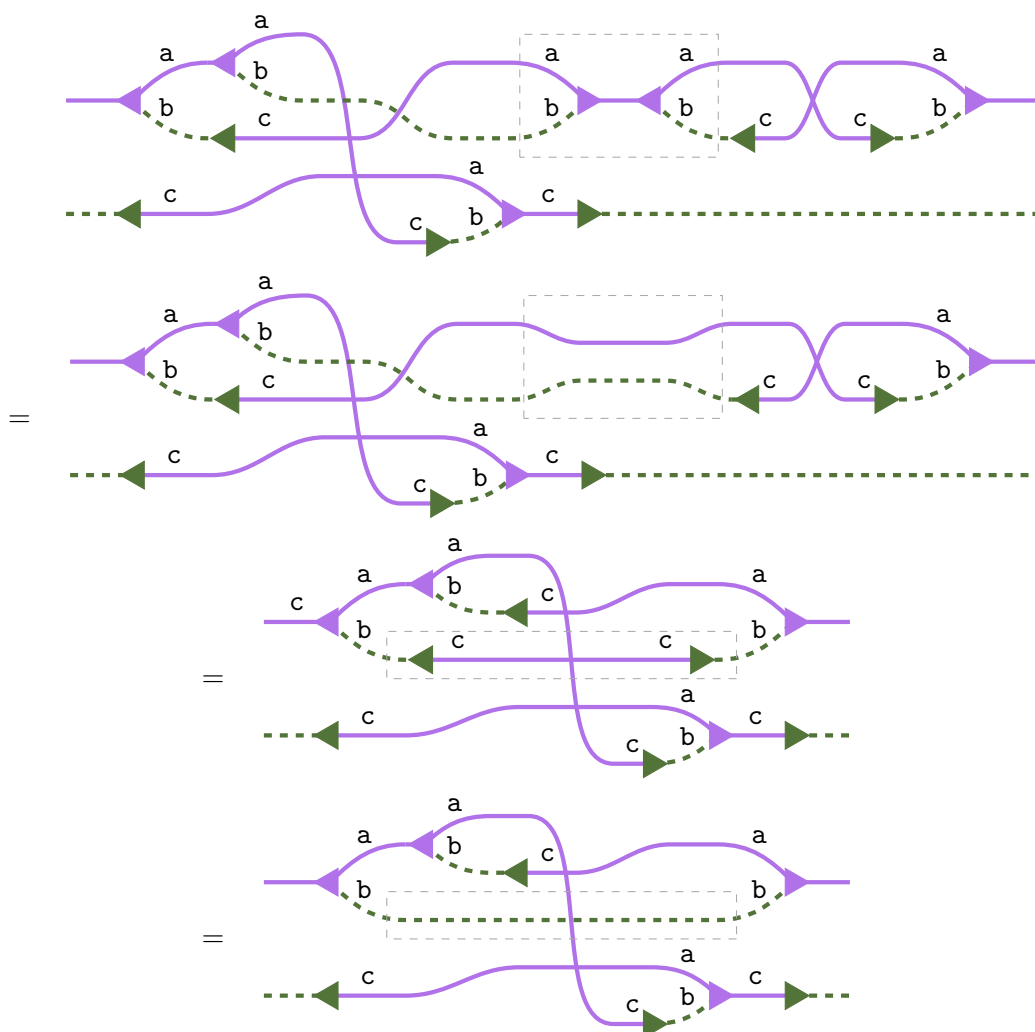


FIGURE 16 – Simplification d'un diagramme représentant  $\rho \circ \tau$ .

en sortie), alors la fonction représentée par le diagramme ne sera pas définie sur  $X_q$  (resp. ne prendra pas de valeurs dans  $X_q$ ). Aussi, si une couleur  $q$  est présente deux fois en entrée, selon si on utilise le premier ou le deuxième fil d'entrée de couleur  $q$ , la sortie sera a priori différente.

Cependant, on ne peut pas étudier uniquement les diagrammes ayant un fil de chaque couleur en entrée et en sortie car l'intérêt serait limité : on a besoin de décrire ce qu'il se passe sur les sous-parties d'un diagramme pour le simplifier. Pour formaliser cette représentation graphique, notre méthode va donner une interprétation à tous les diagrammes construits à partir des générateurs.

Cela nous demande de généraliser la notion d'application cylindre. On montrera que les applications cylindres généralisées définies au début de la partie suivante forment une PROP colorée, une catégorie dont les morphismes se représentent graphiquement [Sel11].

## 4 Catégories pour les applications cylindres généralisées

Dans cette partie, nous définissons les applications cylindres généralisés (partie 4.1) puis nous montrons que ce sont les morphismes d'une catégorie PROP colorée (partie 4.2). Toute PROP admet

une représentation graphique [Sel11], cependant notre objectif est plus large : il faut arriver à une représentation qui utilise seulement des diagrammes élémentaires, les générateurs. Nous définirons alors une autre PROP (partie 4.3), où les morphismes sont des diagrammes générés par les diagrammes élémentaires lecture et écriture. À ces générateurs s'ajoutent des relations pour définir une équivalence sur les diagrammes. Finalement, on montrera que ces deux PROPs sont équivalentes (partie 4.4). Cette méthode est semblable à celle utilisée pour les groupes des machines de Turing réversible [Jea24].

Faisons une analogie avec les groupes : on peut définir un groupe en donnant ses éléments et en définissant sa loi. Alternativement, on peut donner quelques éléments particuliers, les générateurs (tels que tout élément du groupe se décompose en produit de générateurs et de leurs inverses), et des relations entre des éléments du groupe avant de quotienter par ces relations.

Dans ce rapport, la PROP des applications cylindres généralisées correspond à cette première description et celle des diagrammes à la deuxième. Montrer l'équivalence entre ces PROPs est l'analogie de prouver l'isomorphisme des groupes.

Dans toute cette partie,  $X = X_G$  est un edge shift généré par un graphe  $G = (Q, A, s, t)$ .

## 4.1 Applications cylindres généralisées

Les applications cylindres généralisées seront, à terme, toutes les fonctions que l'on peut représenter par des diagrammes.

Tandis que les applications cylindres étaient définies sur  $X = \bigsqcup_{q \in Q} X_q$ , les applications cylindres généralisées seront définies de  $X[\mathbf{p}] := \bigoplus_{i=1}^{|\mathbf{p}|} X_{\mathbf{p}[i]}$  vers  $X[\mathbf{q}] := \bigoplus_{j=1}^{|\mathbf{q}|} X_{\mathbf{q}[j]}$ , où  $\mathbf{p}$  et  $\mathbf{q}$  sont des listes des sommets. La proposition 5 indique pour quels  $\mathbf{p}$  et  $\mathbf{q}$  on retrouve les applications cylindres.

Les listes sont des séquences finies de sommets. Pour une liste  $\mathbf{p}$ , sa taille est notée  $|\mathbf{p}|$ , et son  $k$ -ième élément  $\mathbf{p}[k]$ , pour  $k \in \llbracket 1, |\mathbf{p}| \rrbracket$ . Ainsi,  $\mathbf{p} = \langle \mathbf{p}[1], \mathbf{p}[2], \dots, \mathbf{p}[|\mathbf{p}|] \rangle$ . La liste de taille nulle est la liste vide  $\mathbf{o}$  et la concaténation est notée  $\mathbf{p} + \mathbf{q}$  ou  $\mathbf{p}\mathbf{q}$ .

*Remarque 5.* Bien que cette définition soit la même que celle des mots, elle offre une nouvelle notation utile pour distinguer les mots finis sur les arêtes  $(\mathbf{u}, \mathbf{v}, \dots \in A^*)$  des listes de sommets  $(\mathbf{p}, \mathbf{q}, \dots)$ .  $\triangleleft$

Pour une famille finie d'ensembles  $(E_i)_{i \in I}$ , la somme directe est notée  $\bigoplus_{i \in I} E_i := \{(i, x) \mid x \in E_i, i \in I\}$ . Une définition des sommes directes, accompagnée de précisions sur les sommes directes de familles indexées par des listes se trouve dans l'annexe A.

*Remarque 6.* On peut définir une topologie naturelle sur  $X[\mathbf{p}] = \bigoplus_i X_{\mathbf{p}[i]}$ , générée par une base de cylindres ouverts et fermés :  $\mathcal{B}_{X[\mathbf{p}]} := \{(k, [\mathbf{u}]_X)\}$ , où  $(k, [\mathbf{u}]_X) := \{(k, x) \in X[\mathbf{p}] : x \in [\mathbf{u}]_X\}$ .

Si  $\mathbf{u} = \varepsilon$ ,  $(k, [\varepsilon]_X) = \{(k, x) : x \in X_{\mathbf{p}[k]}\}$ . Si  $\mathbf{u} \neq \varepsilon$  et  $s(\mathbf{u}_1) \neq \mathbf{p}[k]$ , alors  $(k, [\mathbf{u}]_X) = \emptyset$  car  $\mathbf{u}$  ne commence pas au sommet  $\mathbf{p}[k]$ .  $\triangleleft$

**Définition 7.** Une *application cylindre généralisée* est une application

$$f : \underbrace{\bigoplus_{i=1}^{|\mathbf{p}|} X_{\mathbf{p}[i]}}_{=X[\mathbf{p}]} \longrightarrow \underbrace{\bigoplus_{j=1}^{|\mathbf{q}|} X_{\mathbf{q}[j]}}_{=X[\mathbf{q}]}$$

(où  $\mathbf{p}$  et  $\mathbf{q}$  sont deux listes de sommets dans  $Q$ ), vérifiant l'existence d'un ensemble fini  $L$  et de deux familles  $(i^{(l)}, \mathbf{u}^{(l)})_{l \in L} \in (\llbracket 1, |\mathbf{p}| \rrbracket \times A^*)^L$  et  $(j^{(l)}, \mathbf{v}^{(l)})_{l \in L} \in (\llbracket 1, |\mathbf{q}| \rrbracket \times A^*)^L$  telles que :

- $\{(i^{(l)}, [\mathbf{u}^{(l)}]_X) \mid l \in L\}$  est une partition de  $X[\mathbf{p}]$ ,
- $\{(j^{(l)}, [\mathbf{v}^{(l)}]_X) \mid l \in L\}$  est une partition de  $X[\mathbf{q}]$ ,
- $(\mathbf{u}^{(l)})^{-1} X_{\mathbf{p}[i^{(l)}]} = (\mathbf{v}^{(l)})^{-1} X_{\mathbf{q}[j^{(l)}]}$ , pour tout  $l \in L$ ,

d.  $f(i^{(l)}, \mathbf{u}^{(l)}z) = (j^{(l)}, \mathbf{v}^{(l)}z)$ , pour tous  $l \in L$ ,  $z \in (\mathbf{u}^{(l)})^{-1}X_{\mathfrak{p}[i^{(l)}]}$ .

On note  $f \in \mathcal{C}_G[\mathfrak{p}, \mathfrak{q}]$ .

*Remarque 7.* Par rapport à la définition des applications cylindres, on a ajouté dans les familles  $(i^{(l)}, \mathbf{u}^{(l)})$  et  $(j^{(l)}, \mathbf{v}^{(l)})$  les indices  $i^{(l)}$  et  $j^{(l)}$ . Lorsque nous définirons les diagrammes, ces indices indiqueront les fils utilisés en entrée ou en sortie : si  $f : (k, x) \in X[\mathfrak{q}] \mapsto (k', x')$ , cela revient graphiquement à placer  $x$  au  $k$ -ième fil d'entrée pour obtenir  $x'$  au  $k'$ -ième fil de sortie.

Les applications cylindres généralisées, comme les applications cylindres, sont bijectives. Il suffit aussi d'échanger les familles définissant une application pour obtenir son inverse.

On note  $\text{id}_{X[\mathfrak{p}]}$  l'identité sur  $X[\mathfrak{p}]$ . Elle peut se définir avec des familles vides ( $L = \emptyset$ ). ◁

Lorsque  $f$  et  $g$  sont des applications cylindres généralisées dont l'image de l'une est le domaine de définition de l'autre, on peut les composer séquentiellement (composition usuelle des fonctions, définition 8). Dans ce cas,  $g \circ f$  est aussi une application cylindre généralisée (proposition 4).

Afin de le démontrer, il faut adapter la notion de raffinement de la définition 5 aux applications cylindres généralisées (définition 9).

**Définition 8.** La *composition séquentielle* de deux fonctions  $f : X[\mathfrak{p}] \rightarrow X[\mathfrak{q}]$  et  $g : X[\mathfrak{q}] \rightarrow X[\mathfrak{r}]$  consiste à appliquer l'une puis l'autre :

$$g \circ f : (k, x) \in X[\mathfrak{p}] \mapsto g(\underbrace{f(k, x)}_{\in X[\mathfrak{q}]}).$$

**Définition 9.** Les familles  $(i^{(l,k)}, \mathbf{u}^{(l,k)})_{l \in L, k \in K(l)}$ ,  $(j^{(l,k)}, \mathbf{v}^{(l,k)})_{l \in L, k \in K(l)}$  forment un *raffinement* de  $f \in \mathcal{C}_G[\mathfrak{p}, \mathfrak{q}]$  (définie par  $(i^{(l)}, \mathbf{u}^{(l)})_{l \in L}$ ,  $(j^{(l)}, \mathbf{v}^{(l)})_{l \in L}$ ) lorsque :

- $\{(i^{(l,k)}, [\mathbf{u}^{(l,k)}]_X) \mid k \in K(l)\}$  est une partition de  $(i^{(l)}, [\mathbf{u}^{(l)}]_X)$ , pour tout  $l \in L$ ,
- $\{(j^{(l,k)}, [\mathbf{v}^{(l,k)}]_X) \mid k \in K(l)\}$  est une partition de  $(j^{(l)}, [\mathbf{v}^{(l)}]_X)$ , pour tout  $l \in L$ ,
- $(\mathbf{u}^{(l,k)})^{-1}X_{\mathfrak{p}[i^{(l,k)}]} = (\mathbf{v}^{(l,k)})^{-1}X_{\mathfrak{q}[j^{(l,k)}]}$ , pour tous  $l \in L$  et  $k \in K(l)$ .

Alors,  $f'$  définie par  $(i^{(l,k)}, \mathbf{u}^{(l,k)})_{l \in L, k \in K(l)}$ ,  $(j^{(l,k)}, \mathbf{v}^{(l,k)})_{l \in L, k \in K(l)}$  est une application cylindre généralisée telle que  $f' = f$ .

**Proposition 4.** Soient  $\mathfrak{p}, \mathfrak{q}, \mathfrak{r}$  des listes de sommets et  $f \in \mathcal{C}_G[\mathfrak{p}, \mathfrak{q}]$ ,  $g \in \mathcal{C}_G[\mathfrak{q}, \mathfrak{r}]$  deux applications cylindres généralisées. Alors,  $g \circ f \in \mathcal{C}_G[\mathfrak{p}, \mathfrak{r}]$ .

La démonstration, qui consiste à créer deux raffinements compatibles pour pouvoir les composer, se trouve dans l'annexe B.1.

*Remarque 8.* Avec  $\mathfrak{p}$  une liste de sommets,  $\mathcal{C}_G[\mathfrak{p}, \mathfrak{p}]$  est un groupe : on peut inverser et composer les applications cylindres généralisées. En effet, l'inverse obtenu en échangeant les familles  $(i^{(l)}, \mathbf{u}^{(l)})$  et  $(j^{(l)}, \mathbf{v}^{(l)})$  est bien l'inverse pour la composition séquentielle. ◁

On montre maintenant que la notion d'application cylindre généralisée est une généralisation des applications cylindres. Lorsque  $\mathfrak{p}$  et  $\mathfrak{q}$  sont des listes qui contiennent une et une seule fois chaque sommet de  $G$ , on obtient un groupe isomorphe au groupe plein topologique  $\Gamma_G$ .

**Proposition 5.** Avec  $Q = \{q_1, \dots, q_n\}$  et en fixant  $\mathfrak{r} = \langle q_1, \dots, q_n \rangle$ ,  $\mathcal{C}_G[\mathfrak{r}, \mathfrak{r}]$  décrit exactement les applications cylindres :

$$\mathcal{C}_G[\langle q_1, \dots, q_n \rangle, \langle q_1, \dots, q_n \rangle] \simeq \Gamma_X$$

La démonstration se trouve dans l'annexe B.2.

Idée de la preuve : on va exhiber un homéomorphisme  $h \in \text{Homeo}(X, X[\mathfrak{r}])$ , puis on montrera que les définitions 4 et 7 des applications cylindres et des applications cylindres généralisées dans le cas  $\mathfrak{p} = \mathfrak{q} = \langle q_1, \dots, q_n \rangle$  sont équivalentes à homéomorphisme  $h$  près. C'est-à-dire,  $\tau \in \Gamma_X \iff h \circ \tau \circ h^{-1} \in \mathcal{C}_G[\mathfrak{r}, \mathfrak{r}]$ . Dans ce cas,  $\varphi : \tau \mapsto h \circ \tau \circ h^{-1}$  est un isomorphisme entre les groupes  $\Gamma_X$  et  $\mathcal{C}_G[\mathfrak{r}, \mathfrak{r}]$ .

## 4.2 PROP colorées

Nous définissons ici les catégories appelées PROP colorées [CP20]. Les applications cylindres généralisées sur  $X_G$  ont la structure de PROP colorée (théorème 1).

**Définition 10.** Une PROP colorée  $\mathbf{P}$  est la donnée d'un ensemble  $C$  de couleurs et d'un ensemble de morphismes  $\mathbf{P}[\mathfrak{p}, \mathfrak{q}]$  pour toute paire  $(\mathfrak{p}, \mathfrak{q}) \in (C^*)^2$  de listes de couleurs. Elle doit vérifier les propriétés suivantes :

- l'existence d'une composition séquentielle  $\circ : \mathbf{P}[\mathfrak{q}, \mathfrak{r}] \times \mathbf{P}[\mathfrak{p}, \mathfrak{q}] \rightarrow \mathbf{P}[\mathfrak{p}, \mathfrak{r}]$  sur les morphismes qui est associative,
- l'existence d'une composition parallèle  $\otimes : \mathbf{P}[\mathfrak{p}, \mathfrak{q}] \times \mathbf{P}[\mathfrak{p}', \mathfrak{q}'] \rightarrow \mathbf{P}[\mathfrak{p} + \mathfrak{p}', \mathfrak{q} + \mathfrak{q}']$  sur les morphismes qui est associative,
- pour toute couleur  $c \in C$ , l'existence d'un morphisme identité  $\text{id}_{\langle c \rangle} \in \mathbf{P}[\langle c \rangle, \langle c \rangle]$ ,
- l'existence d'un morphisme identité pour la liste vide  $\text{id}_{\circ} \in \mathbf{P}[\circ, \circ]$ ,
- pour tous  $c, c' \in C$ , l'existence d'un morphisme permutation  $\text{swap}_{\langle c \rangle, \langle c' \rangle} \in \mathbf{P}[\langle c, c' \rangle, \langle c', c \rangle]$ ,
- pour tous  $f \in \mathbf{P}[\mathfrak{p}, \mathfrak{q}]$ ,  $g \in \mathbf{P}[\mathfrak{q}, \mathfrak{r}]$ ,  $f' \in \mathbf{P}[\mathfrak{p}', \mathfrak{q}']$  et  $g' \in \mathbf{P}[\mathfrak{q}', \mathfrak{r}']$ ,  $(g \circ f) \otimes (g' \circ f') = (g \otimes g') \circ (f \otimes f')$ ,
- pour tout  $f \in \mathbf{P}[\mathfrak{p}, \mathfrak{q}]$ ,  $f \circ \text{id}_{\mathfrak{p}} = \text{id}_{\mathfrak{q}} \circ f = f$ ,
- pour tout  $f \in \mathbf{P}[\mathfrak{p}, \mathfrak{q}]$ ,  $f \otimes \text{id}_{\circ} = \text{id}_{\circ} \otimes f = f$ ,
- pour tous  $c, c' \in C$ ,  $\text{swap}_{\langle c' \rangle, \langle c \rangle} \circ \text{swap}_{\langle c \rangle, \langle c' \rangle} = \text{id}_{\langle c, c' \rangle}$ ,
- pour tous  $f \in \mathbf{P}[\mathfrak{p}, \mathfrak{q}]$ ,  $c \in C$ ,  $\text{swap}_{\mathfrak{q}, \langle c \rangle} \circ (f \otimes \text{id}_{\langle c \rangle}) = (\text{id}_{\langle c \rangle} \otimes f) \circ \text{swap}_{\mathfrak{p}, \langle c \rangle}$ .

Où les morphismes usuels sont définis récursivement par :

- $\text{id}_{\circ} = \text{id}_{\circ}$  et  $\text{id}_{\mathfrak{p} + \langle c \rangle} = \text{id}_{\mathfrak{p}} \otimes \text{id}_{\langle c \rangle}$ ,
- $\text{swap}_{\circ, \langle c \rangle} = \text{id}_{\langle c \rangle}$  et  $\text{swap}_{\mathfrak{p} + \langle c \rangle, \langle c' \rangle} = (\text{swap}_{\mathfrak{p}, \langle c' \rangle} \otimes \text{id}_{\langle c \rangle}) \circ (\text{id}_{\mathfrak{p}} \otimes \text{swap}_{\langle c \rangle, \langle c' \rangle})$ ,
- $\text{swap}_{\mathfrak{p}, \circ} = \text{id}_{\mathfrak{p}}$  et  $\text{swap}_{\mathfrak{p}, \mathfrak{p}' + \langle c \rangle} = (\text{id}_{\mathfrak{p}'} \otimes \text{swap}_{\mathfrak{p}, \langle c \rangle}) \circ (\text{swap}_{\mathfrak{p}, \mathfrak{p}'} \otimes \text{id}_{\langle c \rangle})$ .

On notera  $f : \mathfrak{p} \rightarrow \mathfrak{q}$  pour  $f \in \mathbf{P}[\mathfrak{p}, \mathfrak{q}]$ .

*Remarque 9.* En théorie des catégories, la définition suivante est équivalente : une PROP colorée est une petite catégorie monoïdale stricte et symétrique dont le monoïde des objets est engendré librement par un ensemble de couleurs  $C$ .

Le terme *PROP* signifie *produit* et *permutation*. ◁

Avant de montrer que les applications cylindres généralisées sont les morphismes d'une PROP, il faut définir leur composition parallèle.

**Définition 11.** La composition parallèle  $f \otimes f'$  de deux fonctions  $f : X[\mathfrak{p}] \rightarrow X[\mathfrak{q}]$  et  $f' : X[\mathfrak{p}'] \rightarrow X[\mathfrak{q}']$  est définie ainsi :

$$f \otimes f' : X[\mathfrak{p} + \mathfrak{p}'] \rightarrow X[\mathfrak{q} + \mathfrak{q}']$$

$$(k, x) \mapsto \begin{cases} f(k, x) & \text{si } k \in \llbracket 1, |\mathfrak{p}| \rrbracket \\ +^{|\mathfrak{q}'|} f'(k - |\mathfrak{p}|, x) & \text{si } k \in \llbracket |\mathfrak{p}| + 1, |\mathfrak{p}| + |\mathfrak{p}'| \rrbracket. \end{cases}$$

Où  $+^{|\mathfrak{q}'|} : X[\mathfrak{q}'] \rightarrow X[\mathfrak{q} + \mathfrak{q}']$  telle que pour tout  $(k, x) \in X[\mathfrak{q}']$ ,  $+^{|\mathfrak{q}'|}(k, x) := (|\mathfrak{q}'| + k, x)$  (voir annexe A).

Suivant la valeur de  $k$ , on applique soit l'application  $f$ , soit l'application  $f'$  en appliquant le décalage nécessaire ( $-|p|$  avant et  $+|q|$  après).

On obtient aisément les deux propositions suivantes démontrées dans l'annexe B.3.

**Proposition 6.** *La composition parallèle est associative.*

De plus, la composition parallèle de deux applications cylindres généralisées est une application cylindre généralisée.

**Proposition 7.** *Soient  $f \in C_G[p, q]$  et  $f' \in C_G[p', q']$ . Alors,  $f \otimes f' \in C_G[p + p', q + q']$ .*

Munies des compositions séquentielle et parallèle (définitions 8 et 11), les applications cylindres généralisées forment une PROP.

**Théorème 1.** *On définit les fonctions suivantes, pour tous  $q, q' \in Q$  :*

$$\begin{array}{lll} \text{id}_{\langle q \rangle} : X[\langle q \rangle] \rightarrow X[\langle q \rangle] & \text{id}_\circ : \emptyset \rightarrow \emptyset & \text{swap}_{\langle q \rangle, \langle q' \rangle} : X[\langle q, q' \rangle] \rightarrow X[\langle q', q \rangle]. \\ (1, x) \mapsto (1, x) & & (k, x) \mapsto (3 - k, x) \end{array}$$

Alors,  $C_G$  est une PROP colorée avec  $C = Q$  l'ensemble des couleurs et où les applications cylindres généralisées de  $C_G[p, q]$  sont les morphismes entre les listes de sommets  $p$  et  $q$ .

La démonstration de ce théorème consiste à vérifier point par point les conditions de la définition 10. Elle se trouve dans l'annexe C.1.

### 4.3 Diagrammes

Nous définissons ici formellement les diagrammes présentés dans la section 3 comme une PROP colorée  $\mathcal{D}_G$  définie à partir de diagrammes élémentaires : les générateurs.

Le degré sortant d'un sommet  $q \in Q$  est le nombre d'arêtes qui partent de  $q$  :  $\deg(q) := |s^{-1}(q)|$ .

On ordonne les arêtes qui partent du sommet  $q$ , elles seront désignées par  $\alpha_q(1), \dots, \alpha_q(\deg(q)) \in s^{-1}(q)$ . Pour tout  $j \in \llbracket 1, \deg(q) \rrbracket$ , on pose  $\varphi_q(j) := t(\alpha_q(j))$ , le sommet d'arrivée de  $\alpha_q(j)$ .

Ainsi, on a dans  $G$  :  $q \xrightarrow{\alpha_q(j)} \varphi_q(j)$ .

On note  $\varphi_q := \langle \varphi_q(1), \dots, \varphi_q(\deg(q)) \rangle$  la liste des sommets voisins  $q$  (dans l'ordre des arêtes issues de  $q$ , avec répétitions s'il existe  $j \neq j', \varphi_q(j) = \varphi_q(j')$ ).

**Définition 12.** On définit la PROP colorée  $\mathcal{D}_G$  dont les objets sont des listes de sommets et les morphismes sont compositions séquentielles et parallèles des générateurs lecture et écriture, des identités et des permutations.

- Les générateurs lecture sont les diagrammes  $q \leftarrow \begin{array}{c} \varphi_q(1) \\ \vdots \\ \varphi_q(\deg(q)) \end{array} \in \mathcal{D}_G[\langle q \rangle, \varphi_q]$ , pour  $q \in Q$ .
- Les générateurs écriture sont les diagrammes  $\begin{array}{c} \varphi_q(1) \\ \vdots \\ \varphi_q(\deg(q)) \end{array} \rightarrow q \in \mathcal{D}_G[\varphi_q, \langle q \rangle]$ , pour  $q \in Q$ .
- Les morphismes identité sont les diagrammes  $q \longrightarrow q \in \mathcal{D}_G[\langle q \rangle, \langle q \rangle]$ , pour  $q \in Q$ .
- Les morphismes permutation sont les diagrammes  $\begin{array}{c} q \\ q' \end{array} \begin{array}{c} \curvearrowright \\ \curvearrowleft \end{array} \begin{array}{c} q' \\ q \end{array} \in \mathcal{D}_G[\langle q, q' \rangle, \langle q', q \rangle]$ , pour  $q, q' \in Q$ .

- La composition séquentielle de  $\mathfrak{p} \begin{array}{c} \boxed{D} \\ \vdots \\ \mathfrak{q} \end{array}$  et  $\mathfrak{q} \begin{array}{c} \boxed{E} \\ \vdots \\ \mathfrak{r} \end{array}$  est notée  $\mathfrak{p} \begin{array}{c} \boxed{D} \boxed{E} \\ \vdots \\ \mathfrak{r} \end{array}$ .
- La composition parallèle de  $\mathfrak{p} \begin{array}{c} \boxed{D} \\ \vdots \\ \mathfrak{q} \end{array}$  et  $\mathfrak{p}' \begin{array}{c} \boxed{D'} \\ \vdots \\ \mathfrak{q}' \end{array}$  est notée  $\begin{array}{c} \mathfrak{p} \begin{array}{c} \boxed{D} \\ \vdots \\ \mathfrak{q} \end{array} \\ \mathfrak{p}' \begin{array}{c} \boxed{D'} \\ \vdots \\ \mathfrak{q}' \end{array} \end{array}$ .

*Remarque 10.* Pour garder une définition générale, on n'a pas dessiné de couleurs sur les fils, mais elle est indiquée par le sommet attaché au fil.

Les identités sur des listes sont notés  $\mathfrak{q} \begin{array}{c} \overline{\quad} \\ \vdots \\ \mathfrak{q} \end{array}$ , pour  $\mathfrak{q}$  liste de sommets, avec en particulier le diagramme vide pour  $\mathfrak{p} = \mathfrak{o}$ .

Les permutations sur des listes sont notés  $\begin{array}{c} \mathfrak{p} \begin{array}{c} \vdots \\ \vdots \\ \vdots \end{array} \\ \mathfrak{q} \begin{array}{c} \vdots \\ \vdots \\ \vdots \end{array} \end{array} \begin{array}{c} \mathfrak{q} \\ \vdots \\ \mathfrak{p} \end{array}$ , pour  $\mathfrak{p}, \mathfrak{q}$  listes de sommets. ◁

**Définition 13.** On définit une relation d'équivalence entre diagrammes induite par les relations suivantes, pour tout  $q \in Q$  :

$$\varphi_q \begin{array}{c} \curvearrowright \\ \vdots \\ \curvearrowleft \end{array} \begin{array}{c} \curvearrowleft \\ \vdots \\ \curvearrowright \end{array} \varphi_q = \varphi_q \begin{array}{c} \overline{\quad} \\ \vdots \\ \varphi_q \end{array} \varphi_q, \quad \text{et} \quad q \begin{array}{c} \curvearrowleft \\ \vdots \\ \curvearrowright \end{array} q = q \begin{array}{c} \curvearrowright \\ \vdots \\ \curvearrowleft \end{array} q = q \text{ — } q.$$

#### 4.4 Équivalence entre les PROPs $\mathcal{C}_G$ et $\mathcal{D}_G$

Dans cette dernière partie, nous montrons l'équivalence entre les PROPs  $\mathcal{C}_G$  et  $\mathcal{D}_G$ , en donnant une interprétation aux diagrammes à l'aide d'un foncteur  $F : \mathcal{D}_G \Rightarrow \mathcal{C}_G$ . Ce foncteur sera défini sur les générateurs lecture et écriture de  $\mathcal{D}_G$ . Nous montrerons qu'il est bien défini (*sound*), qu'il est surjectif (*full*) et injectif (*faithful*).

Définissons tout d'abord les interprétations des générateurs de  $\mathcal{D}_G$ .

**Définition 14.** On associe à chaque sommet  $q \in Q$  les fonctions suivantes :

$$\begin{array}{ll} \text{pop}_q : X[\langle q \rangle] \rightarrow X[\varphi_q] & \text{push}_q : X[\varphi_q] \rightarrow X[\langle q \rangle] \\ (1, \alpha_q(j) \cdot z) \mapsto (j, z) & (j, z) \mapsto (1, \alpha_q(j) \cdot z). \end{array}$$

Elles sont bien définies :

- si  $(1, \alpha_q(j) \cdot z) \in X[\langle q \rangle]$ , alors  $\alpha_q(j) \cdot z \in X_q$  et  $s(z_0) = t(\alpha_q(j)) = \varphi_q(j) = \varphi_q[j]$  donc  $z \in X_{\varphi_q[j]}$ , c'est-à-dire  $(j, z) \in X[\varphi_q]$ ,
- si  $(j, z) \in X[\varphi_q]$ , alors  $s(z_0) = \varphi_q[j] = \varphi_q(j)$  et  $t(\alpha_q(j)) = \varphi_q(j)$  donc  $\alpha_q(j) \cdot z \in X_{s(\alpha_q(j))} = X_q$ , c'est-à-dire  $(1, \alpha_q(j) \cdot z) \in X[\langle q \rangle]$ .

*Remarque 11.* Ce sont des applications cylindres généralisées :  $\text{pop}_q$  est définie par  $L = \llbracket 1, \text{deg}(q) \rrbracket$  et les familles  $(1, \alpha_q(j))_{j \in \llbracket 1, \text{deg}(q) \rrbracket}$  et  $(j, \varepsilon)_{j \in \llbracket 1, \text{deg}(q) \rrbracket}$ .

Puisqu'elles sont inverses l'une de l'autre ( $\text{pop}_q \circ \text{push}_q = \text{id}_{\varphi_q}$  et  $\text{push}_q \circ \text{pop}_q = \text{id}_q$ ),  $\text{push}_q$  est définie en échangeant le rôle des deux familles.

Ces applications qui correspondent aux générateurs lecture et écriture des parties 3.1 et 4.3 (voir en particulier l'interprétation donnée à ces diagrammes, figures 12 et 13). ◁

Ainsi, nous pouvons définir le foncteur  $F$  qui donne une interprétation aux diagrammes de  $\mathcal{D}_G$ .

**Définition 15.** Soit  $F : \mathcal{D}_G \Rightarrow \mathcal{C}_G$  un foncteur tel que, pour tous  $\mathfrak{p}$  liste de sommets et  $q \in Q$  :

- $F : \mathfrak{p} \in \text{Ob}(\mathcal{D}_G) \mapsto \mathfrak{p} \in \text{Ob}(\mathcal{C}_G)$ ,

$$\begin{aligned}
- F \left( q \begin{array}{c} \curvearrowright \\ \vdots \\ \varphi_q \end{array} \right) &= \text{pop}_q, \\
- F \left( \begin{array}{c} \varphi_q \\ \vdots \end{array} \curvearrowright q \right) &= \text{push}_q.
\end{aligned}$$

**Soundness**  $F$  est bien défini : il faut vérifier qu'il est compatible avec les relations sur les diagrammes.

Soit  $q \in Q$ , on a :

$$\begin{aligned}
F \left( \begin{array}{c} \varphi_q \\ \vdots \end{array} \curvearrowright \begin{array}{c} \curvearrowleft \\ \vdots \\ \varphi_q \end{array} \right) &= F \left( q \begin{array}{c} \curvearrowleft \\ \vdots \\ \varphi_q \end{array} \right) \circ F \left( \begin{array}{c} \varphi_q \\ \vdots \end{array} \curvearrowright q \right) \\
&= \text{pop}_q \circ \text{push}_q = \text{id}_{\varphi_q} = F \left( \varphi_q \begin{array}{c} \text{---} \\ \vdots \\ \varphi_q \end{array} \right),
\end{aligned}$$

et

$$\begin{aligned}
F \left( q \begin{array}{c} \curvearrowleft \\ \vdots \\ \curvearrowright \end{array} q \right) &= F \left( \begin{array}{c} \varphi_q \\ \vdots \end{array} \curvearrowright q \right) \circ F \left( q \begin{array}{c} \curvearrowleft \\ \vdots \\ \varphi_q \end{array} \right) \\
&= \text{push}_q \circ \text{pop}_q = \text{id}_q = F \left( q \text{---} q \right).
\end{aligned}$$

**Fullness** La surjectivité de  $F$  est obtenue par le théorème 2 : les morphismes de  $\mathcal{C}_G$  (c'est-à-dire, les applications cylindres généralisées) sont générées par les fonctions  $\text{pop}_q$  et  $\text{push}_q$ .

**Théorème 2.** *Toute application cylindre généralisée  $f \in \mathcal{C}_G[\mathfrak{p}, \mathfrak{q}]$  peut s'écrire comme composition parallèle et séquentielle de fonctions  $\text{pop}_q$  et  $\text{push}_q$ , de permutations ( $\text{swap}_{\tau, \tau'}$ ) et d'identités ( $\text{id}_\tau$ ).*

La démonstration, par récurrence sur  $N = \sum_{l \in L} |\mathbf{u}^{(l)}| + |\mathbf{v}^{(l)}|$ , se trouve dans l'annexe C.2.

Ainsi, à partir de cette écriture de  $f \in \mathcal{C}_G[\mathfrak{p}, \mathfrak{q}]$ , on peut obtenir aisément un diagramme  $\boxed{D}$  tel que  $F \left( \boxed{D} \right) = f$  et la surjectivité de  $F$ .

**Faithfulness** L'injectivité de  $F$  est l'objet du théorème 3. La démonstration de ce théorème, qui utilise deux lemmes, ainsi que les notions de diagrammes gauche, droite, et de miroir d'un diagramme, se situe dans l'annexe C.3.

**Définition 16.** Un diagramme gauche est un diagramme sans générateurs écriture. Un tel diagramme est noté  $\boxed{D}$ .

Inversement, un diagramme droite est un diagramme sans générateurs lecture. Il est noté  $\boxed{D}$ .

On définit inductivement le miroir  $\left( \boxed{D} \right)^\dagger \in \mathcal{D}_G[\mathfrak{q}, \mathfrak{p}]$  d'un diagramme  $\boxed{D} \in \mathcal{D}_G[\mathfrak{p}, \mathfrak{q}]$  par :

$$\begin{aligned}
\left( \begin{array}{c} \varphi_q \\ \vdots \end{array} \curvearrowright q \right)^\dagger &= q \begin{array}{c} \curvearrowleft \\ \vdots \\ \varphi_q \end{array} & \left( q \begin{array}{c} \curvearrowleft \\ \vdots \\ \varphi_q \end{array} \right)^\dagger &= \begin{array}{c} \varphi_q \\ \vdots \end{array} \curvearrowright q \\
(q \text{---} q)^\dagger &= q \text{---} q & \left( \begin{array}{c} q \\ \curvearrowright \\ q' \\ \curvearrowleft \\ q \end{array} \right)^\dagger &= \begin{array}{c} q' \\ \curvearrowright \\ q \\ \curvearrowleft \\ q \end{array} \\
\left( \mathfrak{p} \begin{array}{c} \vdots \\ \boxed{D} \\ \vdots \\ \boxed{E} \\ \vdots \\ \tau \end{array} \right)^\dagger &= \tau \begin{array}{c} \vdots \\ \boxed{E^\dagger} \\ \vdots \\ \boxed{D^\dagger} \\ \vdots \\ \mathfrak{p} \end{array} & \left( \mathfrak{p} \begin{array}{c} \vdots \\ \boxed{D} \\ \vdots \\ \mathfrak{q} \end{array} \right)^\dagger &= \mathfrak{q} \begin{array}{c} \vdots \\ \boxed{D^\dagger} \\ \vdots \\ \mathfrak{p} \end{array} \\
& & \left( \mathfrak{p}' \begin{array}{c} \vdots \\ \boxed{E} \\ \vdots \\ \mathfrak{q}' \end{array} \right)^\dagger &= \mathfrak{q}' \begin{array}{c} \vdots \\ \boxed{E^\dagger} \\ \vdots \\ \mathfrak{p}' \end{array}.
\end{aligned}$$

On note parfois  $\boxed{D^\dagger}$  au lieu de  $\left(\boxed{D}\right)^\dagger$ .

Le miroir d'un diagramme a les propriétés suivantes qui se démontrent par induction.

**Proposition 8.** Soit  $D \in \mathcal{D}_G[\mathfrak{p}, \mathfrak{q}]$ . Alors :

- $\left(\boxed{D^\dagger}\right)^\dagger = \boxed{D}$ ,
- $\boxed{D} \boxed{D^\dagger} = \mathfrak{p} \text{ --- } \mathfrak{p}$  et  $\boxed{D^\dagger} \boxed{D} = \mathfrak{q} \text{ --- } \mathfrak{q}$ ,
- Si  $\boxed{D}$  est un diagramme gauche, alors  $\boxed{D^\dagger}$  est un diagramme droite et inversement.

La démonstration du théorème 3 repose sur les deux lemmes suivants adaptés de [Jea24].

**Lemme 1.** Soit  $\boxed{D} \in \mathcal{D}_G[\mathfrak{p}, \mathfrak{q}]$ .

Alors, il existe deux diagrammes gauche et droite  $\boxed{D_1}, \boxed{D_2} \in \mathcal{D}_G$  tels que  $\boxed{D} = \boxed{D_1} \boxed{D_2}$ .

**Lemme 2.** Soit  $\boxed{D}, \boxed{E} \in \mathcal{D}_G[\mathfrak{p}, \mathfrak{q}]$  deux diagrammes gauches tels que  $F(\boxed{D}) = F(\boxed{E})$ .

Alors,  $\boxed{D} = \boxed{E}$ .

**Théorème 3.** Soit  $\boxed{D}, \boxed{E} \in \mathcal{D}_G[\mathfrak{p}, \mathfrak{q}]$  tels que  $F(\boxed{D}) = F(\boxed{E})$ .

Alors,  $\boxed{D} = \boxed{E}$ .

La démonstration du théorème se trouve dans l'annexe C.3.

## 5 Conclusion

Le résultat principal de ce stage est le formalisme de diagrammes pour représenter les applications cylindres généralisées. Celles-ci généralisent les applications cylindres, des homéomorphismes sur les subshifts qui forment le groupe plein topologique d'un subshift. Les diagrammes sont donc les éléments d'une généralisation du groupe plein topologique qui a une structure de groupoïde et dont la définition usuelle est complètement topologique [GPS99; Kat19; Mat25; MM17].

Notre travail s'applique sur une classe particulière de subshifts, les edge shifts. Il serait intéressant de travailler sur les classes plus générales des subshifts de type finis ou des subshifts sofiqes. Pour cela, il faudrait considérer d'autres partitions d'un subshift que celle utilisée qui ne considèrerait que le sommet de départ dans le graphe, par exemple trouver une décomposition finie de langages quotients dans le cas des sofiqes.

Un autre axe de généralisation du sujet est celui des subshifts de mots bi-infinis. La définition du groupe plein topologique est un peu différente : au lieu de remplacer un préfixe par un autre, les homéomorphismes traduisent les mots bi-infinis continuellement (c'est-à-dire, la translation d'un mot dépend d'un nombre fini de lettres du mot). Ce cas ressemble davantage à celui des machines de Turing qui ne peuvent pas modifier les symboles du ruban.

## Références

- [BM14] James BELK et Francesco MATUCCI. « Conjugacy and dynamics in Thompson's groups ». In : *Geometriae Dedicata* (2014), p. 239-261. DOI : [10.1007/s10711-013-9853-2](https://doi.org/10.1007/s10711-013-9853-2).
- [Boy08] Mike BOYLE. « Open Problems in Symbolic Dynamics ». In : *Geometric and probabilistic structures in dynamics*. American Mathematical Society, 2008, p. 69-118.
- [Bri22] Kevin A. BRIX. *Invertible and noninvertible symbolic dynamics and their C\*-algebras*. 2022. DOI : <https://doi.org/10.48550/arXiv.2212.11188>.
- [CFP96] James W. CANNON, William J. FLOYD et Walter R. PARRY. « Introductory Notes on Richard Thompson's Groups ». In : *L'Enseignement Mathématique*. T. 44. 1996, p. 251-256.
- [CK17] Bob COECKE et Aleks KISSINGER. *Picturing Quantum Processes*. Cambridge University Press, 2017. DOI : [10.1017/9781316219317](https://doi.org/10.1017/9781316219317).
- [CP20] Titouan CARETTE et Simon PERDRIX. *Colored props for large scale graphical reasoning*. 2020. DOI : [10.48550/arXiv.2007.03564](https://doi.org/10.48550/arXiv.2007.03564).
- [GPS99] Thierry GIORDANO, Ian F. PUTNAM et Christian F. SKAU. « Full groups of Cantor minimal systems ». In : *Israel Journal of Mathematics* 111 (1999), p. 285-320. DOI : [10.1007/BF02810689](https://doi.org/10.1007/BF02810689).
- [Jea21] Emmanuel JEANDEL. *Strong shift equivalence as a category notion*. 2021. DOI : [10.48550/arXiv.2107.10734](https://doi.org/10.48550/arXiv.2107.10734).
- [Jea24] Emmanuel JEANDEL. « A categorical approach to reversible Turing machines and Brin-Thompson groups ». (unpublished). 2024.
- [JS91] André JOYAL et Ross STREET. « The geometry of tensor calculus, I ». In : *Advances in Mathematics* 88.1 (1991), p. 55-112. DOI : [10.1016/0001-8708\(91\)90003-P](https://doi.org/10.1016/0001-8708(91)90003-P).
- [Kat19] Leonhard KATZLINGER. *Topological Full Groups*. 2019. DOI : [10.48550/arXiv.1907.07424](https://doi.org/10.48550/arXiv.1907.07424).
- [Kit98] Bruce P. KITCHENS. *Symbolic dynamics*. Springer Berlin, 1998. DOI : [10.1007/978-3-642-58822-8](https://doi.org/10.1007/978-3-642-58822-8).
- [Kûr03] Petr KÛRKA. *Topological and symbolic dynamics*. T. 11. Cours spécialisés. Société Mathématique de France, 2003. ISBN : 2-85629-143-0.
- [LM95] Douglas LIND et Brian MARCUS. *An Introduction to Symbolic Dynamics and Coding*. Cambridge University Press, 1995.
- [Mat15] Hiroki MATUI. « Topological full groups of one-sided shifts of finite type ». In : *Journal für die reine und angewandte Mathematik (Crelles Journal)* 2015.705 (2015), p. 35-84. DOI : [10.1515/crelle-2013-0041](https://doi.org/10.1515/crelle-2013-0041).
- [Mat25] Kengo MATSUMOTO. *Symbolic Dynamical Systems and C\*-Algebras*. Mathematical Physics Studies. Springer Singapore, 2025. DOI : [10.1007/978-981-97-9404-1](https://doi.org/10.1007/978-981-97-9404-1).
- [MH40] Marston MORSE et Gustav A. HEDLUND. « Symbolic Dynamics II. Sturmian Trajectories ». In : *American Journal of Mathematics* 62 (1940). DOI : [10.2307/2371431](https://doi.org/10.2307/2371431).
- [MM17] Kengo MATSUMOTO et Hiroki MATUI. « Full groups of Cuntz-Krieger algebras and Higman-Thompson groups ». In : *Groups, Geometry and Dynamics*. T. 11. 2. 2017, p. 499-531. DOI : [10.4171/GGD/405](https://doi.org/10.4171/GGD/405).
- [Sel11] Peter SELINGER. « A Survey of Graphical Languages for Monoidal Categories ». In : *New Structures for Physics*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2011, p. 289-355. ISBN : 978-3-642-12821-9. DOI : [10.1007/978-3-642-12821-9\\_4](https://doi.org/10.1007/978-3-642-12821-9_4).
- [Wil73] R. F. WILLIAMS. « Classification of Subshifts of Finite Type ». In : *Annals of Mathematics* 98 (1973), p. 120-153. DOI : [10.2307/1970908](https://doi.org/10.2307/1970908).

## Annexe A Sommes directes

La somme directe des ensembles  $A$  et  $B$  est  $A \oplus B := \{(1, a) | a \in A\} \uplus \{(2, b) | b \in B\}$ .

La somme directe d'une famille d'ensembles est définie par :  $\bigoplus_{i \in I} A_i := \biguplus_{i \in I} \{(i, a) | a \in A_i\}$ .

Pour deux familles  $(A_i)_{i \in I}$  et  $(B_j)_{j \in J}$  avec  $I$  et  $J$  disjoints, on pose  $C_k = \begin{cases} A_k & \text{si } k \in I \\ B_k & \text{si } k \in J \end{cases}$  pour tout  $k \in I \uplus J$ . On obtient la somme directe des  $A_i$  et  $B_j$  avec  $\bigoplus_{k \in I \uplus J} C_k$ .

Lorsque les familles sont indexées par  $I = \llbracket 1, n \rrbracket$  et  $J = \llbracket 1, m \rrbracket$ , on pose  $C_k = \begin{cases} A_k & \text{si } k \leq n \\ B_{k-n} & \text{sinon} \end{cases}$  pour tout  $k \in \llbracket 1, n+m \rrbracket$  et la somme directe des  $A_i$  et  $B_j$  est :

$$\bigoplus_{k=1}^{n+m} C_k = \biguplus_{k=1}^{n+m} \left\{ (k, c) \mid \begin{cases} c \in A_k & \text{si } k \leq n \\ c \in B_{k-n} & \text{sinon} \end{cases} \right\}$$

Dans ce cas, on utilisera  $+^n : \bigoplus_{j=1}^m B_j \rightarrow \bigoplus_{k=1}^{n+m} C_k$  telle que  $+^n(k, b) := (k+n, b)$ .

**Cas des familles indexées par des listes** Soit  $(D_q)_{q \in Q}$  et deux listes  $\mathbf{p}, \mathbf{q}$  d'éléments de  $Q$ .

Si les familles  $(A_i)_{i \in \llbracket 1, |\mathbf{p}| \rrbracket}$  et  $(B_j)_{j \in \llbracket 1, |\mathbf{q}| \rrbracket}$  vérifient  $A_i = D_{\mathbf{p}[i]}$  et  $B_j = D_{\mathbf{q}[j]}$ , pour tous  $i \in \llbracket 1, |\mathbf{p}| \rrbracket$  et  $j \in \llbracket 1, |\mathbf{q}| \rrbracket$ , on pose :

$$\bigoplus_{i=1}^{|\mathbf{p}|} D_{\mathbf{p}[i]} := \bigoplus_{i=1}^{|\mathbf{p}|} A_i$$

et

$$\bigoplus_{j=1}^{|\mathbf{q}|} D_{\mathbf{q}[j]} := \bigoplus_{j=1}^{|\mathbf{q}|} B_j.$$

Dans ce cas,

$$\bigoplus_{k=1}^{|\mathbf{p}+\mathbf{q}|} D_{(\mathbf{p}+\mathbf{q})[k]} = \bigoplus_{k=1}^{|\mathbf{p}+\mathbf{q}|} C_k = \biguplus_{k=1}^{|\mathbf{p}+\mathbf{q}|} \{(k, c) | c \in D_{(\mathbf{p}+\mathbf{q})[k]}\}.$$

Avec :

$$C_k = \begin{cases} A_k = D_{\mathbf{p}[k]} = D_{(\mathbf{p}+\mathbf{q})[k]} & \text{si } k \in \llbracket 1, |\mathbf{p}| \rrbracket \\ B_{k-|\mathbf{p}|} = D_{\mathbf{q}[k-|\mathbf{p}|]} = D_{(\mathbf{p}+\mathbf{q})[k]} & \text{sinon.} \end{cases}$$

## Annexe B Preuves des résultats sur les applications cylindres généralisées

### B.1 Composition séquentielle

*Démonstration de la proposition 4.* Soient  $f \in \mathcal{C}[\mathbf{p}, \mathbf{q}]$  et  $g \in \mathcal{C}[\mathbf{q}, \mathbf{r}]$ . Montrons que  $g \circ f \in \mathcal{C}[\mathbf{p}, \mathbf{r}]$ .

Soient  $L_1, (i_1^{(l)}, \mathbf{u}_1^{(l)})_{l \in L_1}$  et  $(j_1^{(l)}, \mathbf{v}_1^{(l)})_{l \in L_1}$  qui définissent  $f$ .

Soient  $L_2, (i_2^{(l)}, \mathbf{u}_2^{(l)})_{l \in L_2}$  et  $(j_2^{(l)}, \mathbf{v}_2^{(l)})_{l \in L_2}$  qui définissent  $g$ .

Pour montrer que  $g \circ f \in \mathcal{C}[\mathbf{p}, \mathbf{r}]$ , il faut trouver  $L$  et deux familles qui définissent  $g \circ f$  en vérifiant les points **a**, **b**, **c** et **d**. On aimerait que les familles  $(j_1^{(l)}, \mathbf{v}_1^{(l)})$  et  $(i_2^{(l)}, \mathbf{u}_2^{(l)})$  soient les mêmes, mais ce

n'est pas le cas en général. Il faut alors créer deux raffinements (un pour  $f$  et un pour  $g$ ) pour lesquels la famille de préfixes écrits  $(j_1, \mathbf{v}_1)$  par  $f$  et la famille de préfixes lus  $(i_2, \mathbf{u}_2)$  par  $g$  sont compatibles.

On crée le raffinement  $(i_1^{(l,k)}, \mathbf{u}_1^{(l,k)})_{l \in L_1, k \in K_1(l)}, (j_1^{(l,k)}, \mathbf{v}_1^{(l,k)})_{l \in L_1, k \in K_1(l)}$  où, pour tout  $l \in L_1$ ,

- Si  $\exists k \in L_2, (j_1^{(l)}, [\mathbf{v}_1^{(l)}]_X) \subseteq (i_2^{(k)}, [\mathbf{u}_2^{(k)}]_X)$ , alors on pose  $K_1(l) = \{\star\}$ ,  $(i_1^{(l,\star)}, \mathbf{u}_1^{(l,\star)}) = (i_1^{(l)}, \mathbf{u}_1^{(l)})$ , et  $(j_1^{(l,\star)}, \mathbf{v}_1^{(l,\star)}) = (j_1^{(l)}, \mathbf{v}_1^{(l)})$ ,
- Sinon, on pose  $K_1(l) = \left\{ k \in L_2 : (i_2^{(k)}, [\mathbf{u}_2^{(k)}]_X) \subset (j_1^{(l)}, [\mathbf{v}_1^{(l)}]_X) \right\}$ , et pour tout  $k \in K_1(l)$ ,  $(i_1^{(l,k)}, \mathbf{u}_1^{(l,k)}) = (i_1^{(l)}, \mathbf{u}_1^{(k)})$ , et  $(j_1^{(l,k)}, \mathbf{v}_1^{(l,k)}) = (j_1^{(l)}, \mathbf{v}_1^{(k)})$ .

C'est un raffinement pour  $f$ . On fait exactement la même chose pour  $g$ .

On crée le raffinement  $(i_2^{(l,k)}, \mathbf{u}_2^{(l,k)})_{l \in L_2, k \in K_2(l)}, (j_2^{(l,k)}, \mathbf{v}_2^{(l,k)})_{l \in L_2, k \in K_2(l)}$  où, pour tout  $l \in L_2$ ,

- Si  $\exists k \in L_1, (i_2^{(l)}, [\mathbf{u}_2^{(l)}]_X) \subseteq (j_1^{(k)}, [\mathbf{v}_1^{(k)}]_X)$ , alors on pose  $K_2(l) = \{\star\}$ ,  $(i_2^{(l,\star)}, \mathbf{u}_2^{(l,\star)}) = (i_2^{(l)}, \mathbf{u}_2^{(l)})$ , et  $(j_2^{(l,\star)}, \mathbf{v}_2^{(l,\star)}) = (j_2^{(l)}, \mathbf{v}_2^{(l)})$ ,
- Sinon, on pose  $K_2(l) = \left\{ k \in L_1 : (j_1^{(k)}, [\mathbf{v}_1^{(k)}]_X) \subset (i_2^{(l)}, [\mathbf{u}_2^{(l)}]_X) \right\}$ , et pour tout  $k \in K_2(l)$ ,  $(i_2^{(l,k)}, \mathbf{u}_2^{(l,k)}) = (i_2^{(l)}, \mathbf{u}_1^{(k)})$ , et  $(j_2^{(l,k)}, \mathbf{v}_2^{(l,k)}) = (j_2^{(l)}, \mathbf{v}_1^{(k)})$ .

C'est un raffinement pour  $g$ .

Finalement, on prend  $L = \{(l, k) : l \in L_1, k \in K_1(l)\}$ . Par définition de ces raffinements, il existe un unique couple  $(l', k')$ ,  $l' \in L_2, k' \in K_2(l')$  tel que  $(j_1^{(l,k)}, \mathbf{v}_1^{(l,k)}) = (i_2^{(l',k')}, \mathbf{u}_2^{(l',k')})$ .

On pose  $(i^{(l,k)}, \mathbf{u}^{(l,k)}) = (i_1^{(l,k)}, \mathbf{u}_1^{(l,k)})$  et  $(j^{(l,k)}, \mathbf{v}^{(l,k)}) = (j_2^{(l',k')}, \mathbf{v}_2^{(l',k')})$  pour tout  $(l, k) \in L$ .

Ainsi,  $L$  et ces deux familles définissent  $g \circ f$  : pour tout  $(l, k) \in L, z \in (\mathbf{u}^{(l,k)})^{-1} X_{\mathbf{p}[i^{(l,k)}]}$ ,

$$\begin{aligned} g \circ f(i^{(l,k)}, \mathbf{u}^{(l,k)} z) &= g \circ f(i_1^{(l,k)}, \mathbf{u}_1^{(l,k)} z) = g(j_1^{(l,k)}, \mathbf{v}_1^{(l,k)} z) \\ &= g(i_2^{(l',k')}, \mathbf{u}_2^{(l',k')} z) = (j_2^{(l',k')}, \mathbf{v}_2^{(l',k')} z) = (j^{(l,k)}, \mathbf{v}^{(l,k)} z). \end{aligned}$$

Donc  $g \circ f \in \mathcal{C}[\mathbf{p}, \mathbf{r}]$ .

Cette preuve repose sur le lemme suivant : si l'intersection de deux cylindres est non vide, alors un d'eux est inclus dans l'autre.  $\square$

## B.2 Isomorphisme avec le groupe plein topologique

*Démonstration de la proposition 5.* Soit  $\mathbf{r} = \langle q_1, \dots, q_n \rangle$ .

Idée de la preuve : on va exhiber un homéomorphisme  $h \in \text{Homeo}(X, X[\mathbf{r}])$ , puis on montrera que les définitions 4 et 7 des applications cylindres et des applications cylindres généralisées dans le cas  $\mathbf{p} = \mathbf{q} = \langle q_1, \dots, q_n \rangle$  sont équivalentes à homéomorphisme  $h$  près. C'est-à-dire,  $\tau \in \Gamma_X \iff h \circ \tau \circ h^{-1} \in \mathcal{C}_G[\mathbf{r}, \mathbf{r}]$ . Dans ce cas,  $\varphi : \tau \mapsto h \circ \tau \circ h^{-1}$  est un isomorphisme entre les groupes  $\Gamma_X$  et  $\mathcal{C}_G[\mathbf{r}, \mathbf{r}]$ .

Soit  $h : X \rightarrow X[\mathbf{r}]$  tel que pour tout  $x \in X, h(x) = (k, x)$  où  $k \in \llbracket 1, n \rrbracket$  est unique vérifiant  $s(x_0) = q_k$ .  $h$  est bien défini : pour tout  $x \in X, x \in X_{s(x_0)} = X_{q_k} = X_{\mathbf{r}[k]}$  car  $s(x_0) = q_k = \mathbf{r}[k]$ , donc  $(k, x) \in X[\mathbf{r}]$ .

$h$  est bijective, son inverse est  $h^{-1} : (k, x) \in X[\mathbf{r}] \mapsto x \in X$ .

Soit  $(k, [\mathbf{u}]_X) \in \mathcal{B}_{X[\mathbf{r}]}$  un cylindre de  $X[\mathbf{r}]$  (voir remarque 6). L'image réciproque  $h^{-1}((k, [\mathbf{u}]_X)) = h^{-1}(\{(k, x) \in X[\mathbf{r}] : x \in [\mathbf{u}]_X\}) = \{x \in X : x \in [\mathbf{u}]_X\} = [\mathbf{u}]_X$  est un cylindre dans  $\mathcal{B}_X$ . Donc  $h$  est continue.

Ainsi  $h \in \text{Homeo}(X, X[\mathfrak{r}])$  (on utilise de nouveau le fait qu'une fonction bijective et continue sur des espaces topologiques compacts est un homéomorphisme).

Soit  $\tau : X \rightarrow X$ . On pose  $\varphi(\tau) = h \circ \tau \circ h^{-1}$ .

Montrons l'équivalence  $\tau \in \Gamma_X \iff h \circ \tau \circ h^{-1} \in \mathcal{C}_G[\mathfrak{r}, \mathfrak{r}]$ .

Supposons que  $\tau \in \Gamma_X$ . Alors, il existe  $L, (\mathbf{u}^{(l)})$  et  $(\mathbf{v}^{(l)})$  telles que  $\tau$  vérifie les points **1**, **2**, **3** et **4** de la définition **4**. Montrons que  $h \circ \tau \circ h^{-1} \in \mathcal{C}_G[\mathfrak{r}, \mathfrak{r}]$ .

- Si  $L = \emptyset$ ,  $X = X[\mathfrak{r}] = \emptyset$  par **1** donc  $h \circ \tau \circ h^{-1}$  est la fonction vide. Avec  $L' = \emptyset$ ,  $h \circ \tau \circ h^{-1}$  vérifie trivialement **a**, **b**, **c** et **d** de la définition **7**.
- S'il existe  $l \in L$  tel que  $|\mathbf{u}^{(l)}| = 0$  ( $\mathbf{u} = \varepsilon$ ), ou  $|\mathbf{v}^{(l)}| = 0$ , alors  $\tau = \text{id}_X$  et  $L = \{\star\}$  (par **1** ou **2**). Donc  $h \circ \tau \circ h^{-1}$  est définie par  $L' = \llbracket 1, n \rrbracket$  et les familles  $(i^{(l)} = l, \mathbf{u}^{(l)} = \varepsilon)_{l \in L'}$  et  $(j^{(l)} = l, \mathbf{v}^{(l)} = \varepsilon)_{l \in L'}$  vérifient les points de la définition **7**.
- Sinon, on pose  $L' = L$  et les familles  $(i^{(l)}, \mathbf{u}^{(l)})_{l \in L'}$  et  $(j^{(l)}, \mathbf{v}^{(l)})_{l \in L'}$ , où  $s(\mathbf{u}_1^{(l)}) = q_{i^{(l)}}$  et  $s(\mathbf{v}_1^{(l)}) = q_{j^{(l)}}$ , pour tout  $l \in L'$ . Montrons que  $h \circ \tau \circ h^{-1}$ ,  $L'$  et les familles  $(i^{(l)}, \mathbf{u}^{(l)})_{L'}$ ,  $(j^{(l)}, \mathbf{v}^{(l)})_{L'}$  vérifient la définition **7**.
  - a. Soit  $(k, x) \in X[\mathfrak{r}]$ .  $x \in X_{\mathfrak{r}[k]} \subseteq X$ . Par **1**, il existe un unique  $l \in L$  tel que  $x \in [\mathbf{u}^{(l)}]_X$ . De plus,  $q_k = s(x_0) = s(\mathbf{u}_1^{(l)}) = q_{i^{(l)}}$  donc  $k = i^{(l)}$ . D'où  $(k, x) \in (i^{(l)}, \mathbf{u}^{(l)})$ . S'il existe  $l', l'' \in L' = L$  tels que  $(k, x) \in (i^{(l')}, \mathbf{u}^{(l')})$  et  $(k, x) \in (i^{(l'')}, \mathbf{u}^{(l'')})$ , alors  $l = l'$  par unicité du  $l \in L$  tel que  $x \in [\mathbf{u}^{(l)}]_X$  (point **1**).
  - Ainsi,  $\{(i^{(l)}, [\mathbf{u}^{(l)}]_X) | l \in L\}$  est une partition de  $X[\mathfrak{r}]$  et le point **a**.
  - b. Le même raisonnement convient pour **b**, en utilisant **2**.
  - c. Soit  $l \in L'$ . Le point **c** est vérifié car :
$$\begin{aligned} (\mathbf{u}^{(l)})^{-1} X_{\mathfrak{r}[i^{(l)}]} &= (\mathbf{u}^{(l)})^{-1} X_{s(\mathbf{u}_1^{(l)})} = (\mathbf{u}^{(l)})^{-1} X && \text{car } \mathfrak{r}[i^{(l)}] = q_{i^{(l)}} = s(\mathbf{u}_1^{(l)}) \\ &= (\mathbf{v}^{(l)})^{-1} X && \text{par } \mathbf{3}. \\ &= (\mathbf{v}^{(l)})^{-1} X_{s(\mathbf{v}_1^{(l)})} = (\mathbf{v}^{(l)})^{-1} X_{\mathfrak{r}[j^{(l)}]} && \text{car } s(\mathbf{v}_1^{(l)}) = q_{j^{(l)}} = \mathfrak{r}[j^{(l)}]. \end{aligned}$$
- d. Soit  $(i^{(l)}, \mathbf{u}^{(l)} z) \in X[\mathfrak{r}]$ . On a :  $h \circ \tau \circ h^{-1}(i^{(l)}, \mathbf{u}^{(l)} z) = h \circ \tau(\mathbf{u}^{(l)} z) = h(\mathbf{v}^{(l)} z) = (j^{(l)}, \mathbf{v}^{(l)} z)$  car  $s(\mathbf{u}_1^{(l)}) = q_{i^{(l)}}$  et  $s(\mathbf{v}_1^{(l)}) = q_{j^{(l)}}$ . D'où **d**.

Ainsi,  $h \circ \tau \circ h^{-1} \in \mathcal{C}_G[\mathfrak{r}, \mathfrak{r}]$ .

La preuve de l'implication réciproque est très semblable. Il faut cependant considérer un raffinement tel que  $|\mathbf{u}^{(l)}| \geq 1$  et  $|\mathbf{v}^{(l)}| \geq 1$  pour tout  $l \in L$ . Ainsi, à partir des familles  $(i^{(l)}, \mathbf{u}^{(l)})$ ,  $(j^{(l)}, \mathbf{v}^{(l)})$ , on prend  $L' = L$ , et les familles  $(\mathbf{u}^{(l)})$ ,  $(\mathbf{v}^{(l)})$ , puis on vérifie les points **1**, **2**, **3** et **4** pour  $\tau$ .

Pour conclure, l'homéomorphisme  $h \in \text{Homeo}(X, X[\mathfrak{r}])$  induit un isomorphisme  $\varphi : \tau \mapsto h \circ \tau \circ h^{-1}$  de  $\Gamma_X$  vers  $\mathcal{C}_G[\mathfrak{r}, \mathfrak{r}]$ , avec  $\varphi^{-1} : f \mapsto h^{-1} \circ f \circ h$ . Donc  $\mathcal{C}_G[\langle q_1, \dots, q_n \rangle, \langle q_1, \dots, q_n \rangle] \simeq \Gamma_X$ .  $\square$

### B.3 Composition parallèle

On démontre ici les propositions **6** (la composition parallèle est associative) et **7** (la composition parallèle de deux applications cylindres généralisées est une application cylindre généralisée).

*Démonstration de la proposition 6.* Soient  $f : X[\mathfrak{p}] \rightarrow X[\mathfrak{q}]$ ,  $g : X[\mathfrak{p}'] \rightarrow X[\mathfrak{q}']$ ,  $h : X[\mathfrak{p}'' ] \rightarrow X[\mathfrak{q}'' ]$ .

Soit  $(k, x) \in X[\mathbf{pp}'\mathbf{p}'']$ . On a, d'une part :

$$\begin{aligned}
(f \otimes (g \otimes h))(k, x) &= \begin{cases} f(k, x) & \text{si } k \in \llbracket 1, |\mathbf{p}| \rrbracket \\ +^{|\mathbf{q}|} (g \otimes h)(k - |\mathbf{p}|, x) & \text{sinon } (k \in \llbracket |\mathbf{p}| + 1, |\mathbf{p}| + |\mathbf{p}'\mathbf{p}''| \rrbracket) \end{cases} \\
&= \begin{cases} f(k, x) & \text{si } k \in \llbracket 1, |\mathbf{p}| \rrbracket \\ \begin{cases} +^{|\mathbf{q}|} g(k - |\mathbf{p}|, x) & \text{si } k - |\mathbf{p}| \in \llbracket 1, |\mathbf{p}'| \rrbracket \\ +^{|\mathbf{q}|} +^{|\mathbf{q}'|} h(k - |\mathbf{p}| - |\mathbf{p}'|, x) & \text{si } k - |\mathbf{p}| \in \llbracket |\mathbf{p}'| + 1, |\mathbf{p}'| + |\mathbf{p}''| \rrbracket \end{cases} & \text{sinon} \end{cases} \\
&= \begin{cases} f(k, x) & \text{si } k \in \llbracket 1, |\mathbf{p}| \rrbracket \\ +^{|\mathbf{q}|} g(k - |\mathbf{p}|, x) & \text{si } k \in \llbracket |\mathbf{p}| + 1, |\mathbf{pp}'| \rrbracket \\ +^{|\mathbf{q}\mathbf{q}'|} h(k - |\mathbf{pp}'|, x) & \text{si } k \in \llbracket |\mathbf{pp}'| + 1, |\mathbf{pp}'\mathbf{p}''| \rrbracket, \end{cases}
\end{aligned}$$

et d'autre part :

$$\begin{aligned}
((f \otimes g) \otimes h)(k, x) &= \begin{cases} (f \otimes g)(k, x) & \text{si } k \in \llbracket 1, |\mathbf{pp}'| \rrbracket \\ +^{|\mathbf{q}\mathbf{q}'|} h(k - |\mathbf{pp}'|, x) & \text{sinon } (k \in \llbracket |\mathbf{pp}'| + 1, |\mathbf{pp}'| + |\mathbf{p}''| \rrbracket) \end{cases} \\
&= \begin{cases} \begin{cases} f(k, x) & \text{si } \llbracket 1, |\mathbf{p}| \rrbracket \\ +^{|\mathbf{q}|} g(k - |\mathbf{p}|, x) & \text{si } \llbracket |\mathbf{p}| + 1, |\mathbf{p}| + |\mathbf{p}'| \rrbracket \end{cases} & \text{si } k \in \llbracket 1, |\mathbf{pp}'| \rrbracket \\ +^{|\mathbf{q}\mathbf{q}'|} h(k - |\mathbf{pp}'|, x) & \text{sinon} \end{cases} \\
&= \begin{cases} f(k, x) & \text{si } k \in \llbracket 1, |\mathbf{p}| \rrbracket \\ +^{|\mathbf{q}|} g(k - |\mathbf{p}|, x) & \text{si } k \in \llbracket |\mathbf{p}| + 1, |\mathbf{pp}'| \rrbracket \\ +^{|\mathbf{q}\mathbf{q}'|} h(k - |\mathbf{pp}'|, x) & \text{si } k \in \llbracket |\mathbf{pp}'| + 1, |\mathbf{pp}'\mathbf{p}''| \rrbracket. \end{cases}
\end{aligned}$$

Ainsi, pour tout  $(k, x) \in X[\mathbf{pp}'\mathbf{p}'']$ ,  $(f \otimes (g \otimes h))(k, x) = ((f \otimes g) \otimes h)(k, x)$ .

La composition parallèle est associative.  $\square$

*Démonstration de la proposition 7.* Soient  $f \in \mathcal{C}_G[\mathbf{p}, \mathbf{q}]$  et  $g \in \mathcal{C}_G[\mathbf{p}', \mathbf{q}']$ . Soient  $L_1, (i_1^{(l)}, \mathbf{u}_1^{(l)})_{l \in L_1}$  et  $(j_1^{(l)}, \mathbf{v}_1^{(l)})_{l \in L_1}$  qui définissent  $f$ , et  $L_2, (i_2^{(l)}, \mathbf{u}_2^{(l)})_{l \in L_2}$  et  $(j_2^{(l)}, \mathbf{v}_2^{(l)})_{l \in L_2}$  qui définissent  $g$ .

Alors on pose  $L = L_1 \oplus L_2$ , puis on définit les familles  $(i^{(l)}, \mathbf{u}^{(l)})_{l \in L_1 \oplus L_2}$  et  $(j^{(l)}, \mathbf{v}^{(l)})_{l \in L_1 \oplus L_2}$  ainsi :

- pour tout  $l_1 \in L_1$ ,  $(i^{(1, l_1)}, \mathbf{u}^{(1, l_1)}) = (i_1^{(l_1)}, \mathbf{u}_1^{(l_1)})$  et  $(j^{(1, l_1)}, \mathbf{v}^{(1, l_1)}) = (j_1^{(l_1)}, \mathbf{v}_1^{(l_1)})$ ,
- pour tout  $l_2 \in L_2$ ,  $(i^{(2, l_2)}, \mathbf{u}^{(2, l_2)}) = (i_2^{(l_2)} + |\mathbf{p}|, \mathbf{u}_2^{(l_2)})$  et  $(j^{(2, l_2)}, \mathbf{v}^{(2, l_2)}) = (j_2^{(l_2)} + |\mathbf{q}|, \mathbf{v}_2^{(l_2)})$ .

Ainsi,  $L$  et ces deux familles définissent  $f \otimes g$ . Les points **a**, **b** et **c** sont vérifiés par construction.

Le point **d** est vérifié : pour tous  $l \in L$  et  $z \in (\mathbf{u}^{(l)})^{-1} X_{\mathbf{p}[i^{(l)}]}$ ,

- si  $l = (1, l_1)$ ,  $l_1 \in L_1$ , alors  $i^{(l)} = i_1^{(l_1)} \leq |\mathbf{p}|$ ,

$$f \otimes g(i^{(l)}, \mathbf{u}^{(l)} z) = f(i^{(l)}, \mathbf{u}^{(l)} z) = f(i_1^{(l_1)}, \mathbf{u}_1^{(l_1)} z) = (j_1^{(l_1)}, \mathbf{v}_1^{(l_1)} z) = (j^{(l)}, \mathbf{v}^{(l)} z),$$

- sinon,  $l = (2, l_2)$ ,  $l_2 \in L_2$ , donc  $i^{(l)} = i_2^{(l_2)} + |\mathbf{p}| > |\mathbf{p}|$ ,

$$\begin{aligned}
f \otimes g(i^{(l)}, \mathbf{u}^{(l)} z) &= +^{|\mathbf{q}|} g(i^{(l)} - |\mathbf{p}|, \mathbf{u}^{(l)} z) = +^{|\mathbf{q}|} g(i_2^{(l_2)}, \mathbf{u}_2^{(l_2)} z) \\
&= +^{|\mathbf{q}|} (j_2^{(l_2)}, \mathbf{v}_2^{(l_2)} z) = (j_2^{(l_2)} + |\mathbf{q}|, \mathbf{v}_2^{(l_2)} z) = (j^{(l)}, \mathbf{v}^{(l)} z).
\end{aligned}$$

Donc  $f \otimes g \in \mathcal{C}_G[\mathbf{p} + \mathbf{p}', \mathbf{q} + \mathbf{q}']$ .  $\square$

## Annexe C Preuves des théorèmes

### C.1 Les applications cylindres généralisées forment une PROP

*Démonstration du théorème 1.* Posons  $\mathcal{C} = \mathcal{C}_G$ . On vérifie point par point les conditions de la définition 10.

- La composition séquentielle  $\circ : \mathcal{C}[\mathfrak{q}, \mathfrak{r}] \times \mathcal{C}[\mathfrak{p}, \mathfrak{q}] \rightarrow \mathcal{C}[\mathfrak{p}, \mathfrak{r}]$  définie sur les applications cylindres généralisées (stabilité donnée par la proposition 4) est associative (c'est la composition usuelle).
- La composition parallèle  $\otimes : \mathcal{C}[\mathfrak{p}, \mathfrak{q}] \times \mathcal{C}[\mathfrak{p}', \mathfrak{q}'] \rightarrow \mathcal{C}[\mathfrak{p} + \mathfrak{p}', \mathfrak{q} + \mathfrak{q}']$  définie sur les applications cylindres généralisées (stabilité donnée par la proposition 7) est associative (proposition 6).
- Pour tout  $q \in Q$ ,  $\text{id}_{\langle q \rangle} : (1, x) \in X[\langle q \rangle] \mapsto (1, x)$  est bien dans  $\mathcal{C}[\langle q \rangle, \langle q \rangle]$  car on peut la définir par les familles  $((1, \varepsilon))$  et  $((1, \varepsilon))$ .
- Le morphisme  $\text{id}_\circ : \emptyset \rightarrow \emptyset$  est dans  $\mathcal{C}[\circ, \circ]$ , défini par deux familles vides ( $L = \emptyset$ ).
- Pour tous  $q, q' \in Q$ ,  $\text{swap}_{\langle q \rangle, \langle q' \rangle} : (k, x) \in X[\langle q, q' \rangle] \mapsto (3-k, x) \in X[\langle q', q \rangle]$  est dans  $\mathcal{C}[\langle q, q' \rangle, \langle q', q \rangle]$  car elle est définie par  $L = \{1, 2\}$  et les familles  $((l, \varepsilon))_{l \in \{1, 2\}}$  et  $((3-l, \varepsilon))_{l \in \{1, 2\}}$ .
- Soient  $f \in \mathcal{C}[\mathfrak{p}, \mathfrak{q}]$ ,  $g \in \mathcal{C}[\mathfrak{q}, \mathfrak{r}]$ ,  $f' \in \mathcal{C}[\mathfrak{p}', \mathfrak{q}']$ ,  $g' \in \mathcal{C}[\mathfrak{q}', \mathfrak{r}']$  et  $(k, x) \in X[\mathfrak{p} + \mathfrak{p}']$ .
  - Si  $k \in \llbracket 1, |\mathfrak{p}| \rrbracket$ , on a d'une part :

$$\begin{aligned} (g \circ g') \circ (f \otimes f')(k, x) &= (g \otimes g')(k', x') && \text{où } (k', x') = f(k, x) \\ &= g(k', x') \in X[\mathfrak{r} + \mathfrak{r}'] && \text{car } k' \leq |\mathfrak{q}|, \end{aligned}$$

et d'autre part :

$$(g \circ f) \otimes (g' \circ f')(k, x) = g \circ f(k, x) = g(k', x') \in X[\mathfrak{r} + \mathfrak{r}'].$$

- Sinon ( $k \in \llbracket |\mathfrak{p}| + 1, |\mathfrak{p}| + |\mathfrak{p}'| \rrbracket$ ) on a :

$$\begin{aligned} (g \otimes g') \circ (f \otimes f')(k, x) &= (g \otimes g')(k' + |\mathfrak{q}|, x') && \text{où } (k', x') = f'(k - |\mathfrak{p}|, x) \\ &= (g \otimes g')(k' + |\mathfrak{q}|, x') \\ &= +^{|\mathfrak{r}|}(g'(k', x')) \in X[\mathfrak{r} + \mathfrak{r}'], \end{aligned}$$

et d'autre part :

$$(g \circ f) \otimes (g' \circ f')(k, x) = +^{|\mathfrak{r}|}((g' \circ f')(k - |\mathfrak{p}|, x)) = +^{|\mathfrak{r}|}(g'(k', x')) \in X[\mathfrak{r} + \mathfrak{r}'].$$

- Soient  $f \in \mathcal{C}[\mathfrak{p}, \mathfrak{q}]$  et  $(k, x) \in X[\mathfrak{p}]$ . Alors  $f \circ \text{id}_{\mathfrak{p}}(k, x) = f(k, x) = \text{id}_{\mathfrak{q}} \circ f(k, x) \in X[\mathfrak{q}]$ .
- De plus,  $f \otimes \text{id}_{\circ}(k, x) = f(k, x) = \text{id}_{\circ} \otimes f(k, x) \in X[\mathfrak{q}]$ .
- Soient  $q, q' \in Q$  et  $(k, x) \in X[\langle q, q' \rangle]$ .

Alors  $\text{swap}_{\langle q' \rangle, \langle q \rangle} \circ \text{swap}_{\langle q \rangle, \langle q' \rangle}(k, x) = \text{swap}_{\langle q' \rangle, \langle q \rangle}(3 - k, x) = (k, x) \in X[\langle q, q' \rangle]$ .

- Soient  $f \in \mathcal{C}[\mathfrak{p}, \mathfrak{q}]$ ,  $r \in Q$  et  $x \in X[\mathfrak{p} + \langle r \rangle]$ .

- Si  $k \in \llbracket 1, |\mathfrak{p}| \rrbracket$ , on a :

$$\begin{aligned} \text{swap}_{\mathfrak{q}, \langle r \rangle} \circ (f \otimes \text{id}_{\langle r \rangle})(k, x) &= \text{swap}_{\mathfrak{q}, \langle r \rangle}(k', x') && \text{où } (k', x') = f(k, x) \\ &= (k' + 1, x') \in X[\langle r \rangle + \mathfrak{q}] && \text{car } k' \in \llbracket 1, |\mathfrak{q}| \rrbracket, \end{aligned}$$

et :

$$\begin{aligned} (\text{id}_{\langle r \rangle} \otimes f) \circ \text{swap}_{\mathfrak{p}, \langle r \rangle}(k, x) &= (\text{id}_{\langle r \rangle} \otimes f)(k + 1, x) \\ &= +^1(f(k, x)) \\ &= (k' + 1, x') \in X[\langle r \rangle + \mathfrak{q}]. \end{aligned}$$

— Sinon ( $k = |\mathbf{p}| + 1$ ) :

$$\begin{aligned} \text{swap}_{\mathbf{q}, \langle r \rangle} \circ (f \otimes \text{id}_{\langle r \rangle})(|\mathbf{p}| + 1, x) &= \text{swap}_{\mathbf{q}, \langle r \rangle}(|\mathbf{q}| + 1, x) \\ &= (1, x) \in X[\langle r \rangle + \mathbf{q}], \end{aligned}$$

et :

$$\begin{aligned} (\text{id}_{\langle r \rangle} \otimes f) \circ \text{swap}_{\mathbf{p}, \langle r \rangle}(|\mathbf{p}| + 1, x) &= (\text{id}_{\langle r \rangle} \otimes f)(1, x) \\ &= (1, x) \in X[\langle r \rangle + \mathbf{q}]. \end{aligned}$$

□

*Remarque 12.* On a utilisé dans le dernier point la propriété suivante qui se démontre aisément par récurrence sur  $\mathbf{p}$ . ◁

**Proposition 9.** Pour tous  $\mathbf{p}$  liste de sommets et  $r \in Q$ , on a :

- $\text{swap}_{\mathbf{p}, \langle r \rangle}(|\mathbf{p}| + 1, x) = (1, x)$  (avec  $x \in X_r$ ),
- $\text{swap}_{\mathbf{p}, \langle r \rangle}(k, x) = (k + 1, x)$  (avec  $k \in \llbracket 1, |\mathbf{p}| \rrbracket$  et  $x \in X_{\mathbf{p}[k]}$ ).

## C.2 Fullness (surjectivité)

*Démonstration du théorème 2.* Soit  $f \in \mathcal{C}_G[\mathbf{p}, \mathbf{q}]$ . Soient  $L, (i^{(l)}, \mathbf{u}^{(l)})$  et  $(j^{(l)}, \mathbf{v}^{(l)})$  qui définissent  $f$ .

On raisonne par récurrence sur  $N = \sum_{l \in L} |\mathbf{u}^{(l)}| + |\mathbf{v}^{(l)}|$ .

Si  $N = 0$ , alors pour tout  $l \in L$ ,  $\mathbf{u}^{(l)} = \mathbf{v}^{(l)} = \varepsilon$ , donc  $f$  est une composition de permutations et d'identités seulement.

Sinon ( $N > 0$ ), supposons que toute application cylindre généralisée  $g \in \mathcal{C}_G[\mathbf{p}', \mathbf{q}']$  définie par des familles telles que  $\sum_{l \in L'} |\mathbf{u}^{(l)}| + |\mathbf{v}^{(l)}| < N$ , se décompose en  $\text{pop}_q$ ,  $\text{push}_q$ ,  $\text{swap}_{\mathbf{r}, \mathbf{r}'}$  et  $\text{id}_{\mathbf{r}}$ .

Premier cas : on suppose que  $\sum_{l \in L} |\mathbf{u}^{(l)}| > 0$ . Soit  $l_0 \in L$  tel que  $\mathbf{u}^{(l_0)} \neq \varepsilon$ .

On pose  $i_0 = i^{(l_0)}$  et  $q = \mathbf{p}[i_0]$ .

Puisque  $\{(i^{(l)}, [\mathbf{u}^{(l)}]) \mid l \in L\}$  est une partition de  $X[\mathbf{p}] = \bigoplus_{i=0}^{|\mathbf{p}|} X_{\mathbf{p}[i]}$ , alors  $\{[\mathbf{u}^{(k)}] \mid k \in K\}$  est une partition de  $X_q$ , où  $K = \{l \in L : i^{(l)} = i_0\}$ .

Donc, pour tout  $k \in K$ ,  $\mathbf{u}^{(k)} \neq \varepsilon$  et  $s(\mathbf{u}_1^{(k)}) = \mathbf{p}[i^{(k)}] = q$ .

Ainsi, on pose  $g \in \mathcal{C}_G[\mathbf{p}', \mathbf{q}]$ , avec  $\mathbf{p}' = \mathbf{p}[1 : i_0 - 1] + \varphi_q + \mathbf{p}[i_0 + 1 : |\mathbf{p}|]$ , une application cylindre généralisée définie par  $L' = L$ ,  $(i'^{(l)}, \mathbf{u}'^{(l)})$  et  $(j'^{(l)}, \mathbf{v}'^{(l)})$  telles que pour tout  $l \in L'$  :

$$(i'^{(l)}, \mathbf{u}'^{(l)}) = \begin{cases} (i^{(l)}, \mathbf{u}^{(l)}) & \text{si } i^{(l)} < i_0 \\ (i^{(l)} + \alpha_q^{-1}(\mathbf{u}_1^{(l)}) - 1, \mathbf{u}_{\llbracket 2, |\mathbf{u}^{(l)} \rrbracket}^{(l)}) & \text{si } i^{(l)} = i_0 \\ (i^{(l)} + \text{deg}(q) - 1, \mathbf{u}^{(l)}) & \text{si } i^{(l)} > i_0 \end{cases}, \quad \text{et } (j'^{(l)}, \mathbf{v}'^{(l)}) = (j^{(l)}, \mathbf{v}^{(l)}).$$

Montrons que  $f = g \circ \underbrace{(\text{id}_{\mathbf{p}[1:i_0-1]} \otimes \text{pop}_q \otimes \text{id}_{\mathbf{p}[i_0+1:|\mathbf{p}|]})}_{=h}$ . Soit  $(i^{(l)}, \mathbf{u}^{(l)} \cdot z) \in X[\mathbf{p}']$ .

On a  $f(i^{(l)}, \mathbf{u}^{(l)} \cdot z) = (j^{(l)}, \mathbf{v}^{(l)} \cdot z)$  et :

- si  $i^{(l)} < i_0$ ,  $g \circ h(i^{(l)}, \mathbf{u}^{(l)} \cdot z) = g(i^{(l)}, \mathbf{u}^{(l)} \cdot z) = (j^{(l)}, \mathbf{v}^{(l)} \cdot z)$ ,
- si  $i^{(l)} = i_0$ ,  $g \circ h(i^{(l)}, \mathbf{u}^{(l)} \cdot z) = g(i^{(l)} + \alpha_q^{-1}(\mathbf{u}_1^{(l)}) - 1, \mathbf{u}_{\llbracket 2, |\mathbf{u}^{(l)} \rrbracket}^{(l)} \cdot z) = (j^{(l)}, \mathbf{v}^{(l)} \cdot z)$ ,

— si  $i^{(l)} > i_0$ ,  $g \circ h(i^{(l)}, \mathbf{u}^{(l)} \cdot z) = g(i^{(l)} + \deg(q) - 1, \mathbf{u}^{(l)} \cdot z) = (j^{(l)}, \mathbf{v}^{(l)} \cdot z)$ .

Puisque  $\sum_{l \in L'} \mathbf{u}'^{(l)} < \sum_{l \in L} \mathbf{u}^{(l)}$ ,  $\sum_{l \in L'} |\mathbf{u}'^{(l)}| + |\mathbf{v}'^{(l)}| < N$  donc par hypothèse de récurrence,  $g$  se décompose en  $\text{pop}_q$ ,  $\text{push}_q$ ,  $\text{swap}_{r,r'}$  et  $\text{id}_r$ .

Ainsi, c'est aussi le cas pour  $f = g \circ (\text{id}_{\mathfrak{p}[1:i_0-1]} \otimes \text{pop}_q \otimes \text{id}_{\mathfrak{p}[i_0+1:|p|]})$ .

Second cas : si  $\sum_{l \in L} |\mathbf{u}^{(l)}| = 0$ , alors  $\sum_{l \in L} |\mathbf{v}^{(l)}| > 0$  ( $N > 0$ ). Le même raisonnement convient en échangeant le rôle des familles  $(i^{(l)}, \mathbf{u}^{(l)})$  et  $(j^{(l)}, \mathbf{v}^{(l)})$ .

Dans ce cas,  $f$  s'écrira  $(\text{id}_{\mathfrak{q}[1:i_0-1]} \otimes \text{push}_q \otimes \text{id}_{\mathfrak{q}[i_0+1:|q|]}) \circ g$ . □

### C.3 Faithfulness (injectivité)

*Démonstration du théorème 3.* Soit  $\boxed{D}$ ,  $\boxed{E}$  tels que  $F(\boxed{D}) = F(\boxed{E})$ .

Soient  $\boxed{D_1}$ ,  $\boxed{D_2}$  tels que  $\boxed{D} = \boxed{D_1} \boxed{D_2}$  donnés par le lemme 1.

Soit  $\boxed{G} = \boxed{E} \boxed{D_2^\dagger}$ , et  $\boxed{G_1}$ ,  $\boxed{G_2}$  tels que  $\boxed{G} = \boxed{G_1} \boxed{G_2}$ .

On a :

$$\begin{aligned}
F(\boxed{D_1} \boxed{G_2^\dagger}) &= F(\boxed{G_2^\dagger}) \circ F(\boxed{D_1}) \\
&= F(\boxed{G_2^\dagger}) \circ F(\boxed{D} \boxed{D_2^\dagger}) \\
&= F(\boxed{G_2^\dagger}) \circ F(\boxed{D_2^\dagger}) \circ F(\boxed{D}) \\
&= F(\boxed{G_2^\dagger}) \circ F(\boxed{D_2^\dagger}) \circ F(\boxed{E}) && \text{par hypothèse,} \\
&= F(\boxed{G_2^\dagger}) \circ F(\boxed{E} \boxed{D_2^\dagger}) \\
&= F(\boxed{G_2^\dagger}) \circ F(\boxed{G}) && \text{par définition de } \boxed{G}, \\
&= F(\boxed{G} \boxed{G_2^\dagger}) = F(\boxed{G_1}).
\end{aligned}$$

Alors, par le lemme 2,  $\boxed{D_1} \boxed{G_2^\dagger} = \boxed{G_1}$ .

Ainsi,

$$\begin{aligned}
\boxed{D} &= \boxed{D_1} \boxed{D_2} = \boxed{D_1} \boxed{G_2^\dagger} \boxed{G_2} \boxed{D_2} \\
&= \boxed{G_1} \boxed{G_2} \boxed{D_2} = \boxed{G} \boxed{D_2} && \text{par le résultat précédent,} \\
&= \boxed{E} \boxed{D_2} \boxed{D_2^\dagger} && \text{par définition de } \boxed{G}, \\
&= \boxed{E}.
\end{aligned}$$

□