Construction d'opérades combinatoires à partir de monoïdes

Samuele Giraudo

Université de Marne-la-Vallée

LaCIM
10 novembre 2011

Plan

Plan

Opérades ensemblistes non symétriques

Définition

Une opérade ensembliste non symétrique est un ensemble

$$\mathcal{P} := \biguplus_{n \geq 1} \mathcal{P}(n),$$

muni d'applications de substitution partielle

$$\circ_i : \mathcal{P}(n) \times \mathcal{P}(m) \to \mathcal{P}(n + m - 1), \qquad n, m \geq 1, \quad i \in [n],$$

et d'une unité $\mathbf{1} \in \mathcal{P}(1)$.

Opérades ensemblistes non symétriques

Définition

Une opérade ensembliste non symétrique est un ensemble

$$\mathcal{P} := \biguplus_{n \geq 1} \mathcal{P}(n),$$

muni d'applications de substitution partielle

$$\circ_i : \mathcal{P}(n) \times \mathcal{P}(m) \to \mathcal{P}(n+m-1), \qquad n, m \geq 1, \quad i \in [n],$$

et d'une unité $\mathbf{1} \in \mathcal{P}(1)$.

Les relations suivantes doivent être vérifiées pour tous $x \in \mathcal{P}(n)$, $y \in \mathcal{P}(m)$ *et* $z \in \mathcal{P}(k)$ *:*

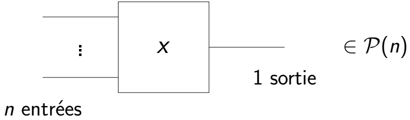
$$\mathbf{1} \circ_1 x = x = x \circ_i \mathbf{1}, \quad i \in [n], \qquad \text{(unité)}$$

$$(x \circ_i y) \circ_{i+j-1} z = x \circ_i (y \circ_j z), \quad i \in [n], j \in [m], \qquad \text{(assoc. série)}$$

$$(x \circ_i y) \circ_{j+m-1} z = (x \circ_j z) \circ_i y, \quad 1 \leq i < j \leq n, \qquad \text{(assoc. parallèle)}.$$

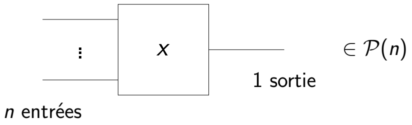
Éléments d'une opérade et substitutions partielles

- Élément de $\mathcal{P}(n) \rightsquigarrow$ opérateur d'arité n :

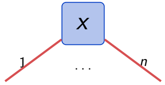


Éléments d'une opérade et substitutions partielles

- Élément de $\mathcal{P}(n) \rightsquigarrow$ opérateur d'arité n :

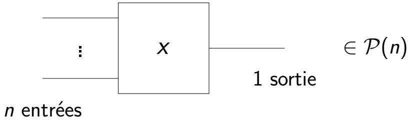


- Opérateur d'arité $n \rightsquigarrow$ arbre plan enraciné à n feuilles :

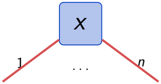


Éléments d'une opérade et substitutions partielles

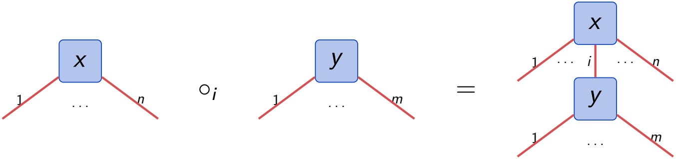
- Élément de $\mathcal{P}(n) \rightsquigarrow$ opérateur d'arité n :



- Opérateur d'arité $n \rightsquigarrow$ arbre plan enraciné à n feuilles :



- Substitution partielle $\rightsquigarrow$ greffe d'arbres :

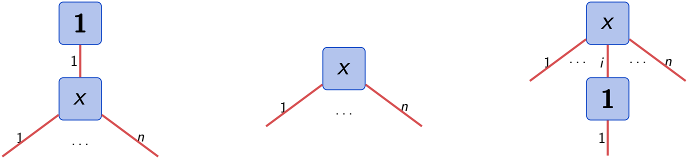


L'axiome d'unité

$$\mathbf{1} \circ_1 x = x = x \circ_i \mathbf{1}, \quad i \in [n]$$

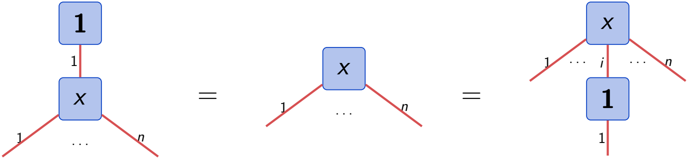
L'axiome d'unité

$$\mathbf{1} \circ_1 x \quad = \quad x \quad = \quad x \circ_i \mathbf{1}, \quad i \in [n]$$



L'axiome d'unité

$$\mathbf{1} \circ_1 x \quad = \quad x \quad = \quad x \circ_i \mathbf{1}, \quad i \in [n]$$

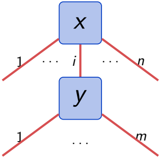


L'axiome d'associativité série

$$(x \circ_i y) \circ_{i+j-1} z \quad = \quad x \circ_i (y \circ_j z), \quad i \in [n], j \in [m],$$

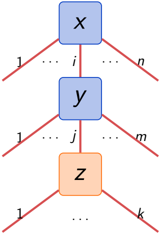
L'axiome d'associativité série

$$(x \circ_i y) \circ_{i+j-1} z \quad = \quad x \circ_i (y \circ_j z), \quad i \in [n], j \in [m],$$



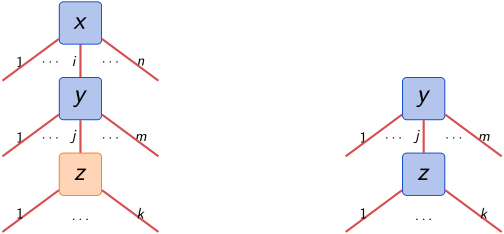
L'axiome d'associativité série

$$(x \circ_i y) \circ_{i+j-1} z \quad = \quad x \circ_i (y \circ_j z), \quad i \in [n], j \in [m],$$



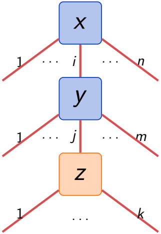
L'axiome d'associativité série

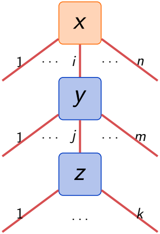
$$(x \circ_i y) \circ_{i+j-1} z \quad = \quad x \circ_i (y \circ_j z), \quad i \in [n], j \in [m],$$



L'axiome d'associativité série

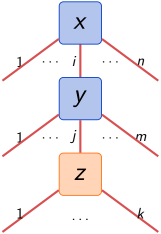
$$(x \circ_i y) \circ_{i+j-1} z \quad = \quad x \circ_i (y \circ_j z), \quad i \in [n], j \in [m],$$



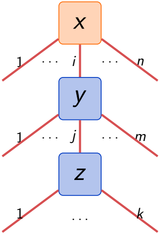


L'axiome d'associativité série

$$(x \circ_i y) \circ_{i+j-1} z \quad = \quad x \circ_i (y \circ_j z), \quad i \in [n], j \in [m],$$



$=$

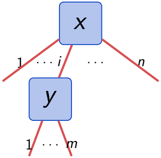


L'axiome d'associativité parallèle

$$(x \circ_i y) \circ_{j+m-1} z \quad = \quad (x \circ_j z) \circ_i y, \quad 1 \leq i < j \leq n,$$

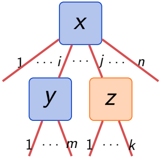
L'axiome d'associativité parallèle

$$(x \circ_i y) \circ_{j+m-1} z \quad = \quad (x \circ_j z) \circ_i y, \quad 1 \leq i < j \leq n,$$



L'axiome d'associativité parallèle

$$(x \circ_i y) \circ_{j+m-1} z \quad = \quad (x \circ_j z) \circ_i y, \quad 1 \leq i < j \leq n,$$

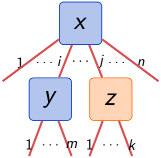


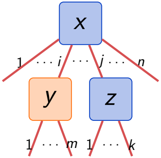
L'axiome d'associativité parallèle

$$(x \circ_i y) \circ_{j+m-1} z \quad = \quad (x \circ_j z) \circ_i y, \quad 1 \leq i < j \leq n,$$

L'axiome d'associativité parallèle

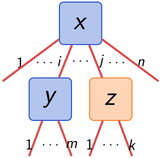
$$(x \circ_i y) \circ_{j+m-1} z \quad = \quad (x \circ_j z) \circ_i y, \quad 1 \le i < j \le n,$$



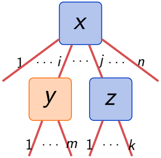


L'axiome d'associativité parallèle

$$(x \circ_i y) \circ_{j+m-1} z \quad = \quad (x \circ_j z) \circ_i y, \quad 1 \leq i < j \leq n,$$



$=$



Plan

Opérades et combinatoire algébrique

- Opérades sur des objets combinatoires $\rightsquigarrow$ nouvelles opérations sur les objets.

Opérades et combinatoire algébrique

- Opérades sur des objets combinatoires $\rightsquigarrow$ nouvelles opérations sur les objets.

- Quelques opérades combinatoires :

Opérade	Objets
Dupliciale	Arbres binaires [**Brouder**, **Frabetti**, 03]
Dendriforme	Arbres binaires [**Loday**, 2001]
NAP	Arbres enracinés étiquetés [**Livernet**, 06]
Pré-Lie	Arbres enracinés étiquetés [**Chapoton**, **Livernet**, 01]

Opérades et combinatoire algébrique

- Opérades sur des objets combinatoires $\rightsquigarrow$ nouvelles opérations sur les objets.
- Quelques opérades combinatoires :

Opérade	Objets
Dupliciale	Arbres binaires [**Brouder**, **Frabetti**, 03]
Dendriforme	Arbres binaires [**Loday**, 2001]
NAP	Arbres enracinés étiquetés [**Livernet**, 06]
Pré-Lie	Arbres enracinés étiquetés [**Chapoton**, **Livernet**, 01]

- Manipulation de divers types d'algèbres (algèbres associatives, algèbres de Lie, algèbres dendriformes, etc.).

Opérades et combinatoire algébrique

- Opérades sur des objets combinatoires $\rightsquigarrow$ nouvelles opérations sur les objets.
- Quelques opérades combinatoires :

Opérade	Objets
Dupliciale	Arbres binaires [**Brouder, Frabetti**, 03]
Dendriforme	Arbres binaires [**Loday**, 2001]
NAP	Arbres enracinés étiquetés [**Livernet**, 06]
Pré-Lie	Arbres enracinés étiquetés [**Chapoton, Livernet**, 01]

- Manipulation de divers types d'algèbres (algèbres associatives, algèbres de Lie, algèbres dendriformes, etc.).
- Plusieurs constructions [**Chapoton, Livernet**, 07]

  opérades $\rightsquigarrow$ algèbres de Hopf combinatoires.

Plan

La construction T (1)

Soit $(M, \bullet, 1)$ un monoïde.

La construction T (1)

Soit $(M, \bullet, 1)$ un monoïde.

- Soit $\mathsf{T}M$ l'ensemble $\mathsf{T}M := \biguplus_{n \geq 1} \mathsf{T}M(n)$ où

$$\mathsf{T}M(n) := \{(x_1, \ldots, x_n) : x_i \in M \text{ pour tout } i \in [n]\}.$$

La construction T (1)

Soit $(M, \bullet, 1)$ un monoïde.

- Soit $\mathsf{T}M$ l'ensemble $\mathsf{T}M := \biguplus_{n \geq 1} \mathsf{T}M(n)$ où

$$\mathsf{T}M(n) := \{(x_1, \dots, x_n) : x_i \in M \text{ pour tout } i \in [n]\}.$$

- Soient les applications de substitution partielle

$$\circ_i : \mathsf{T}M(n) \times \mathsf{T}M(m) \to \mathsf{T}M(n + m - 1),$$

  définies pour tout $x \in \mathsf{T}M(n)$ et $y \in \mathsf{T}M(m)$ par

$$x \circ_i y := (x_1, \ \dots, \ x_{i-1}, \ x_i \bullet y_1, \ \dots, \ x_i \bullet y_m, \ x_{i+1}, \ \dots, \ x_n).$$

La construction T (1)

Soit $(M, \bullet, 1)$ un monoïde.

- Soit $\mathsf{T}M$ l'ensemble $\mathsf{T}M := \biguplus_{n \geq 1} \mathsf{T}M(n)$ où
$$\mathsf{T}M(n) := \{(x_1, \dots, x_n) : x_i \in M \text{ pour tout } i \in [n]\}\,.$$
- Soient les applications de substitution partielle
$$\circ_i : \mathsf{T}M(n) \times \mathsf{T}M(m) \to \mathsf{T}M(n+m-1),$$
définies pour tout $x \in \mathsf{T}M(n)$ et $y \in \mathsf{T}M(m)$ par
$$x \circ_i y := (x_1,\ \dots,\ x_{i-1},\ x_i \bullet y_1,\ \dots,\ x_i \bullet y_m,\ x_{i+1},\ \dots,\ x_n).$$
- Soit $\mathbf{1} := (1) \in \mathsf{T}M(1)$.

La construction T (1)

Soit $(M, \bullet, 1)$ un monoïde.

- Soit $\mathsf{T}M$ l'ensemble $\mathsf{T}M := \biguplus_{n \geq 1} \mathsf{T}M(n)$ où
$$\mathsf{T}M(n) := \{(x_1, \dots, x_n) : x_i \in M \text{ pour tout } i \in [n]\}.$$
- Soient les applications de substitution partielle
$$\circ_i : \mathsf{T}M(n) \times \mathsf{T}M(m) \to \mathsf{T}M(n + m - 1),$$
définies pour tout $x \in \mathsf{T}M(n)$ et $y \in \mathsf{T}M(m)$ par
$$x \circ_i y := (x_1, \ \dots, \ x_{i-1}, \ x_i \bullet y_1, \ \dots, \ x_i \bullet y_m, \ x_{i+1}, \ \dots, \ x_n).$$
- Soit $\mathbf{1} := (1) \in \mathsf{T}M(1)$.
- Soit l'action du groupe symétrique
$$\cdot : \mathsf{T}M(n) \times \mathfrak{S}_n \to \mathsf{T}M(n),$$
définie pour tous $x \in \mathsf{T}M(n)$ et $\sigma \in \mathfrak{S}_n$ par
$$x \cdot \sigma := (x_{\sigma_1}, \dots, x_{\sigma_n}).$$

La construction T (2)

Soient M et N deux monoïdes et $\theta : M \to N$ un morphisme de monoïde.

La construction T (2)

Soient M et N deux monoïdes et $\theta : M \to N$ un morphisme de monoïde.

Soit $\mathsf{T}\theta$ l'application

$$\mathsf{T}\theta : \mathsf{T}M \to \mathsf{T}N,$$

définie pour tout $(x_1, \ldots, x_n) \in \mathsf{T}M(n)$ par

$$\mathsf{T}\theta\,(x_1, \ldots, x_n) := (\theta(x_1), \ldots, \theta(x_n))\,.$$

Quelques exemples

Exemple

- Lorsque M est le monoïde additif $(\mathbb{N}, +)$, on a dans $\mathsf{T}M$:

$$2\mathbf{1}23 \circ_2 30313 = 24142423,$$

et

$$11210 \cdot (23514) = 12011.$$

Quelques exemples

Exemple

- Lorsque M est le monoïde additif $(\mathbb{N}, +)$, on a dans $\mathsf{T}M$:

$$2\mathbf{1}23 \circ_2 30313 = 24142423,$$

et

$$11210 \cdot (23514) = 12011.$$

- Lorsque N est le monoïde libre $\{\mathtt{a}, \mathtt{b}\}^*$, on a dans $\mathsf{T}N$:

$$\begin{array}{ccccc} \mathtt{b} & \mathtt{a} & \mathtt{a} & \epsilon & \mathtt{b} \\ & \mathtt{b} & & & \mathtt{b} \\ & \mathtt{a} & & & \end{array} \circ_3 \begin{array}{cccc} \epsilon & \mathtt{a} & \epsilon & \mathtt{b} \\ & & & \mathtt{b} \end{array} = \begin{array}{cccccccc} \mathtt{b} & \mathtt{a} & \mathtt{a} & \mathtt{a} & \mathtt{a} & \mathtt{a} & \epsilon & \mathtt{b} \\ & \mathtt{b} & & \mathtt{a} & & \mathtt{b} & & \mathtt{b} \\ & \mathtt{a} & & & & \mathtt{b} & & \end{array}.$$

Quelques exemples

Exemple

- Lorsque M est le monoïde additif $(\mathbb{N}, +)$, on a dans $\mathsf{T}M$:

$$2\mathbf{1}23 \circ_2 30313 = 24142423,$$

et

$$11210 \cdot (23514) = 12011.$$

- Lorsque N est le monoïde libre $\{\mathtt{a}, \mathtt{b}\}^*$, on a dans $\mathsf{T}N$:

$$\begin{array}{ccccc} \mathtt{b} & \mathtt{a} & \mathtt{a} & \epsilon & \mathtt{b} \\ & \mathtt{b} & & & \mathtt{b} \\ & \mathtt{a} & & & \end{array} \circ_3 \begin{array}{cccc} \epsilon & \mathtt{a} & \epsilon & \mathtt{b} \\ & & & \mathtt{b} \end{array} = \begin{array}{cccccccc} \mathtt{b} & \mathtt{a} & \mathtt{a} & \mathtt{a} & \mathtt{a} & \mathtt{a} & \epsilon & \mathtt{b} \\ & \mathtt{b} & & \mathtt{a} & & \mathtt{b} & & \mathtt{b} \\ & \mathtt{a} & & & & \mathtt{b} & & \end{array}.$$

- Soit $\theta : N \to M$ défini par $\theta(u) := |u|$. On a

$$\mathsf{T}\theta \left(\begin{array}{ccccc} \mathtt{b} & \mathtt{a} & \mathtt{a} & \epsilon & \mathtt{b} \\ & \mathtt{b} & & & \mathtt{b} \\ & \mathtt{a} & & & \end{array} \right) = 13102.$$

Plan

Propriétés de la construction T (1)

Proposition

Si M est un monoïde alors $\mathsf{T}M$ est une opérade ensembliste.

Propriétés de la construction T (1)

Proposition

Si M est un monoïde alors $\mathsf{T}M$ est une opérade ensembliste.

Proposition

Si M et N sont deux monoïdes et $\theta : M \to N$ est un morphisme de monoïde alors l'application $\mathsf{T}\theta : \mathsf{T}M \to \mathsf{T}N$ est un morphisme d'opérade.

Propriétés de la construction T (2)

Proposition

Si M et N sont deux monoïdes et $\theta : M \to N$ est un morphisme de monoïde injectif (resp. surjectif) alors $\mathsf{T}\theta$ est injectif (resp. surjectif).

Propriétés de la construction T (2)

Proposition

Si M et N sont deux monoïdes et $\theta : M \to N$ est un morphisme de monoïde injectif (resp. surjectif) alors $\mathsf{T}\theta$ est injectif (resp. surjectif).

Théorème

La construction T est un foncteur de la catégorie des monoïdes avec morphismes de monoïde vers la catégorie des opérades avec morphismes d'opérade.

Plan

Quelques opérades obtenues (1)

Voici quelques sous-opérades ns ou quotients de $\mathsf{T}\mathbb{N}$ que l'on obtient :

Opérade	Premières dimensions	Objets combinatoires
End	$1, 4, 27, 256, 3125$	Endofonctions
FP	$1, 3, 16, 125, 1296$	Fonctions de parking
MT	$1, 3, 13, 75, 541$	Mots tassés
Per	$1, 2, 6, 24, 120$	Permutations
APE	$1, 1, 2, 5, 14, 42$	Arbres plans enracinés
$\mathit{FCat}^{(k)}$	Nombres de Fuss-Catalan	k-chemins de Dyck
Schr	$1, 3, 11, 45, 197$	Arbres de Schröder
Motz	$1, 1, 2, 4, 9, 21, 51$	Chemins de Motzkin
Comp	$1, 2, 4, 8, 16, 32$	Compo. d'entiers
AnD	$1, 2, 5, 13, 35, 96$	Animaux dirigés
SComp	$1, 3, 27, 81, 243$	Compo. d'entiers segmentées

Quelques opérades obtenues (2)

Elles s'agencent selon le diagramme suivant ($\rightarrowtail$ signifie injection et $\twoheadrightarrow$ surjection) :

$$
\begin{tikzcd}
 & & & \mathsf{T}\mathbb{N} \arrow[dlll, two heads] \arrow[drr, two heads] & & \\
\mathsf{T}\mathbb{N}_2 & & \mathit{End} \arrow[ur, tail] & & & \mathsf{T}\mathbb{N}_3 \\
 & \mathit{FP} \arrow[ur, tail] & & & & \\
 & \mathit{MT} \arrow[u, tail] \arrow[d, two heads] & & \mathit{FCat}^{(3)} \arrow[uuu, tail, dashed] & & \\
 & \mathit{Per} & \mathit{Schr} \arrow[ul, tail] & \mathit{FCat}^{(2)} \arrow[u, tail] \arrow[drr, two heads] & & \\
 & & & \mathit{FCat}^{(1)} \arrow[ul, tail] \arrow[u, tail] \arrow[dlll, two heads] \arrow[drr, two heads] & & \mathit{SComp} \arrow[uuuu, tail] \\
\mathit{Comp} \arrow[uuuuu, tail] \arrow[drrr, two heads] & & \mathit{Motz} \arrow[ur, tail] & & \mathit{APE} \arrow[ul, tail] \arrow[dl, two heads] & \mathit{AnD} \arrow[u, tail] \arrow[dll, two heads] \\
 & & & \mathit{FCat}^{(0)} \arrow[ul, tail] & &
\end{tikzcd}
$$

Plan

Arbres plans enracinés (1)

Expérimentation avec Sage :

```
sage: M = MonoideAdditif()
```

Arbres plans enracinés (1)

Expérimentation avec Sage :

```
sage: M = MonoideAdditif()
sage: G = tuple([Mot(M, [0, 1])])
```

Arbres plans enracinés (1)

Expérimentation avec Sage :

```
sage: M = MonoideAdditif()
sage: G = tuple([Mot(M, [0, 1])])
sage: P = OperadeMonoide(M, False, G)
```

Arbres plans enracinés (1)

Expérimentation avec Sage :

```
sage: M = MonoideAdditif()
sage: G = tuple([Mot(M, [0, 1])])
sage: P = OperadeMonoide(M, False, G)
sage: print [P.dimension(n) for n in xrange(1, 10)]
```

Arbres plans enracinés (1)

Expérimentation avec Sage :

```
sage: M = MonoideAdditif()
sage: G = tuple([Mot(M, [0, 1])])
sage: P = OperadeMonoide(M, False, G)
sage: print [P.dimension(n) for n in xrange(1, 10)]
[1, 1, 2, 5, 14, 42, 132, 429, 1430]
```

Arbres plans enracinés (1)

Expérimentation avec Sage :

```
sage: M = MonoideAdditif()
sage: G = tuple([Mot(M, [0, 1])])
sage: P = OperadeMonoide(M, False, G)
sage: print [P.dimension(n) for n in xrange(1, 10)]
[1, 1, 2, 5, 14, 42, 132, 429, 1430]

sage: E = list(P.elements(5)); E.sort(); print E
```

Arbres plans enracinés (1)

Expérimentation avec Sage :

```
sage: M = MonoideAdditif()
sage: G = tuple([Mot(M, [0, 1])])
sage: P = OperadeMonoide(M, False, G)
sage: print [P.dimension(n) for n in xrange(1, 10)]
[1, 1, 2, 5, 14, 42, 132, 429, 1430]

sage: E = list(P.elements(5)); E.sort(); print E
[01111, 01112, 01121, 01122, 01123, 01211, 01212, 01221,
01222, 01223, 01231, 01232, 01233, 01234]
```

Arbres plans enracinés (2)

Soit APE la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 01.

Arbres plans enracinés (2)

Soit APE la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 01.

Proposition

Les éléments de APE sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $1 \leq x_{i+1} \leq x_i + 1$ pour tout $i \in [|x| - 1]$.

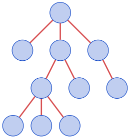
Arbres plans enracinés (2)

Soit APE la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 01.

Proposition

Les éléments de APE sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $1 \leq x_{i+1} \leq x_i + 1$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les arbres plans enracinés à n nœuds :

 $\longleftrightarrow$

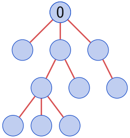
Arbres plans enracinés (2)

Soit APE la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 01.

Proposition

Les éléments de APE sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $1 \leq x_{i+1} \leq x_i + 1$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les arbres plans enracinés à n nœuds :



$\longleftrightarrow \quad 0$

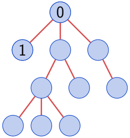
Arbres plans enracinés (2)

Soit *APE* la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 01.

Proposition

Les éléments de APE sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $1 \leq x_{i+1} \leq x_i + 1$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les arbres plans enracinés à n nœuds :



$\longleftrightarrow$ 01

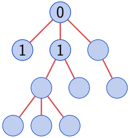
Arbres plans enracinés (2)

Soit *APE* la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 01.

Proposition

Les éléments de APE sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $1 \leq x_{i+1} \leq x_i + 1$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les arbres plans enracinés à n nœuds :



$\longleftrightarrow$ 011

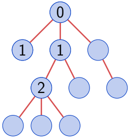
Arbres plans enracinés (2)

Soit *APE* la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 01.

Proposition

Les éléments de APE sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $1 \leq x_{i+1} \leq x_i + 1$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les arbres plans enracinés à n nœuds :



$\longleftrightarrow$ 0112

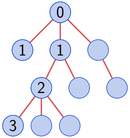
Arbres plans enracinés (2)

Soit APE la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 01.

Proposition

Les éléments de APE sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $1 \leq x_{i+1} \leq x_i + 1$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les arbres plans enracinés à n nœuds :



$\longleftrightarrow$ 01123

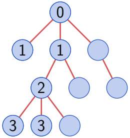
Arbres plans enracinés (2)

Soit *APE* la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 01.

Proposition

Les éléments de APE sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $1 \leq x_{i+1} \leq x_i + 1$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les arbres plans enracinés à n nœuds :



$\longleftrightarrow$ 011233

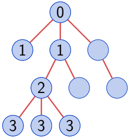
Arbres plans enracinés (2)

Soit *APE* la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 01.

Proposition

Les éléments de APE sont exactement les mots x sur l'alphabet $\mathbb{N}$ *qui vérifient* $x_1 = 0$ *et* $1 \leq x_{i+1} \leq x_i + 1$ *pour tout* $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les arbres plans enracinés à n nœuds :



$\longleftrightarrow$ 0112333

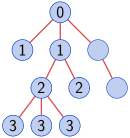
Arbres plans enracinés (2)

Soit *APE* la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 01.

Proposition

Les éléments de APE sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $1 \leq x_{i+1} \leq x_i + 1$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les arbres plans enracinés à n nœuds :



$\longleftrightarrow$ 01123332

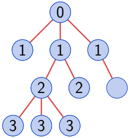
Arbres plans enracinés (2)

Soit APE la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 01.

Proposition

Les éléments de APE sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $1 \leq x_{i+1} \leq x_i + 1$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les arbres plans enracinés à n nœuds :



$\longleftrightarrow$ 011233321

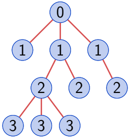
Arbres plans enracinés (2)

Soit *APE* la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 01.

Proposition

Les éléments de APE sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $1 \leq x_{i+1} \leq x_i + 1$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les arbres plans enracinés à n nœuds :



$\longleftrightarrow$ 0112333212

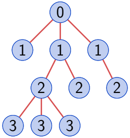
Arbres plans enracinés (2)

Soit APE la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 01.

Proposition

Les éléments de APE sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $1 \leq x_{i+1} \leq x_i + 1$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les arbres plans enracinés à n nœuds :



$\longleftrightarrow$ 0112333212

$\rightsquigarrow$ APE est une opérade ns sur les arbres plans enracinés et $\dim APE(n) = \frac{1}{n}\binom{2n-2}{n-1}$.

Arbres plans enracinés (3)

Interprétation de la substitution partielle en termes d'arbres plans enracinés :

Proposition

Soient S et T deux arbres plans enracinés et s le i^e nœud visité de S selon son parcours en profondeur. La substitution partielle $S \circ_i T$ dans APE *revient à greffer dans S les sous-arbres de la racine de T en tant que fils les plus à gauche de s.*

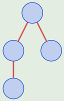
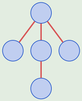
Arbres plans enracinés (3)

Interprétation de la substitution partielle en termes d'arbres plans enracinés :

Proposition

Soient S et T deux arbres plans enracinés et s le i^e nœud visité de S selon son parcours en profondeur. La substitution partielle $S \circ_i T$ dans APE revient à greffer dans S les sous-arbres de la racine de T en tant que fils les plus à gauche de s.

Exemple

 $\circ_2$  $=$

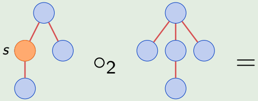
Arbres plans enracinés (3)

Interprétation de la substitution partielle en termes d'arbres plans enracinés :

Proposition

Soient S et T deux arbres plans enracinés et s le i^e nœud visité de S selon son parcours en profondeur. La substitution partielle $S \circ_i T$ dans APE revient à greffer dans S les sous-arbres de la racine de T en tant que fils les plus à gauche de s.

Exemple



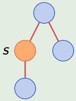
Arbres plans enracinés (3)

Interprétation de la substitution partielle en termes d'arbres plans enracinés :

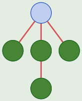
Proposition

Soient S et T deux arbres plans enracinés et s le i^e nœud visité de S selon son parcours en profondeur. La substitution partielle $S \circ_i T$ dans APE revient à greffer dans S les sous-arbres de la racine de T en tant que fils les plus à gauche de s.

Exemple



$\circ_2$



$=$

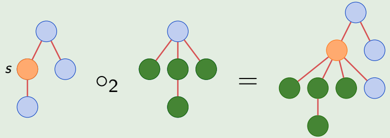
Arbres plans enracinés (3)

Interprétation de la substitution partielle en termes d'arbres plans enracinés :

Proposition

Soient S et T deux arbres plans enracinés et s le i^e nœud visité de S selon son parcours en profondeur. La substitution partielle $S \circ_i T$ dans APE revient à greffer dans S les sous-arbres de la racine de T en tant que fils les plus à gauche de s.

Exemple



Arbres plans enracinés (4)

Structure de APE :

Proposition

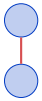
L'opérade ns APE est isomorphe à l'opérade ns libre sur un générateur d'arité 2.

Arbres plans enracinés (4)

Structure de APE :

Proposition

L'opérade ns APE est isomorphe à l'opérade ns libre sur un générateur d'arité 2.

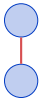
$\rightsquigarrow$ tout arbre plan enraciné peut être obtenu par l'intermédiaire de greffes du générateur . De plus, cette écriture est unique aux règles d'associativité des opérades près.

Arbres plans enracinés (4)

Structure de APE :

Proposition

L'opérade ns APE est isomorphe à l'opérade ns libre sur un générateur d'arité 2.

$\rightsquigarrow$ tout arbre plan enraciné peut être obtenu par l'intermédiaire de greffes du générateur . De plus, cette écriture est unique aux règles d'associativité des opérades près.

L'opérade APE est une version planaire de l'opérade NAP.

k-chemins de Dyck (1)

Expérimentation avec Sage :

```
sage: M = MonoideAdditif()
sage: G = tuple([Mot(M, [0, 0]), Mot(M, [0, 1])])
```

k-chemins de Dyck (1)

Expérimentation avec Sage :

```
sage: M = MonoideAdditif()
sage: G = tuple([Mot(M, [0, 0]), Mot(M, [0, 1])])
sage: P = OperadeMonoide(M, False, G)
sage: print [P.dimension(n) for n in xrange(1, 10)]
[1, 2, 5, 14, 42, 132, 429, 1430, 4862]
```

k-chemins de Dyck (1)

Expérimentation avec Sage :

```
sage: M = MonoideAdditif()
sage: G = tuple([Mot(M, [0, 0]), Mot(M, [0, 1])])
sage: P = OperadeMonoide(M, False, G)
sage: print [P.dimension(n) for n in xrange(1, 10)]
[1, 2, 5, 14, 42, 132, 429, 1430, 4862]

sage: E = list(P.elements(4)); E.sort(); print E
[0000, 0001, 0010, 0011, 0012, 0100, 0101, 0110, 0111,
0112, 0120, 0121, 0122, 0123]
```

k-chemins de Dyck (2)

Expérimentation avec Sage :

```
sage: M = MonoideAdditif()
sage: G = tuple([Mot(M, [0, 0]), Mot(M, [0, 1]), Mot(M,
[0, 2])])
```

k-chemins de Dyck (2)

Expérimentation avec Sage :

```
sage: M = MonoideAdditif()
sage: G = tuple([Mot(M, [0, 0]), Mot(M, [0, 1]), Mot(M,
[0, 2])])
sage: P = OperadeMonoide(M, False, G)
sage: print [P.dimension(n) for n in xrange(1, 10)]
[1, 3, 12, 55, 273, 1428, 7752, 43263, 128185]
```

k-chemins de Dyck (2)

Expérimentation avec Sage :

```
sage: M = MonoideAdditif()
sage: G = tuple([Mot(M, [0, 0]), Mot(M, [0, 1]), Mot(M,
[0, 2])])
sage: P = OperadeMonoide(M, False, G)
sage: print [P.dimension(n) for n in xrange(1, 10)]
[1, 3, 12, 55, 273, 1428, 7752, 43263, 128185]

sage: E = list(P.elements(4)); E.sort(); print E
[0000, 0001, 0002, 0010, 0011, 0012, 0013, 0020, 0021,
0022, 0023, 0024, 0100, 0101, 0102, 0110, 0111, 0112,
0113, 0120, 0121, 0122, 0123, 0124, 0130, 0131, 0132,
0133, 0134, 0135, 0200, 0201, 0202, 0210, 0211, 0212,
0213, 0220, 0221, 0222, 0223, 0224, 0230, 0231, 0232,
0233, 0234, 0235, 0240, 0241, 0242, 0243, 0244, 0245,
0246]
```

k-chemins de Dyck (3)

Expérimentation avec Sage :

```
sage: M = MonoideAdditif()
sage: G = tuple([Mot(M, [0, 0]), Mot(M, [0, 1]), Mot(M,
[0, 2]), Mot(M, [0, 3])])
```

k-chemins de Dyck (3)

Expérimentation avec Sage :

```
sage: M = MonoideAdditif()
sage: G = tuple([Mot(M, [0, 0]), Mot(M, [0, 1]), Mot(M,
[0, 2]), Mot(M, [0, 3])])
sage: P = OperadeMonoide(M, False, G)
sage: print [P.dimension(n) for n in xrange(1, 9)]
[1, 4, 22, 140, 969, 7084, 53820, 144927]
```

k-chemins de Dyck (3)

Expérimentation avec Sage :

```
sage: M = MonoideAdditif()
sage: G = tuple([Mot(M, [0, 0]), Mot(M, [0, 1]), Mot(M,
[0, 2]), Mot(M, [0, 3])])
sage: P = OperadeMonoide(M, False, G)
sage: print [P.dimension(n) for n in xrange(1, 9)]
[1, 4, 22, 140, 969, 7084, 53820, 144927]

sage: E = list(P.elements(3)); E.sort(); print E
[000, 001, 002, 003, 010, 011, 012, 013, 014, 020, 021,
022, 023, 024, 025, 030, 031, 032, 033, 034, 035, 036]
```

k-chemins de Dyck (4)

Soit $k \geq 0$ un entier. Soit $FCat^{(k)}$ la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 00, 01, ..., $0k$.

k-chemins de Dyck (4)

Soit $k \geq 0$ un entier. Soit $FCat^{(k)}$ la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 00, 01, …, $0k$.

Proposition

Les éléments de $FCat^{(k)}$ sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $0 \leq x_{i+1} \leq x_i + k$ pour tout $i \in [|x| - 1]$.

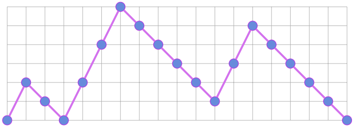
k-chemins de Dyck (4)

Soit $k \geq 0$ un entier. Soit $FCat^{(k)}$ la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 00, 01, ..., $0k$.

Proposition

Les éléments de $FCat^{(k)}$ sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $0 \leq x_{i+1} \leq x_i + k$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les k-chemins de Dyck (2-chemin de Dyck ici) :

 $\longleftrightarrow$

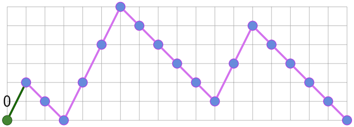
k-chemins de Dyck (4)

Soit $k \geq 0$ un entier. Soit $FCat^{(k)}$ la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 00, 01, …, $0k$.

Proposition

Les éléments de $FCat^{(k)}$ sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $0 \leq x_{i+1} \leq x_i + k$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les k-chemins de Dyck (2-chemin de Dyck ici) :



$\longleftrightarrow \quad 0$

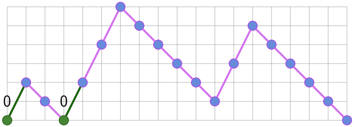
k-chemins de Dyck (4)

Soit $k \geq 0$ un entier. Soit $FCat^{(k)}$ la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 00, 01, ..., $0k$.

Proposition

Les éléments de $FCat^{(k)}$ sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $0 \leq x_{i+1} \leq x_i + k$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les k-chemins de Dyck (2-chemin de Dyck ici) :



$\longleftrightarrow$ 00

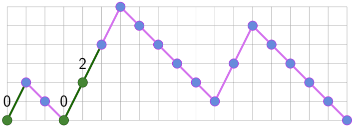
k-chemins de Dyck (4)

Soit $k \geq 0$ un entier. Soit $FCat^{(k)}$ la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 00, 01, ..., $0k$.

Proposition

Les éléments de $FCat^{(k)}$ sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $0 \leq x_{i+1} \leq x_i + k$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les k-chemins de Dyck (2-chemin de Dyck ici) :



$\longleftrightarrow$ 002

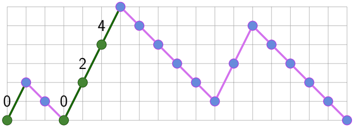
k-chemins de Dyck (4)

Soit $k \geq 0$ un entier. Soit $FCat^{(k)}$ la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 00, 01, ..., $0k$.

Proposition

Les éléments de $FCat^{(k)}$ sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $0 \leq x_{i+1} \leq x_i + k$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les k-chemins de Dyck (2-chemin de Dyck ici) :



$\longleftrightarrow$ 0024

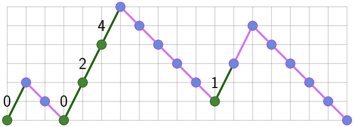
k-chemins de Dyck (4)

Soit $k \geq 0$ un entier. Soit $FCat^{(k)}$ la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 00, 01, ..., $0k$.

Proposition

Les éléments de $FCat^{(k)}$ sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $0 \leq x_{i+1} \leq x_i + k$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les k-chemins de Dyck (2-chemin de Dyck ici) :



$\longleftrightarrow$ 00241

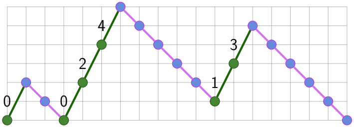
k-chemins de Dyck (4)

Soit $k \geq 0$ un entier. Soit $FCat^{(k)}$ la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 00, 01, ..., $0k$.

Proposition

Les éléments de $FCat^{(k)}$ sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $0 \leq x_{i+1} \leq x_i + k$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les k-chemins de Dyck (2-chemin de Dyck ici) :



$\longleftrightarrow$ 002413

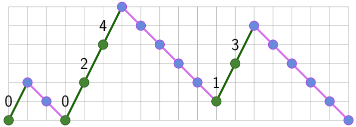
k-chemins de Dyck (4)

Soit $k \geq 0$ un entier. Soit $FCat^{(k)}$ la sous-opérade ns de $\mathrm{T}\mathbb{N}$ engendrée par 00, 01, ..., $0k$.

Proposition

Les éléments de $FCat^{(k)}$ sont exactement les mots x sur l'alphabet $\mathbb{N}$ qui vérifient $x_1 = 0$ et $0 \leq x_{i+1} \leq x_i + k$ pour tout $i \in [|x| - 1]$.

Bijection entre ces mots de longueur n et les k-chemins de Dyck (2-chemin de Dyck ici) :



$\longleftrightarrow$ 002413

$\rightsquigarrow$ $FCat^{(k)}$ est une opérade ns sur les k-chemins de Dyck. Dimensions : nombres de Fuss-Catalan.

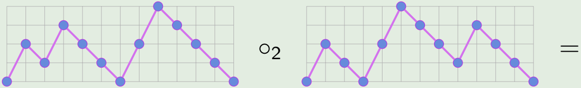
k-chemins de Dyck (5)

La substitution partielle dans $FCat^{(k)}$ se reformule en termes d'insertions dans des k-chemins de Dyck.

k-chemins de Dyck (5)

La substitution partielle dans $FCat^{(k)}$ se reformule en termes d'insertions dans des k-chemins de Dyck.

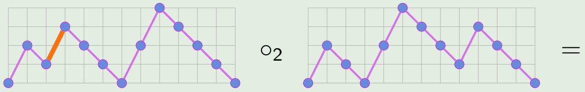
Exemple



k-chemins de Dyck (5)

La substitution partielle dans $FCat^{(k)}$ se reformule en termes d'insertions dans des k-chemins de Dyck.

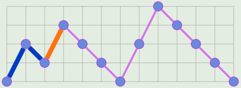
Exemple



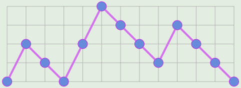
k-chemins de Dyck (5)

La substitution partielle dans $FCat^{(k)}$ se reformule en termes d'insertions dans des k-chemins de Dyck.

Exemple



$\circ_2$

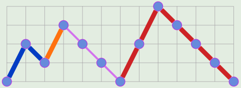
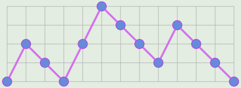


$=$

k-chemins de Dyck (5)

La substitution partielle dans $FCat^{(k)}$ se reformule en termes d'insertions dans des k-chemins de Dyck.

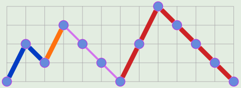
Exemple

 $\circ_2$  $=$

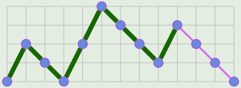
k-chemins de Dyck (5)

La substitution partielle dans $FCat^{(k)}$ se reformule en termes d'insertions dans des k-chemins de Dyck.

Exemple



$\circ_2$

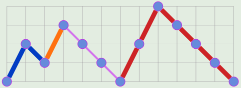


$=$

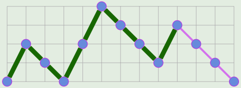
k-chemins de Dyck (5)

La substitution partielle dans $FCat^{(k)}$ se reformule en termes d'insertions dans des k-chemins de Dyck.

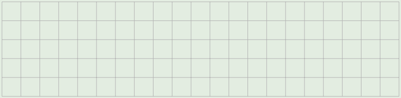
Exemple



$\circ_2$



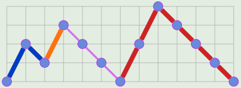
$=$



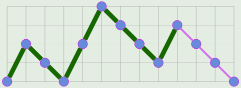
k-chemins de Dyck (5)

La substitution partielle dans $FCat^{(k)}$ se reformule en termes d'insertions dans des k-chemins de Dyck.

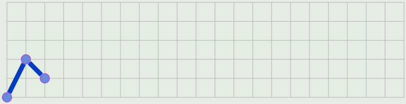
Exemple



$\circ_2$



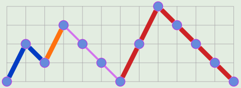
$=$



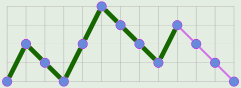
k-chemins de Dyck (5)

La substitution partielle dans $FCat^{(k)}$ se reformule en termes d'insertions dans des k-chemins de Dyck.

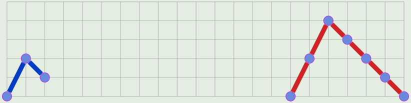
Exemple



$\circ_2$



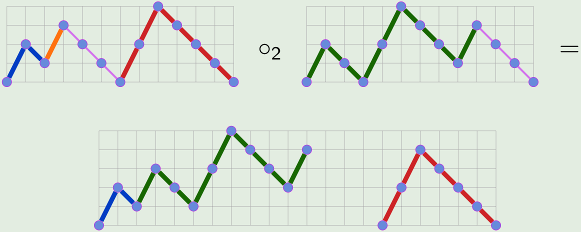
$=$



k-chemins de Dyck (5)

La substitution partielle dans $FCat^{(k)}$ se reformule en termes d'insertions dans des k-chemins de Dyck.

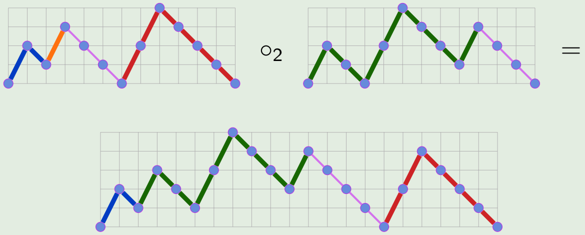
Exemple



k-chemins de Dyck (5)

La substitution partielle dans $FCat^{(k)}$ se reformule en termes d'insertions dans des k-chemins de Dyck.

Exemple



k-chemins de Dyck (6)

Remarque

- *L'opérade ns* $\mathsf{FCat}^{(k)}$ *est une sous-opérade ns de* $\mathsf{FCat}^{(k+1)}$.
- *L'opérade* $\mathsf{FCat}^{(0)}$ *est l'opérade commutative associative* Com.
- *L'opérade ns* APE *est une sous-opérade ns de* $\mathsf{FCat}^{(1)}$.

k-chemins de Dyck (7)

Théorème

L'opérade ns $\mathsf{FCat}^{(1)}$ *est l'opérade libre sur deux générateurs* $\square$ *et* $\blacksquare$ *d'arité* 2 *soumis uniquement aux relations*

$$
\begin{aligned}
\square \circ_1 \square &= \square \circ_2 \square\,,\\
\blacksquare \circ_1 \square &= \square \circ_2 \blacksquare\,,\\
\blacksquare \circ_1 \blacksquare &= \blacksquare \circ_2 \square\,.
\end{aligned}
$$

k-chemins de Dyck (7)

Théorème

L'opérade ns $FCat^{(1)}$ *est l'opérade libre sur deux générateurs* $\square$ *et* $\blacksquare$ *d'arité* 2 *soumis uniquement aux relations*

$$\square \circ_1 \square = \square \circ_2 \square,$$
$$\blacksquare \circ_1 \square = \square \circ_2 \blacksquare,$$
$$\blacksquare \circ_1 \blacksquare = \blacksquare \circ_2 \square.$$

$\rightsquigarrow$ toute algèbre sur l'opérade $FCat^{(1)}$ est un espace vectoriel V muni de deux applications linéaires $\square : V \otimes V \to V$ et $\blacksquare : V \otimes V \to V$ qui vérifient pour tous $x, y, z \in V$ les relations

$$(x \,\square\, y) \,\square\, z = x \,\square\, (y \,\square\, z),$$
$$(x \,\square\, y) \,\blacksquare\, z = x \,\square\, (y \,\blacksquare\, z),$$
$$(x \,\blacksquare\, y) \,\blacksquare\, z = x \,\blacksquare\, (y \,\square\, z).$$

k-chemins de Dyck (8)

On note α_ℓ le générateur 0ℓ de $FCat^{(k)}$ pour tout $0 \leq \ell \leq k$.

k-chemins de Dyck (8)

On note α_ℓ le générateur 0ℓ de $FCat^{(k)}$ pour tout $0 \leq \ell \leq k$.

Travail en cours

Il semble que l'opérade ns $FCat^{(k)}$ soit l'opérade libre sur les générateurs α_0, α_1, ..., α_k d'arité 2 soumis uniquement aux relations

$$\alpha_i \circ_1 \alpha_j = \alpha_j \circ_2 \alpha_{i-j},$$

pour tous $0 \leq j \leq i \leq k$.

Animaux dirigés (1)

Expérimentation avec Sage :

```
sage: M = MonoideAdditifModulo(3)
```

Animaux dirigés (1)

Expérimentation avec Sage :

```
sage: M = MonoideAdditifModulo(3)
sage: G = tuple([Mot(M, [0, 0]), Mot(M, [0, 1])])
```

Animaux dirigés (1)

Expérimentation avec Sage :

```
sage: M = MonoideAdditifModulo(3)
sage: G = tuple([Mot(M, [0, 0]), Mot(M, [0, 1])])
sage: P = OperadeMonoide(M, False, G)
sage: print [P.dimension(n) for n in xrange(1, 10)]
[1, 2, 5, 13, 35, 96, 267, 750, 2123]
```

Animaux dirigés (1)

Expérimentation avec Sage :

```
sage: M = MonoideAdditifModulo(3)
sage: G = tuple([Mot(M, [0, 0]), Mot(M, [0, 1])])
sage: P = OperadeMonoide(M, False, G)
sage: print [P.dimension(n) for n in xrange(1, 10)]
[1, 2, 5, 13, 35, 96, 267, 750, 2123]

sage: E = list(P.elements(5)); E.sort(); print E
[00000, 00001, 00010, 00011, 00012, 00100, 00101, 00110,
00111, 00112, 00120, 00121, 00122, 01000, 01001, 01010,
01011, 01012, 01100, 01101, 01110, 01111, 01112, 01120,
01121, 01122, 01200, 01201, 01202, 01210, 01211, 01212,
01220, 01221, 01222]
```

Animaux dirigés (2)

Soit *AnD* la sous-opérade ns de $\mathrm{T}\mathbb{N}_3$ engendrée par 00 et 01.

Animaux dirigés (2)

Soit AnD la sous-opérade ns de $\mathrm{T}\mathbb{N}_3$ engendrée par 00 et 01.

Animal dirigé : sous-ensemble A de $\mathbb{N}^2$ tel que $(0,0) \in A$ et

$$(i,j) \in A \text{ avec } i \geq 1 \text{ ou } j \geq 1 \quad \Longrightarrow \quad (i-1,j) \in A \text{ ou } (i,j-1) \in A.$$

Animaux dirigés (2)

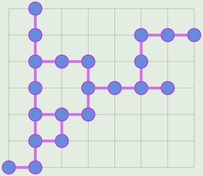
Soit AnD la sous-opérade ns de $\mathrm{T}\mathbb{N}_3$ engendrée par 00 et 01.

Animal dirigé : sous-ensemble A de $\mathbb{N}^2$ tel que $(0,0) \in A$ et

$$(i,j) \in A \text{ avec } i \geq 1 \text{ ou } j \geq 1 \quad \Longrightarrow \quad (i-1,j) \in A \text{ ou } (i,j-1) \in A.$$

Exemple

Un animal dirigé de taille 21 :



Animaux dirigés (3)

Bijection [**Gouyou-Beauchamps et Viennot**, 88] :

$$\begin{array}{c}\text{animaux dirigés}\\ \text{de taille } n\end{array} \quad \longleftrightarrow \quad \begin{array}{c}\text{mots de } \mathcal{L}\\ \text{de longueur } n-1,\end{array}$$

où $\mathcal{L}$ est le langage des mots x sur l'alphabet $\{-1, 0, 1\}$ tel que pour tout préfixe $x_1 \ldots x_\ell$ de x, l'entier $x_1 + \cdots + x_\ell$ est positif.

Animaux dirigés (3)

Bijection [**Gouyou-Beauchamps et Viennot**, 88] :

$$\begin{array}{c}\text{animaux dirigés}\\ \text{de taille } n\end{array} \longleftrightarrow \begin{array}{c}\text{mots de } \mathcal{L}\\ \text{de longueur } n-1,\end{array}$$

où $\mathcal{L}$ est le langage des mots x sur l'alphabet $\{-1, 0, 1\}$ tel que pour tout préfixe $x_1 \dots x_\ell$ de x, l'entier $x_1 + \cdots + x_\ell$ est positif.

Exemple

- $0 1 -1 1 0 -1 1 \in \mathcal{L}$;

Animaux dirigés (3)

Bijection [**Gouyou-Beauchamps et Viennot**, 88] :

$$\begin{array}{c}\text{animaux dirigés}\\ \text{de taille } n\end{array} \quad\longleftrightarrow\quad \begin{array}{c}\text{mots de } \mathcal{L}\\ \text{de longueur } n-1,\end{array}$$

où $\mathcal{L}$ est le langage des mots x sur l'alphabet $\{-1, 0, 1\}$ tel que pour tout préfixe $x_1 \ldots x_\ell$ de x, l'entier $x_1 + \cdots + x_\ell$ est positif.

Exemple

- $0\,1\,{-1}\,1\,0\,{-1}\,1 \in \mathcal{L}$;
- $1\,0\,{-1}\,{-1}\,1\,1\,1\,1 \notin \mathcal{L}$.

Animaux dirigés (4)

Soit l'application

$$\phi : \{0, 1, 2\}^2 \to \{-1, 0, 1\},$$

définie par

$$\phi(\mathtt{a} \cdot \mathtt{b}) := \begin{cases} \mathtt{b} - \mathtt{a} & \text{si } |\mathtt{b} - \mathtt{a}| \leq 1, \\ 1 & \text{si } \mathtt{a} = 2 \text{ et } \mathtt{b} = 0, \\ -1 & \text{sinon } (\mathtt{a} = 0 \text{ et } \mathtt{b} = 2). \end{cases}$$

Animaux dirigés (4)

Soit l'application

$$\phi : \{0, 1, 2\}^2 \to \{-1, 0, 1\},$$

définie par

$$\phi(\mathtt{a} \cdot \mathtt{b}) := \begin{cases} \mathtt{b} - \mathtt{a} & \text{si } |\mathtt{b} - \mathtt{a}| \leq 1, \\ 1 & \text{si } \mathtt{a} = 2 \text{ et } \mathtt{b} = 0, \\ -1 & \text{sinon } (\mathtt{a} = 0 \text{ et } \mathtt{b} = 2). \end{cases}$$

On l'étend en l'application

$$\phi : \{0, 1, 2\}^n \to \{-1, 0, 1\}^{n-1},$$

en posant pour tous $\mathtt{a}, \mathtt{b} \in \{0, 1, 2\}$ et $u \in \{0, 1, 2\}^*$,

$$\phi(\mathtt{a}) := \epsilon \quad \text{et} \quad \phi(\mathtt{a} \cdot \mathtt{b} \cdot u) := \phi(\mathtt{a} \cdot \mathtt{b}) \cdot \phi(\mathtt{b} \cdot u).$$

Animaux dirigés (4)

Soit l'application

$$\phi : \{0,1,2\}^2 \to \{-1,0,1\},$$

définie par

$$\phi(\mathtt{a} \cdot \mathtt{b}) := \begin{cases} \mathtt{b} - \mathtt{a} & \text{si } |\mathtt{b} - \mathtt{a}| \leq 1, \\ 1 & \text{si } \mathtt{a} = 2 \text{ et } \mathtt{b} = 0, \\ -1 & \text{sinon } (\mathtt{a} = 0 \text{ et } \mathtt{b} = 2). \end{cases}$$

On l'étend en l'application

$$\phi : \{0,1,2\}^n \to \{-1,0,1\}^{n-1},$$

en posant pour tous $\mathtt{a}, \mathtt{b} \in \{0,1,2\}$ et $u \in \{0,1,2\}^*$,

$$\phi(\mathtt{a}) := \epsilon \quad \text{et} \quad \phi(\mathtt{a} \cdot \mathtt{b} \cdot u) := \phi(\mathtt{a} \cdot \mathtt{b}) \cdot \phi(\mathtt{b} \cdot u).$$

Exemple

$$\phi(011220201) =$$

Animaux dirigés (4)

Soit l'application

$$\phi : \{0,1,2\}^2 \to \{-1,0,1\},$$

définie par

$$\phi(\mathtt{a}\cdot\mathtt{b}) := \begin{cases} \mathtt{b}-\mathtt{a} & \text{si } |\mathtt{b}-\mathtt{a}| \leq 1, \\ 1 & \text{si } \mathtt{a}=2 \text{ et } \mathtt{b}=0, \\ -1 & \text{sinon } (\mathtt{a}=0 \text{ et } \mathtt{b}=2). \end{cases}$$

On l'étend en l'application

$$\phi : \{0,1,2\}^n \to \{-1,0,1\}^{n-1},$$

en posant pour tous $\mathtt{a}, \mathtt{b} \in \{0,1,2\}$ et $u \in \{0,1,2\}^*$,

$$\phi(\mathtt{a}) := \epsilon \quad \text{et} \quad \phi(\mathtt{a}\cdot\mathtt{b}\cdot u) := \phi(\mathtt{a}\cdot\mathtt{b})\cdot\phi(\mathtt{b}\cdot u).$$

Exemple

$$\phi(011220201) = 1$$

Animaux dirigés (4)

Soit l'application

$$\phi : \{0,1,2\}^2 \to \{-1,0,1\},$$

définie par

$$\phi(\mathtt{a} \cdot \mathtt{b}) := \begin{cases} \mathtt{b} - \mathtt{a} & \text{si } |\mathtt{b} - \mathtt{a}| \leq 1, \\ 1 & \text{si } \mathtt{a} = 2 \text{ et } \mathtt{b} = 0, \\ -1 & \text{sinon } (\mathtt{a} = 0 \text{ et } \mathtt{b} = 2). \end{cases}$$

On l'étend en l'application

$$\phi : \{0,1,2\}^n \to \{-1,0,1\}^{n-1},$$

en posant pour tous $\mathtt{a}, \mathtt{b} \in \{0,1,2\}$ et $u \in \{0,1,2\}^*$,

$$\phi(\mathtt{a}) := \epsilon \quad \text{et} \quad \phi(\mathtt{a} \cdot \mathtt{b} \cdot u) := \phi(\mathtt{a} \cdot \mathtt{b}) \cdot \phi(\mathtt{b} \cdot u).$$

Exemple

$$\phi(011220201) = 10$$

Animaux dirigés (4)

Soit l'application

$$\phi : \{0,1,2\}^2 \to \{-1,0,1\},$$

définie par

$$\phi(\mathtt{a}\cdot\mathtt{b}) := \begin{cases} \mathtt{b}-\mathtt{a} & \text{si } |\mathtt{b}-\mathtt{a}| \leq 1, \\ 1 & \text{si } \mathtt{a}=2 \text{ et } \mathtt{b}=0, \\ -1 & \text{sinon } (\mathtt{a}=0 \text{ et } \mathtt{b}=2). \end{cases}$$

On l'étend en l'application

$$\phi : \{0,1,2\}^n \to \{-1,0,1\}^{n-1},$$

en posant pour tous $\mathtt{a},\mathtt{b} \in \{0,1,2\}$ et $u \in \{0,1,2\}^*$,

$$\phi(\mathtt{a}) := \epsilon \quad \text{et} \quad \phi(\mathtt{a}\cdot\mathtt{b}\cdot u) := \phi(\mathtt{a}\cdot\mathtt{b})\cdot\phi(\mathtt{b}\cdot u).$$

Exemple

$$\phi(011220201) = 101$$

Animaux dirigés (4)

Soit l'application

$$\phi : \{0,1,2\}^2 \to \{-1,0,1\},$$

définie par

$$\phi(\mathtt{a} \cdot \mathtt{b}) := \begin{cases} \mathtt{b} - \mathtt{a} & \text{si } |\mathtt{b} - \mathtt{a}| \leq 1, \\ 1 & \text{si } \mathtt{a} = 2 \text{ et } \mathtt{b} = 0, \\ -1 & \text{sinon } (\mathtt{a} = 0 \text{ et } \mathtt{b} = 2). \end{cases}$$

On l'étend en l'application

$$\phi : \{0,1,2\}^n \to \{-1,0,1\}^{n-1},$$

en posant pour tous $\mathtt{a}, \mathtt{b} \in \{0,1,2\}$ et $u \in \{0,1,2\}^*$,

$$\phi(\mathtt{a}) := \epsilon \quad \text{et} \quad \phi(\mathtt{a} \cdot \mathtt{b} \cdot u) := \phi(\mathtt{a} \cdot \mathtt{b}) \cdot \phi(\mathtt{b} \cdot u).$$

Exemple

$$\phi(011220201) = 1010$$

Animaux dirigés (4)

Soit l'application

$$\phi : \{0,1,2\}^2 \to \{-1,0,1\},$$

définie par

$$\phi(\mathtt{a} \cdot \mathtt{b}) := \begin{cases} \mathtt{b} - \mathtt{a} & \text{si } |\mathtt{b} - \mathtt{a}| \leq 1, \\ 1 & \text{si } \mathtt{a} = 2 \text{ et } \mathtt{b} = 0, \\ -1 & \text{sinon } (\mathtt{a} = 0 \text{ et } \mathtt{b} = 2). \end{cases}$$

On l'étend en l'application

$$\phi : \{0,1,2\}^n \to \{-1,0,1\}^{n-1},$$

en posant pour tous $\mathtt{a}, \mathtt{b} \in \{0,1,2\}$ et $u \in \{0,1,2\}^*$,

$$\phi(\mathtt{a}) := \epsilon \quad \text{et} \quad \phi(\mathtt{a} \cdot \mathtt{b} \cdot u) := \phi(\mathtt{a} \cdot \mathtt{b}) \cdot \phi(\mathtt{b} \cdot u).$$

Exemple

$$\phi(01122 0201) = 10101$$

Animaux dirigés (4)

Soit l'application

$$\phi : \{0,1,2\}^2 \to \{-1,0,1\},$$

définie par

$$\phi(\mathtt{a} \cdot \mathtt{b}) := \begin{cases} \mathtt{b} - \mathtt{a} & \text{si } |\mathtt{b} - \mathtt{a}| \leq 1, \\ 1 & \text{si } \mathtt{a} = 2 \text{ et } \mathtt{b} = 0, \\ -1 & \text{sinon } (\mathtt{a} = 0 \text{ et } \mathtt{b} = 2). \end{cases}$$

On l'étend en l'application

$$\phi : \{0,1,2\}^n \to \{-1,0,1\}^{n-1},$$

en posant pour tous $\mathtt{a}, \mathtt{b} \in \{0,1,2\}$ et $u \in \{0,1,2\}^*$,

$$\phi(\mathtt{a}) := \epsilon \quad \text{et} \quad \phi(\mathtt{a} \cdot \mathtt{b} \cdot u) := \phi(\mathtt{a} \cdot \mathtt{b}) \cdot \phi(\mathtt{b} \cdot u).$$

Exemple

$$\phi(01122\,02\,01) = 10101\;-1$$

Animaux dirigés (4)

Soit l'application

$$\phi : \{0, 1, 2\}^2 \to \{-1, 0, 1\},$$

définie par

$$\phi(\mathtt{a} \cdot \mathtt{b}) := \begin{cases} \mathtt{b} - \mathtt{a} & \text{si } |\mathtt{b} - \mathtt{a}| \leq 1, \\ 1 & \text{si } \mathtt{a} = 2 \text{ et } \mathtt{b} = 0, \\ -1 & \text{sinon } (\mathtt{a} = 0 \text{ et } \mathtt{b} = 2). \end{cases}$$

On l'étend en l'application

$$\phi : \{0, 1, 2\}^n \to \{-1, 0, 1\}^{n-1},$$

en posant pour tous $\mathtt{a}, \mathtt{b} \in \{0, 1, 2\}$ et $u \in \{0, 1, 2\}^*$,

$$\phi(\mathtt{a}) := \epsilon \quad \text{et} \quad \phi(\mathtt{a} \cdot \mathtt{b} \cdot u) := \phi(\mathtt{a} \cdot \mathtt{b}) \cdot \phi(\mathtt{b} \cdot u).$$

Exemple

$$\phi(011220201) = 10101\,{-1}1$$

Animaux dirigés (4)

Soit l'application

$$\phi : \{0, 1, 2\}^2 \to \{-1, 0, 1\},$$

définie par

$$\phi(\mathtt{a} \cdot \mathtt{b}) := \begin{cases} \mathtt{b} - \mathtt{a} & \text{si } |\mathtt{b} - \mathtt{a}| \leq 1, \\ 1 & \text{si } \mathtt{a} = 2 \text{ et } \mathtt{b} = 0, \\ -1 & \text{sinon } (\mathtt{a} = 0 \text{ et } \mathtt{b} = 2). \end{cases}$$

On l'étend en l'application

$$\phi : \{0, 1, 2\}^n \to \{-1, 0, 1\}^{n-1},$$

en posant pour tous $\mathtt{a}, \mathtt{b} \in \{0, 1, 2\}$ et $u \in \{0, 1, 2\}^*$,

$$\phi(\mathtt{a}) := \epsilon \quad \text{et} \quad \phi(\mathtt{a} \cdot \mathtt{b} \cdot u) := \phi(\mathtt{a} \cdot \mathtt{b}) \cdot \phi(\mathtt{b} \cdot u).$$

Exemple

$$\phi(011220201) = 10101\,{-1}11$$

Animaux dirigés (5)

Proposition

L'application ϕ définit une bijection entre les éléments de $AnD(n)$ et les mots de $\mathcal{L}$ de longueur $n-1$.

Animaux dirigés (5)

Proposition

L'application ϕ définit une bijection entre les éléments de $AnD(n)$ et les mots de $\mathcal{L}$ de longueur $n-1$.

$\rightsquigarrow$ AnD est une opérade sur les animaux dirigés.

Animaux dirigés (5)

Proposition

L'application ϕ définit une bijection entre les éléments de $AnD(n)$ et les mots de $\mathcal{L}$ de longueur $n-1$.

$\rightsquigarrow$ *AnD* est une opérade sur les animaux dirigés.

Remarque

L'opérade ns AnD est un quotient de l'opérade ns $FCat^{(1)}$.

Animaux dirigés (5)

Proposition

L'application ϕ définit une bijection entre les éléments de $AnD(n)$ et les mots de $\mathcal{L}$ de longueur $n-1$.

$\rightsquigarrow$ *AnD* est une opérade sur les animaux dirigés.

Remarque

L'opérade ns AnD est un quotient de l'opérade ns $FCat^{(1)}$.

Travail en cours

Interpréter la substitution partielle de *AnD* en termes d'animaux dirigés.

L'opérade diassociative (1)

Expérimentation avec Sage :

```
sage: M = MonoideMultiplicatif()
```

L'opérade diassociative (1)

Expérimentation avec Sage :

```
sage: M = MonoideMultiplicatif()
sage: G = tuple([Mot(M, [0, 1]), Mot(M, [1, 0])])
```

L'opérade diassociative (1)

Expérimentation avec Sage :

```
sage: M = MonoideMultiplicatif()
sage: G = tuple([Mot(M, [0, 1]), Mot(M, [1, 0])])
sage: P = OperadeMonoide(M, False, G)
sage: print [P.dimension(n) for n in xrange(1, 10)]
[1, 2, 3, 4, 5, 6, 7, 8, 9]
```

L'opérade diassociative (1)

Expérimentation avec Sage :

```
sage: M = MonoideMultiplicatif()
sage: G = tuple([Mot(M, [0, 1]), Mot(M, [1, 0])])
sage: P = OperadeMonoide(M, False, G)
sage: print [P.dimension(n) for n in xrange(1, 10)]
[1, 2, 3, 4, 5, 6, 7, 8, 9]

sage: E = list(P.elements(8)); E.sort(); print E
[00000001, 00000010, 00000100, 00001000, 00010000,
00100000, 01000000, 10000000]
```

L'opérade diassociative (2)

Soit M le monoïde multiplicatif restreint à l'ensemble $\{0, 1\}$.

L'opérade diassociative (2)

Soit M le monoïde multiplicatif restreint à l'ensemble $\{0, 1\}$.

Soit D la sous-opérade ns de $\mathsf{T}M$ engendrée par 01 et 10.

L’opérade diassociative (2)

Soit M le monoïde multiplicatif restreint à l’ensemble $\{0, 1\}$.

Soit D la sous-opérade ns de $\mathsf{T}M$ engendrée par 01 et 10.

Proposition

Les éléments de l’opérade ns D sont exactement les mots sur l’alphabet $\{0, 1\}$ qui admettent une unique occurrence de 1.

L'opérade diassociative (2)

Soit M le monoïde multiplicatif restreint à l'ensemble $\{0, 1\}$.

Soit D la sous-opérade ns de $\mathsf{T}M$ engendrée par 01 et 10.

Proposition

Les éléments de l'opérade ns D sont exactement les mots sur l'alphabet $\{0, 1\}$ qui admettent une unique occurrence de 1.

$\rightsquigarrow$ implique $\dim D(n) = n$.

L'opérade diassociative (3)

L'opérade diassociative *Dias* [**Loday**, 01] est l'opérade ns engendrée par les générateurs $\dashv$ et $\vdash$ soumis exactement aux relations

$$\begin{aligned}
\dashv \circ_1 \dashv \; &= \; \dashv \circ_2 \dashv \; = \; \dashv \circ_2 \vdash, \\
\dashv \circ_1 \vdash \; &= \; \vdash \circ_2 \dashv, \\
\vdash \circ_2 \vdash \; &= \; \vdash \circ_1 \vdash \; = \; \vdash \circ_1 \dashv .
\end{aligned}$$

L'opérade diassociative (3)

L'opérade diassociative *Dias* [**Loday**, 01] est l'opérade ns engendrée par les générateurs $\dashv$ et $\vdash$ soumis exactement aux relations

$$
\begin{aligned}
\dashv \circ_1 \dashv &= \dashv \circ_2 \dashv \;= \dashv \circ_2 \vdash, \\
\dashv \circ_1 \vdash &= \vdash \circ_2 \dashv, \\
\vdash \circ_2 \vdash &= \vdash \circ_1 \vdash \;= \vdash \circ_1 \dashv .
\end{aligned}
$$

Proposition

Les opérades ns Dias et D sont isomorphes. L'application ψ : *Dias* $\to$ *D définie par* $\psi(\dashv) := 10$ *et* $\psi(\vdash) := 01$ *est un isomorphisme.*

L'opérade diassociative (3)

L'opérade diassociative *Dias* [**Loday**, 01] est l'opérade ns engendrée par les générateurs $\dashv$ et $\vdash$ soumis exactement aux relations

$$\begin{aligned} \dashv \circ_1 \dashv &= \dashv \circ_2 \dashv \; = \dashv \circ_2 \vdash, \\ \dashv \circ_1 \vdash &= \vdash \circ_2 \dashv, \\ \vdash \circ_2 \vdash &= \vdash \circ_1 \vdash \; = \vdash \circ_1 \dashv . \end{aligned}$$

Proposition

Les opérades ns Dias et D sont isomorphes. L'application ψ : *Dias* $\to$ *D définie par* $\psi(\dashv) := 10$ *et* $\psi(\vdash) := 01$ *est un isomorphisme.*

$\rightsquigarrow$ *D* offre une réalisation simple de *Dias*.

Quelques perspectives

- Plusieurs sous-opérades (symétriques ou non symétriques) de $\mathrm{T}\mathbb{N}$ possèdent une série de Hilbert non connue $\rightsquigarrow$ découvrir les objets combinatoires mis en jeu ;

Quelques perspectives

- Plusieurs sous-opérades (symétriques ou non symétriques) de $\mathsf{T}\mathbb{N}$ possèdent une série de Hilbert non connue $\rightsquigarrow$ découvrir les objets combinatoires mis en jeu ;
- s'intéresser aux sous-opérades de $\mathsf{T}\{\mathtt{a}, \mathtt{b}\}^*$;

Quelques perspectives

- Plusieurs sous-opérades (symétriques ou non symétriques) de $\mathsf{T}\mathbb{N}$ possèdent une série de Hilbert non connue $\rightsquigarrow$ découvrir les objets combinatoires mis en jeu ;
- s'intéresser aux sous-opérades de $\mathsf{T}\{\mathtt{a}, \mathtt{b}\}^*$;
- démontrer les présentations des opérades obtenues qui semblent binaires et quadratiques ;

Quelques perspectives

- Plusieurs sous-opérades (symétriques ou non symétriques) de $\mathsf{T}\mathbb{N}$ possèdent une série de Hilbert non connue $\rightsquigarrow$ découvrir les objets combinatoires mis en jeu ;
- s'intéresser aux sous-opérades de $\mathsf{T}\{\mathtt{a}, \mathtt{b}\}^*$;
- démontrer les présentations des opérades obtenues qui semblent binaires et quadratiques ;
- donner un critère pour que les sous-opérades de $\mathsf{T}M$ soient de Koszul ;

Quelques perspectives

- Plusieurs sous-opérades (symétriques ou non symétriques) de $\mathsf{T}\mathbb{N}$ possèdent une série de Hilbert non connue $\rightsquigarrow$ découvrir les objets combinatoires mis en jeu ;
- s'intéresser aux sous-opérades de $\mathsf{T}\{\mathtt{a}, \mathtt{b}\}^*$;
- démontrer les présentations des opérades obtenues qui semblent binaires et quadratiques ;
- donner un critère pour que les sous-opérades de $\mathsf{T}M$ soient de Koszul ;
- comprendre la structure des algèbres sur $\mathsf{T}M$ ainsi que la structure des algèbres sur des sous-opérades de $\mathsf{T}M$;

Quelques perspectives

- Plusieurs sous-opérades (symétriques ou non symétriques) de $\mathsf{T}\mathbb{N}$ possèdent une série de Hilbert non connue $\rightsquigarrow$ découvrir les objets combinatoires mis en jeu ;
- s'intéresser aux sous-opérades de $\mathsf{T}\{\mathtt{a}, \mathtt{b}\}^*$;
- démontrer les présentations des opérades obtenues qui semblent binaires et quadratiques ;
- donner un critère pour que les sous-opérades de $\mathsf{T}M$ soient de Koszul ;
- comprendre la structure des algèbres sur $\mathsf{T}M$ ainsi que la structure des algèbres sur des sous-opérades de $\mathsf{T}M$;
- étendre le foncteur T en un foncteur de la catégorie des algèbres vers la catégorie des opérades vectorielles.