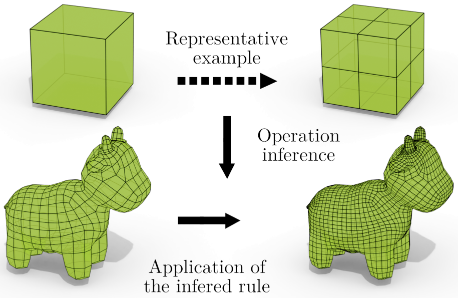
Une approche pour inférer les expressions de calcul géométrique en modélisation à base topologique

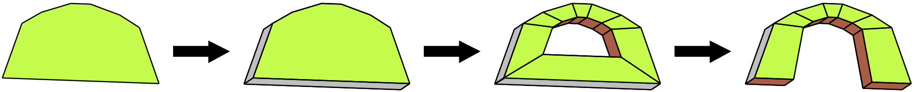
AFADL 2023

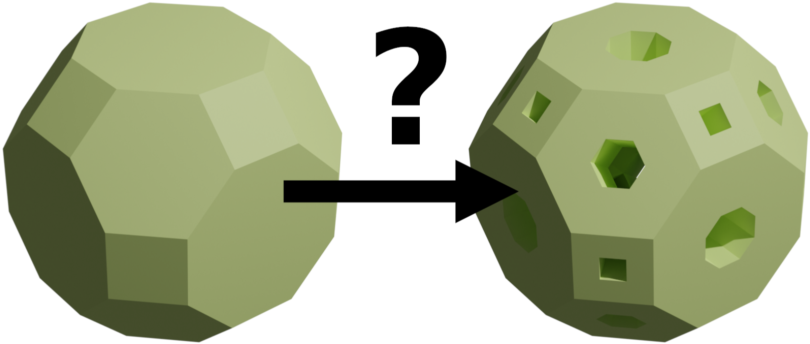


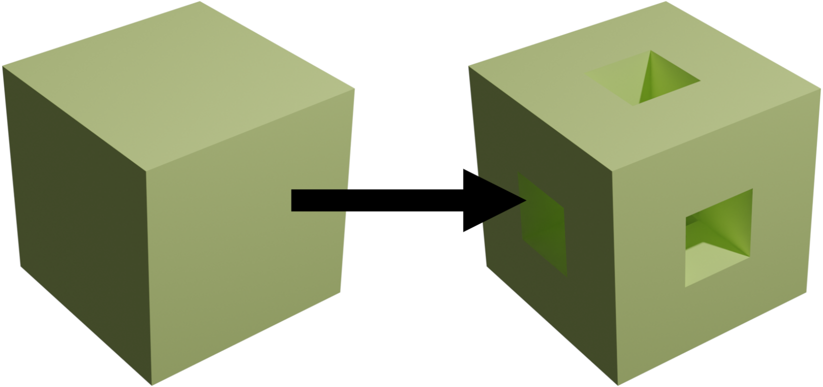
Romain Pascual

Pascale Le Gall, Hakim Belhaouari, et Agnès Arnould

8 Juin, 2023

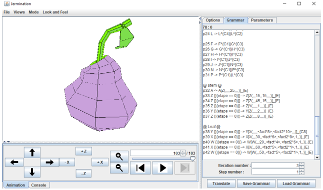




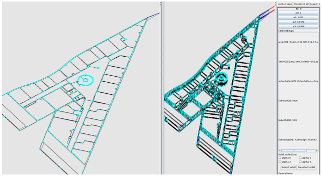


Jerboa

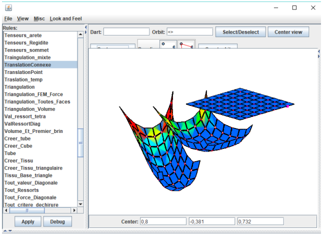
Croissance de plantes



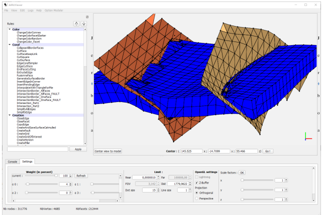
Architecture

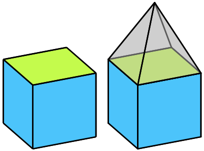


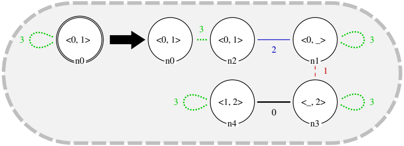
Simulation physique



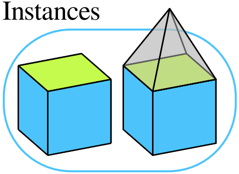
Géologie

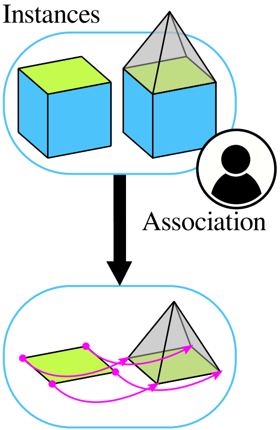


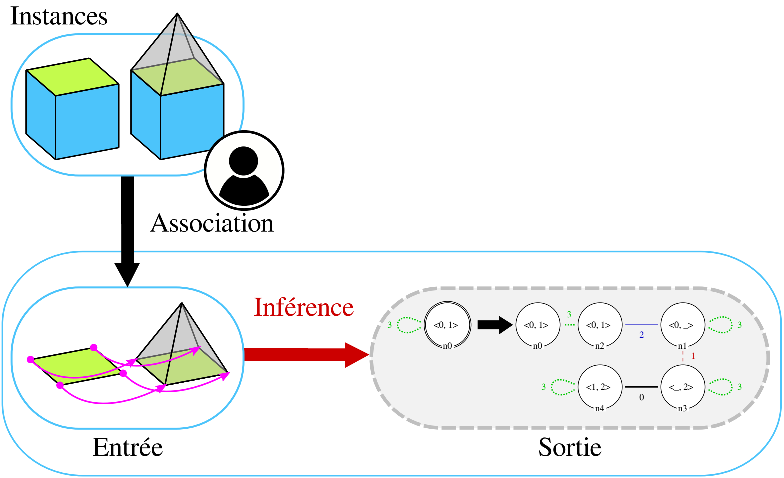


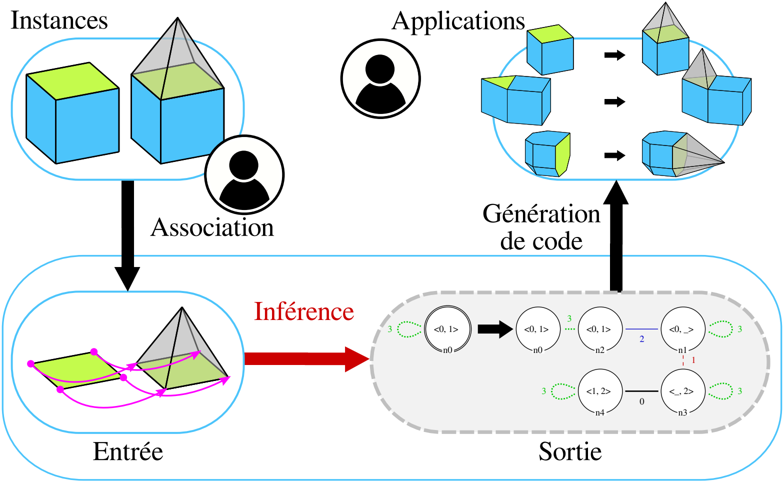

<0, 1>
n0
<0, 1>
n0
<0, 1>
n2
<0, _>
n1
<1, 2>
n4
<_, 2>
n3
3
3
2
3
1
3
0
3

Instances




Instances
Association

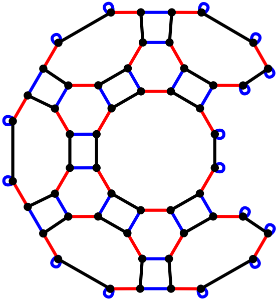

Instances
Association
Inférence
Entrée
Sortie
<0, 1>
n0
<0, 1>
n0
<0, 1>
n2
<0, _>
n1
<1, 2>
n4
<_, 2>
n3


Instances
Applications
Association
Génération de code
Inférence
<0, 1>
n0
<0, 1>
n0
<0, 1>
n2
<0, _>
n1
<1, 2>
n4
<_, 2>
n3
Entrée
Sortie

Cartes généralisées plongées

- Comment représenter les objets ?

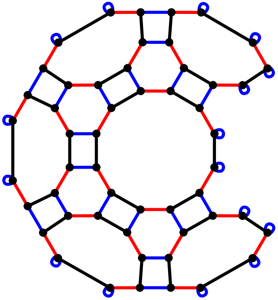
Cartes généralisées[1] (topologie)



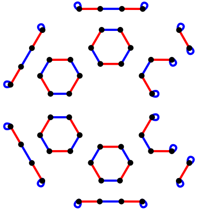
Légende : 0, 1, 2.

[1] Damiand et al. 2014.

Cartes généralisées[1] (topologie)



Orbite : Sous-graphe induit par un sous-ensemble $\langle o \rangle$ de dimensions

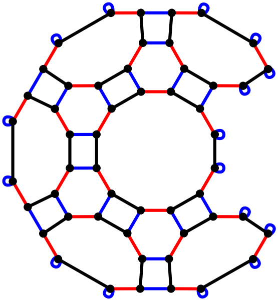


Légende : 0, 1, 2. Sommets : orbites $\langle 1, 2 \rangle$

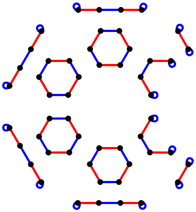
[1]Damiand et al. 2014.

Cartes généralisées[1] (topologie)

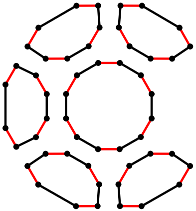
Orbite : Sous-graphe induit par un sous-ensemble $\langle o \rangle$ de dimensions



Légende : 0, 1, 2.



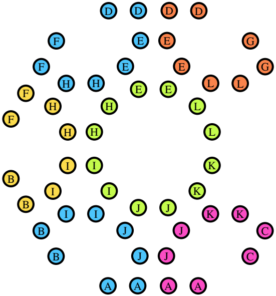
Sommets : orbites $\langle 1, 2 \rangle$



Faces : orbites $\langle 0, 1 \rangle$

[1] Damiand et al. 2014.

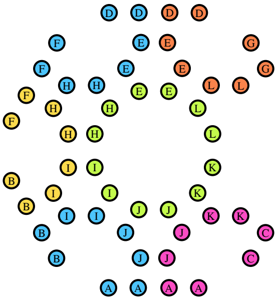
Plongements (géométrie)

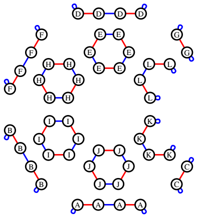


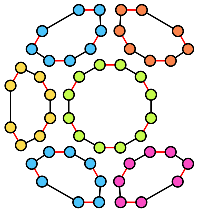
Légende : 0, 1, 2.

Plongements (géométrie)

Plongement : fonction $\pi : \langle o_\pi \rangle \to \tau_\pi$
avec τ un type de données abstrait





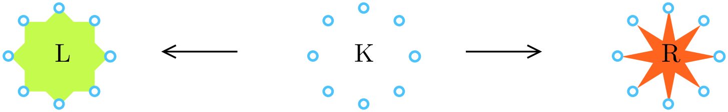


Légende : 0, 1, 2. *position* : $\langle 1, 2 \rangle \to$ `Point3` *color* : $\langle 0, 1 \rangle \to$ `ColorRGB`

Réécriture de graphes

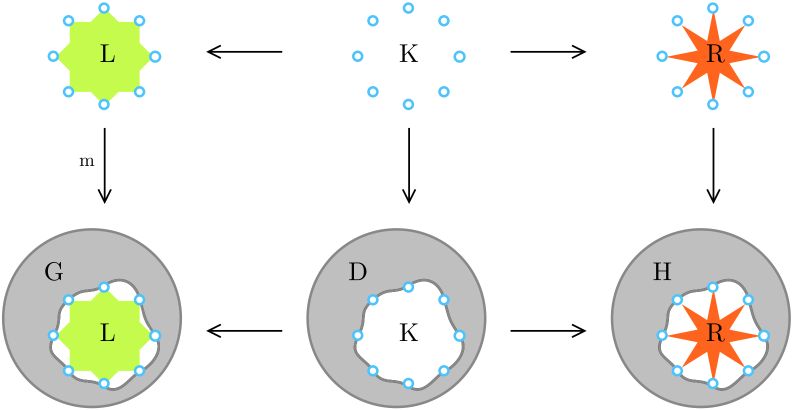
- Comment formaliser les transformations d'objets ?

Règles de transformation de graphes[1]



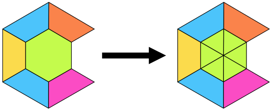
[1] Rozenberg 1997; Ehrig et al. 2006; Heckel et al. 2020.

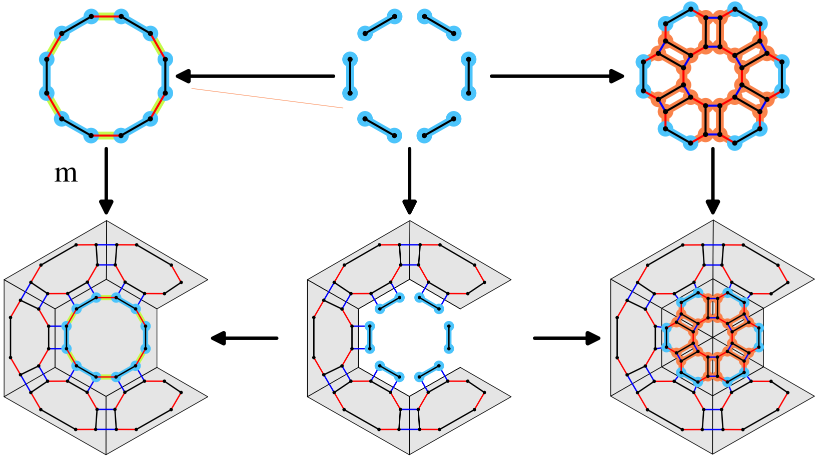
Règles de transformation de graphes[1]



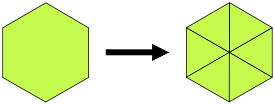
[1] Rozenberg 1997; Ehrig et al. 2006; Heckel et al. 2020.

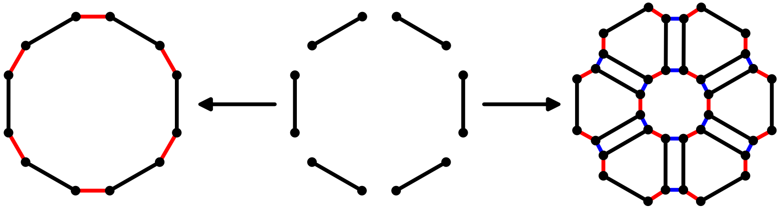
Réécriture topologique de G-cartes



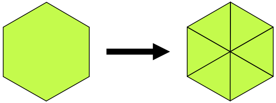


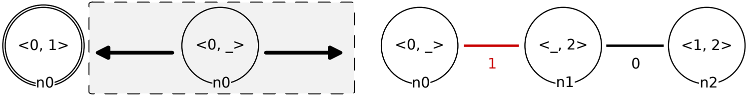
Réécriture d'orbites



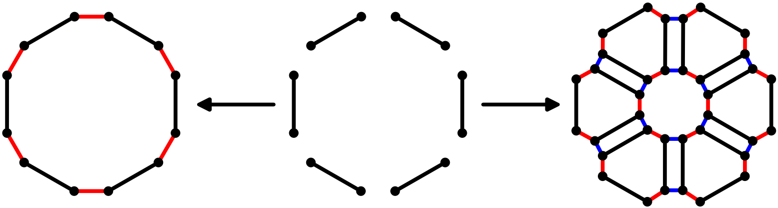


Réécriture d’orbites



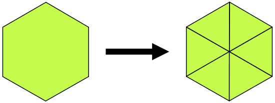


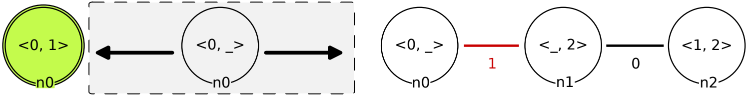
Implicitement
calculée



Règle
instanciée

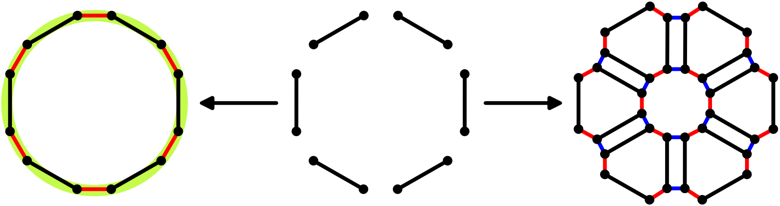
Réécriture d’orbites





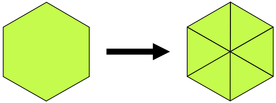
Implicitement
calculée

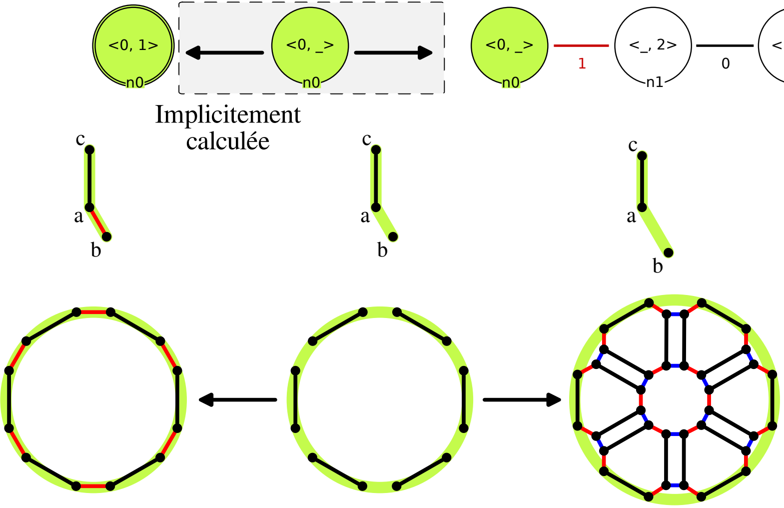
Local



Règle
instanciée

Réécriture d'orbites

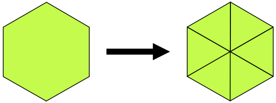


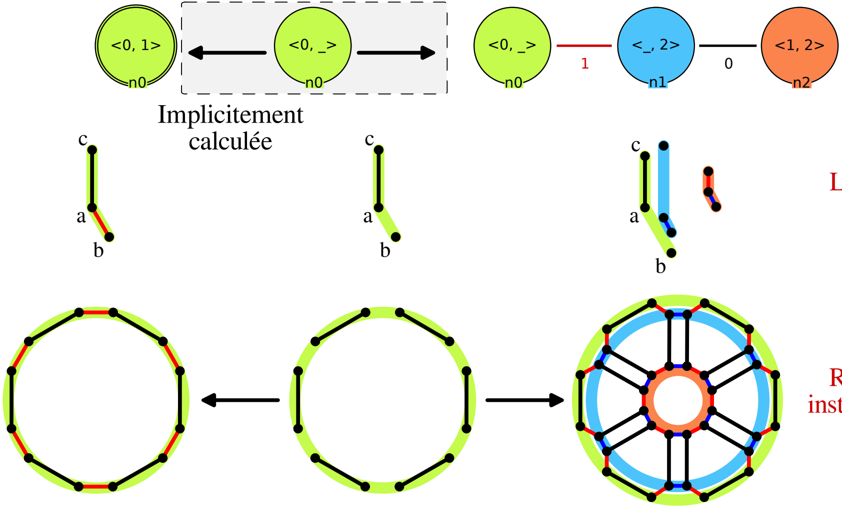


Local

Règle instanciée

Réécriture d'orbites

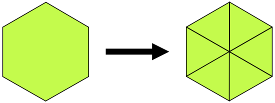


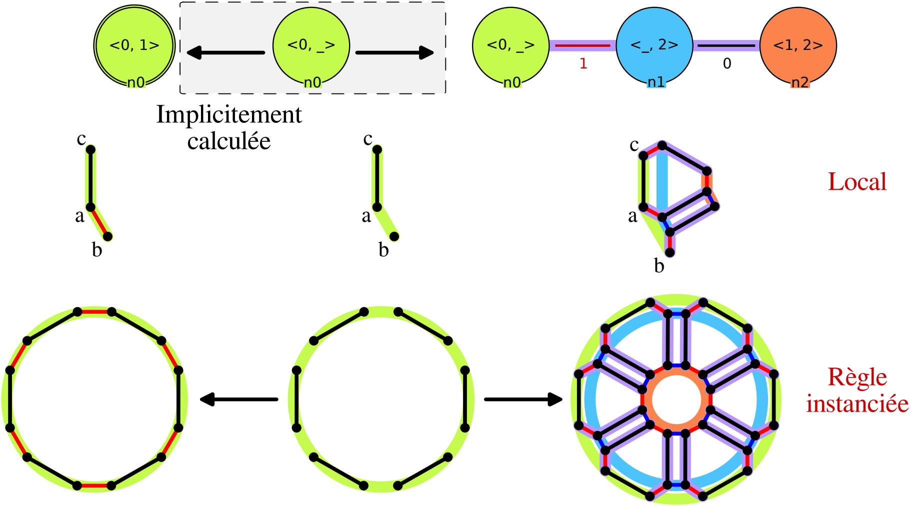


Local

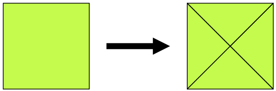
Règle instanciée

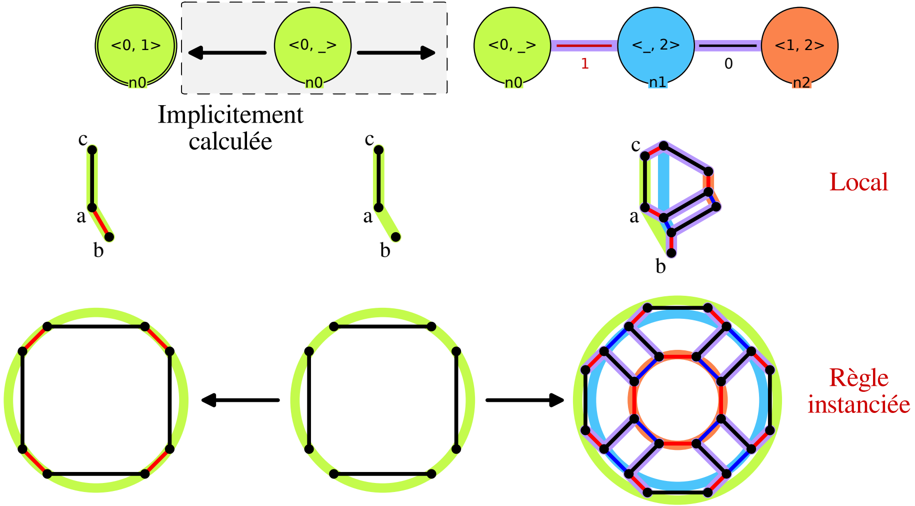
Réécriture d'orbites



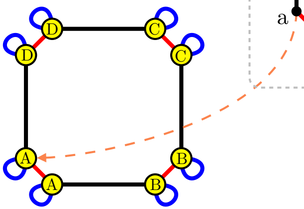


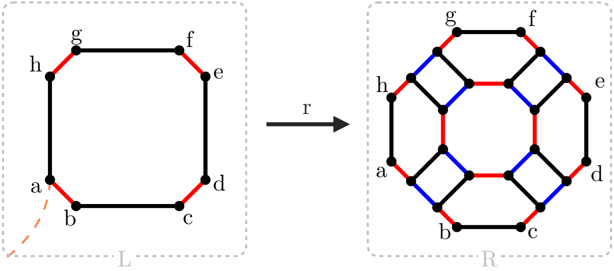
Réécriture d'orbites





Expressions de plongement[1]





Trois familles d'expressions :

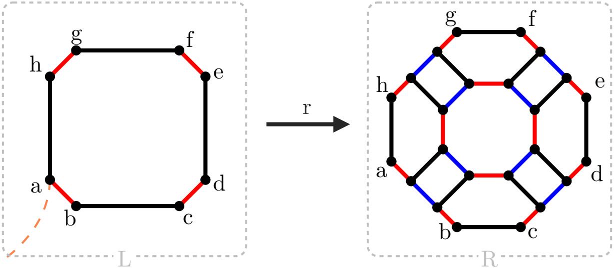
- Accès aux valeurs

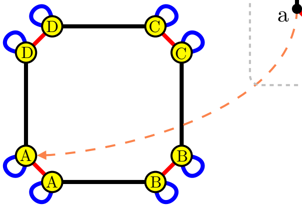
  a.couleur =

  a.position = A

[1]Bellet et al. 2017; Arnould et al. 2022.

Expressions de plongement[1]





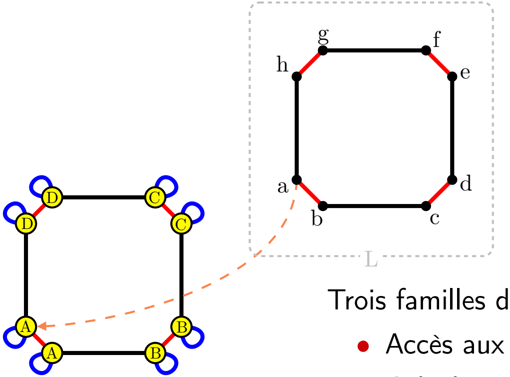
Trois familles d'expressions :

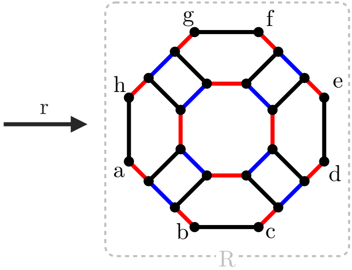
- Accès aux valeurs
- Calculs

  $\bullet + \bullet = \bullet$ (RGB)

[1]Bellet et al. 2017; Arnould et al. 2022.

Expressions de plongement[1]



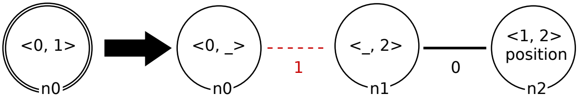


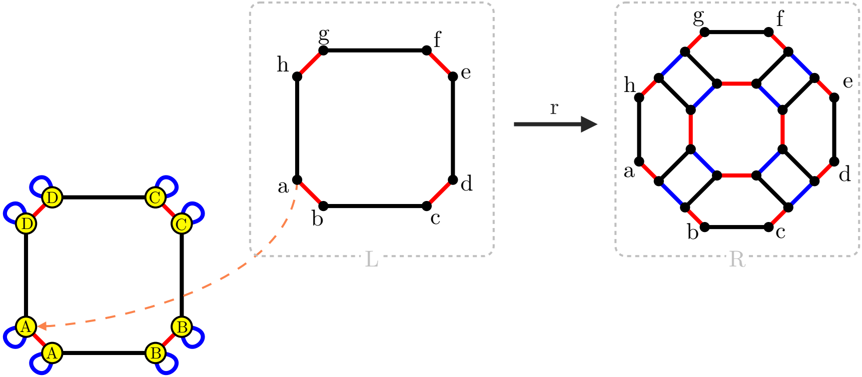
Trois familles d'expressions :

- Accès aux valeurs
- Calculs
- Parcours sur la G-carte
  a@0.position $= D$
  $\text{position}_{\langle 0,1 \rangle}(a) = \{A, B, C, D\}$

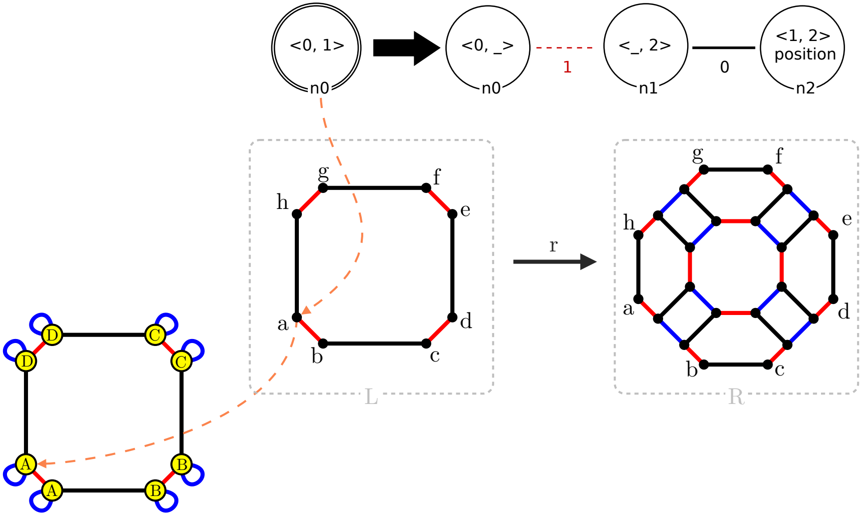
[1]Bellet et al. 2017; Arnould et al. 2022.

Extension aux schémas

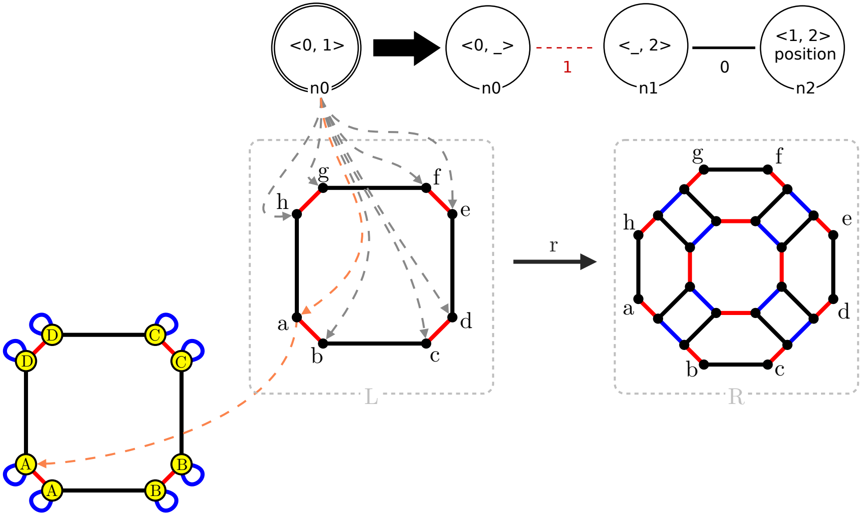




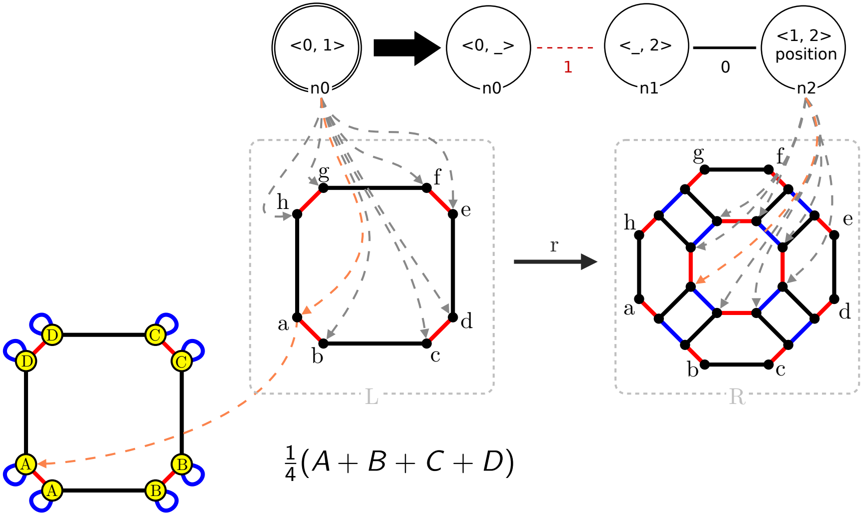
Extension aux schémas



Extension aux schémas

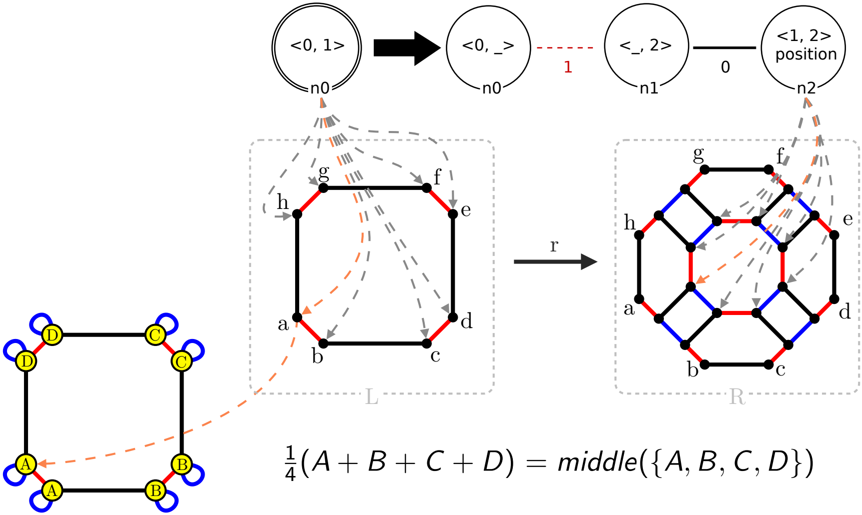


Extension aux schémas



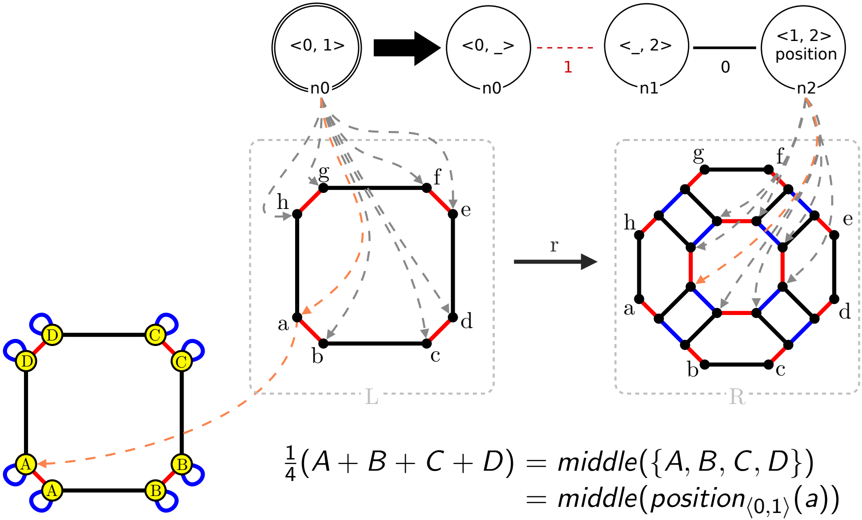
$\frac{1}{4}(A + B + C + D)$

Extension aux schémas



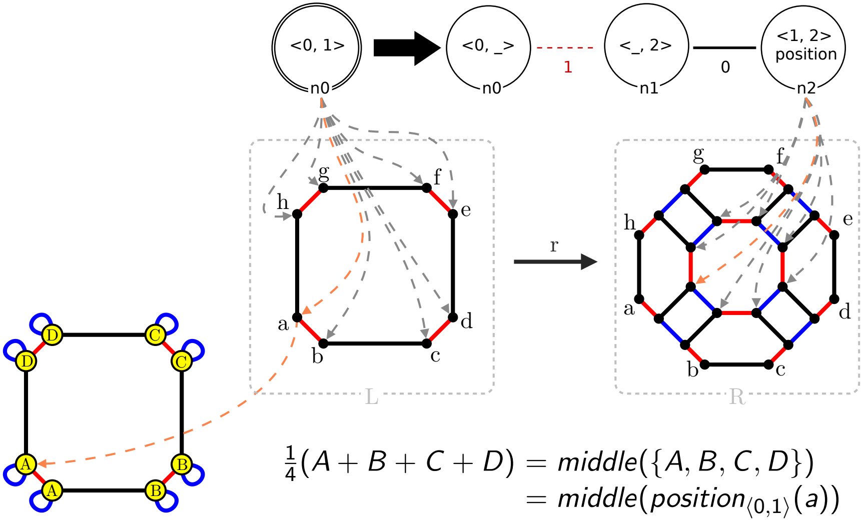
$$\frac{1}{4}(A + B + C + D) = middle(\{A, B, C, D\})$$

Extension aux schémas



$$\frac{1}{4}(A + B + C + D) = middle(\{A, B, C, D\})$$
$$= middle(position_{\langle 0,1\rangle}(a))$$

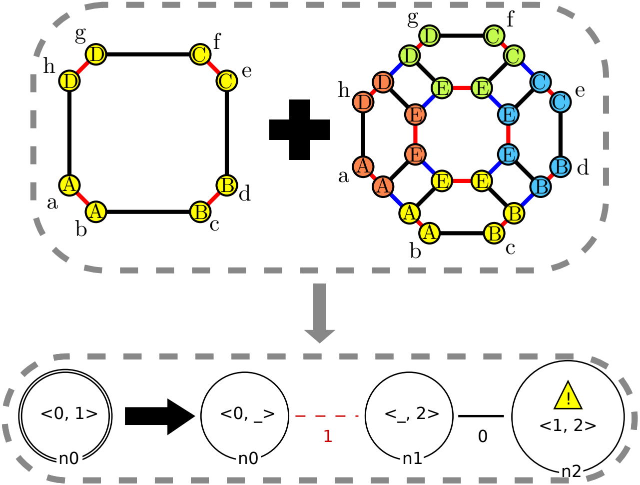
Extension aux schémas



$$\frac{1}{4}(A + B + C + D) = middle(\{A, B, C, D\})$$
$$= middle(position_{\langle 0,1 \rangle}(a))$$
$$= middle(position_{\langle 0,1 \rangle}(n0))$$

Équations symboliques

- Comment retrouver les expressions de calcul géométrique ?



Il manque les calculs des plongements

Espace de recherche (inférence des positions)

Hypothèse : combinaisons affines de points

Espace de recherche (inférence des positions)

Hypothèse : combinaisons affines de points

Pour chaque sommet de la cible, on cherche une position exprimée comme

$$p = \sum_{i=0}^{k} a_i p_i + t$$

avec

p : position cible (connu)

Espace de recherche (inférence des positions)

Hypothèse : combinaisons affines de points

Pour chaque sommet de la cible, on cherche une position exprimée comme

$$p = \sum_{i=0}^{k} a_i p_i + t$$

avec

p	:	position cible	(connu)
p_i	:	position du sommet d'origine i	(connu)

Espace de recherche (inférence des positions)

Hypothèse : combinaisons affines de points

Pour chaque sommet de la cible, on cherche une position exprimée comme

$$p = \sum_{i=0}^{k} a_i p_i + t$$

avec

p :	position cible	(connu)
p_i :	position du sommet d'origine i	(connu)
a_i :	coefficient	(inconnu)

Espace de recherche (inférence des positions)

Hypothèse : combinaisons affines de points

Pour chaque sommet de la cible, on cherche une position exprimée comme

$$p = \sum_{i=0}^{k} a_i p_i + t$$

avec

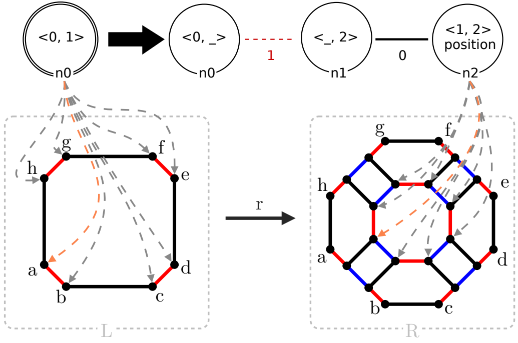
p	:	position cible	(connu)
p_i	:	position du sommet d'origine i	(connu)
a_i	:	coefficient	(inconnu)
t	:	translation intrinsèque	(inconnu)

Besoin d'abstraction sur les schémas

$(a_i)_i$ tels que $p = \sum_{i=0}^{k} a_i p_i + t$

Besoin d'abstraction sur les schémas

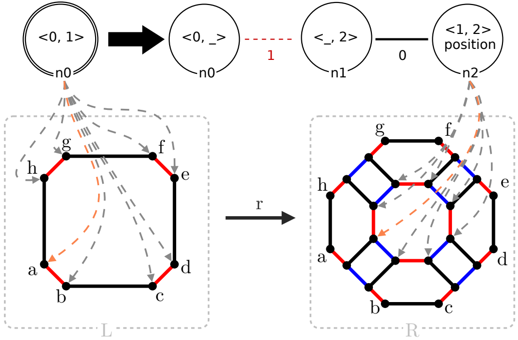
$(a_i)_i$ tels que $p = \sum_{i=0}^{k} a_i p_i + t$



Abstraction topologique dans les schémas

Besoin d'abstraction sur les schémas

$(a_i)_i$ tels que $p = \sum_{i=0}^{k} a_i p_i + t$

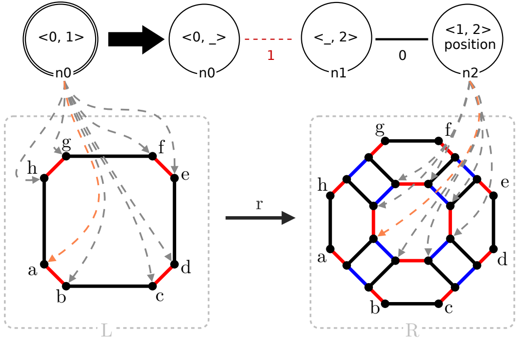


Abstraction topologique dans les schémas

Problème : partage d'expressions entre les brins

Besoin d'abstraction sur les schémas

$(a_i)_i$ tels que $p = \sum_{i=0}^{k} a_i p_i + t$



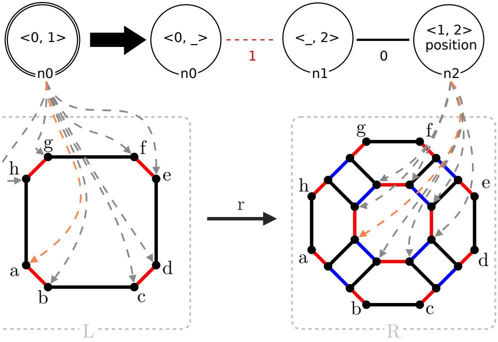
Abstraction topologique dans les schémas

Problème : partage d'expressions entre les brins

Solution : Exploiter la topologie

Besoin d'abstraction sur les schémas

$(a_i)_i$ tels que $p = \sum_{i=0}^{k} a_i p_i + t$



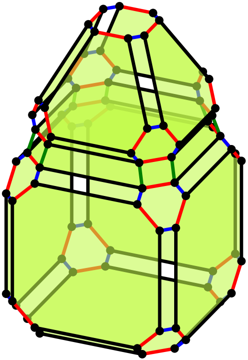
Abstraction topologique dans les schémas

Problème : partage d'expressions entre les brins

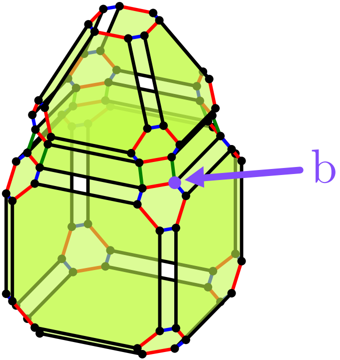
Solution : Exploiter la topologie

Points d'intérêt

Points d'intérêt



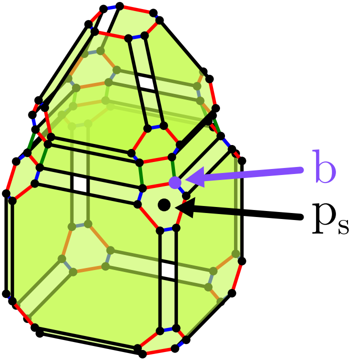
Points d'intérêt



Points d'intérêt

avec

- p_s : position du sommet

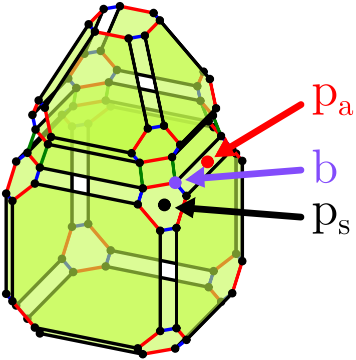


$$p_s = middle(position_{\langle\rangle}(b))$$

Points d'intérêt

avec

- p_s : position du sommet
- p_a : milieu de l'arête

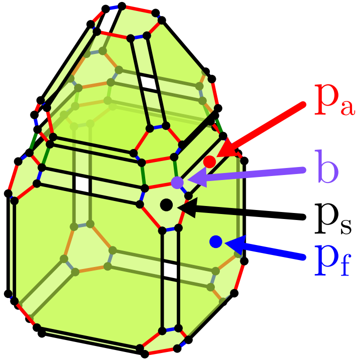


$$p_a = middle(position_{\langle 0 \rangle}(b))$$

Points d'intérêt

avec

- p_s : position du sommet
- p_a : milieu de l'arête
- p_f : milieu de la face

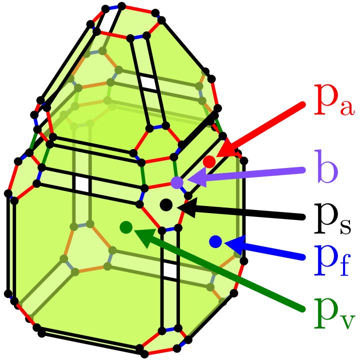


$$p_f = middle(position_{\langle 0,1 \rangle}(b))$$

Points d'intérêt

avec

- p_s : position du sommet
- p_a : milieu de l'arête
- p_f : milieu de la face
- p_v : milieu du volume

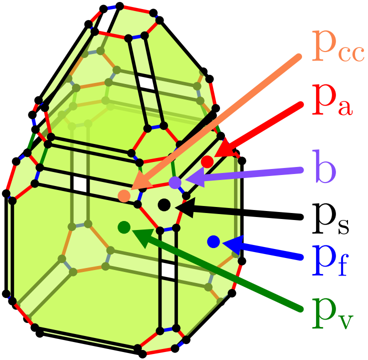


$$p_v = middle(position_{\langle 0,1,2 \rangle}(b))$$

Points d'intérêt

avec

- p_s : position du sommet
- p_a : milieu de l'arête
- p_f : milieu de la face
- p_v : milieu du volume
- p_{cc} : milieu de la composante connexe

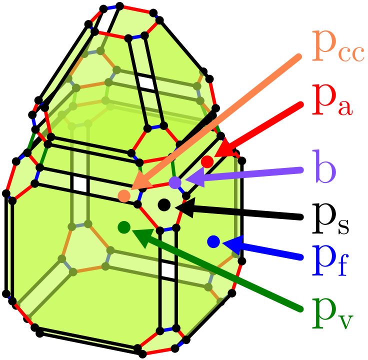


$$p_{cc} = middle(position_{\langle 0,1,2,3\rangle}(b))$$

Points d'intérêt

avec

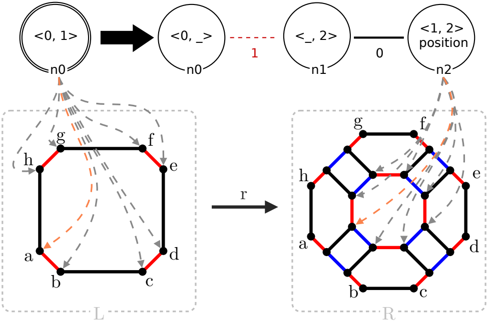
- p_s : position du sommet
- p_a : milieu de l'arête
- p_f : milieu de la face
- p_v : milieu du volume
- p_{cc} : milieu de la composante connexe



Via les points d'intérêt, le système se réécrit

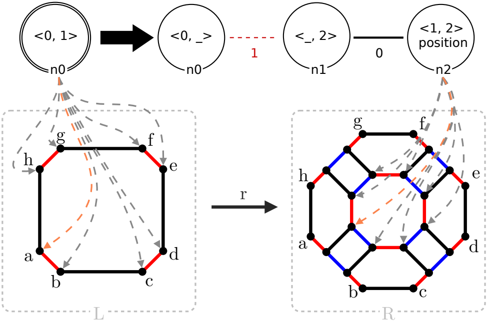
$$p = a_s p_s + a_a p_a + a_f p_f + a_v p_v + a_{cc} p_{cc} + t$$

Exemple de résolution



Position de *n*2 dépend uniquement de *n*0

Exemple de résolution

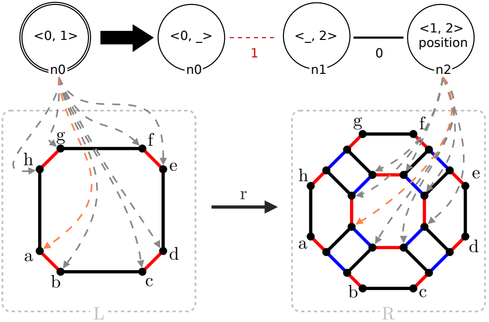


Position de $n2$ dépend uniquement de $n0$

Equation symbolique

$$n2.position = a_s n0.p_s + a_a \color{red}{n0.p_a} + a_f \color{blue}{n0.p_f} + a_v \color{green}{n0.p_v} + a_{cc} \color{orange}{n0.p_{cc}} + t$$

Exemple de résolution



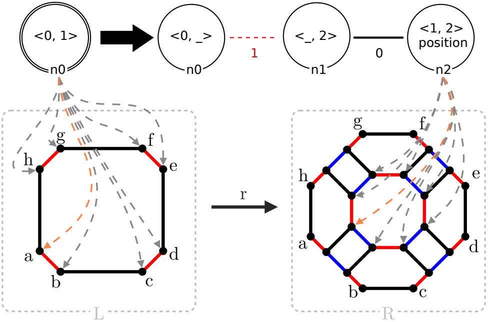
Position de $n2$ dépend uniquement de $n0$

- Une équation par brin de $n0$ (8 brins)

Equation symbolique

$$n2.position = a_s n0.p_s + a_a n0.p_a + a_f n0.p_f + a_v n0.p_v + a_{cc} n0.p_{cc} + t$$

Exemple de résolution



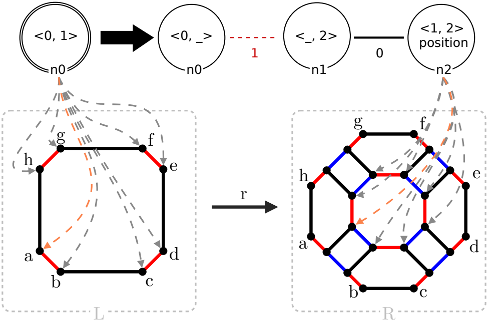
Position de $n2$ dépend uniquement de $n0$

- Une équation par brin de $n0$ (8 brins)
- Séparation sur x, y, z

Equation symbolique

$$n2.position = a_s n0.p_s + a_a n0.p_a + a_f n0.p_f + a_v n0.p_v + a_{cc} n0.p_{cc} + t$$

Exemple de résolution



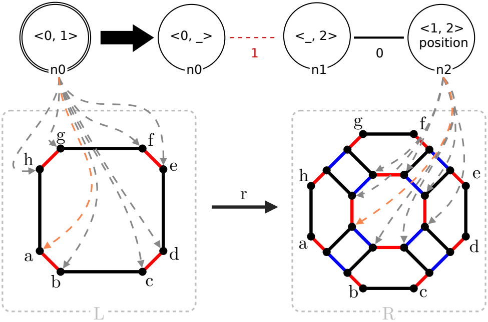
Position de $n2$ dépend uniquement de $n0$

- Une équation par brin de $n0$ (8 brins)
- Séparation sur x, y, z
- Système de 24 équations avec 8 variables

Equation symbolique

$$n2.position = a_s n0.p_s + a_a n0.p_a + a_f n0.p_f + a_v n0.p_v + a_{cc} n0.p_{cc} + t$$

Exemple de résolution



Position de $n2$ dépend uniquement de $n0$

- Une équation par brin de $n0$ (8 brins)
- Séparation sur x, y, z
- Système de 24 équations avec 8 variables

Equation symbolique

$$n2.position = a_s n0.p_s + a_a \textcolor{red}{n0.p_a} + a_f \textcolor{blue}{n0.p_f} + a_v \textcolor{green}{n0.p_v} + a_{cc} \textcolor{orange}{n0.p_{cc}} + t$$

Résolution par CSP (Z3, OR-Tools)

Résolution

Equation symbolique

$$n2.position = a_s n0.p_s + a_a n0.p_a + a_f n0.p_f + a_v n0.p_v + a_{cc} n0.p_{cc} + t$$

Résolution

Equation symbolique

$$n2.position = a_s n0.p_s + a_a n0.p_a + a_f n0.p_f + a_v n0.p_v + a_{cc} n0.p_{cc} + t$$

Système généré

$$\begin{cases}
(0.5;0.5) = w_v * (0;0) + w_e * (0.5;0) + w_f * (0.5;0.5) + w_s * (0.5;0.5) + w_{cc} * (0.5;0.5) + (tx;ty) \\
(0.5;0.5) = w_v * (1;0) + w_e * (0.5;0) + w_f * (0.5;0.5) + w_s * (0.5;0.5) + w_{cc} * (0.5;0.5) + (tx;ty) \\
(0.5;0.5) = w_v * (1;0) + w_e * (1;0.5) + w_f * (0.5;0.5) + w_s * (0.5;0.5) + w_{cc} * (0.5;0.5) + (tx;ty) \\
(0.5;0.5) = w_v * (1;1) + w_e * (1;0.5) + w_f * (0.5;0.5) + w_s * (0.5;0.5) + w_{cc} * (0.5;0.5) + (tx;ty) \\
\vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots
\end{cases}$$

Résolution

Equation symbolique

$$n2.position = a_s n0.p_s + a_a n0.p_a + a_f n0.p_f + a_v n0.p_v + a_{cc} n0.p_{cc} + t$$

Système généré

$$\begin{cases} (0.5;0.5) = w_v * (0;0) + w_e * (0.5;0) + w_f * (0.5;0.5) + w_s * (0.5;0.5) + w_{cc} * (0.5;0.5) + (tx;ty) \\ (0.5;0.5) = w_v * (1;0) + w_e * (0.5;0) + w_f * (0.5;0.5) + w_s * (0.5;0.5) + w_{cc} * (0.5;0.5) + (tx;ty) \\ (0.5;0.5) = w_v * (1;0) + w_e * (1;0.5) + w_f * (0.5;0.5) + w_s * (0.5;0.5) + w_{cc} * (0.5;0.5) + (tx;ty) \\ (0.5;0.5) = w_v * (1;1) + w_e * (1;0.5) + w_f * (0.5;0.5) + w_s * (0.5;0.5) + w_{cc} * (0.5;0.5) + (tx;ty) \\ \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \end{cases}$$

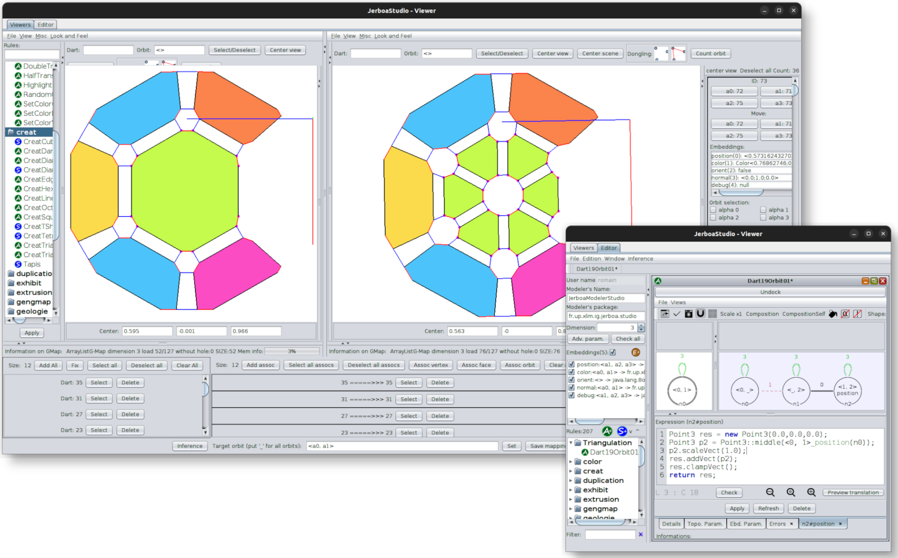
Solution trouvée

- $a_s = 0.0$
- $a_a = 0.0$
- $a_f = 1.0$
- $a_v = 0.0$
- $a_{cc} = 0.0$
- $t = (0.0, 0.0)$

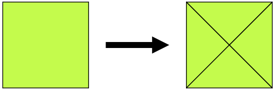
JerboaStudio et applications

- Mise en œuvre dans Jerboa

JerboaStudio

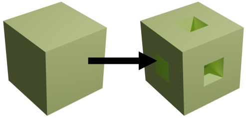


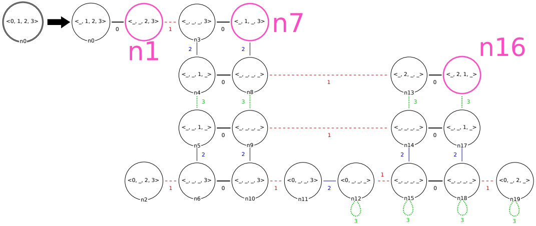
Code généré pour la triangulation

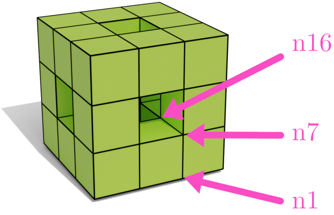


```
// translation nulle
Point3 res = new Point3(0.0,0.0,0.0);

// face
Point3 p2 = Point3::middle(<0,1>_position(n0));

// poids
p2.scale(1.0);

// ajout au resultat
res.addVect(p2);

// retour de la valeur
return res;
```

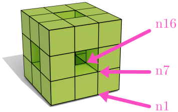
Éponge de Menger







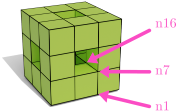
Éponge de Menger



Nœud $n1$

```
Point3 res = new Point3(0.0,0.0,0.0);
Point3 p0 = Point3::middle(<>_position(n0));
p0.scale(0.333333134651184);
res.addVect(p0);
Point3 p1 = Point3::middle(<0>_position(n0));
p1.scale(0.666666865348816);
res.addVect(p1);
return res;
```

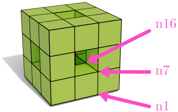
Éponge de Menger



Nœud *n7*

```
Point3 res = new Point3(0.0,0.0,0.0);
Point3 p0 = Point3::middle(<>_position(n0));
p0.scale(0.333333134651184);
res.addVect(p0);
Point3 p2 = Point3::middle(<0,1>_position(n0));
p2.scale(0.666666865348816);
res.addVect(p2);
return res;
```

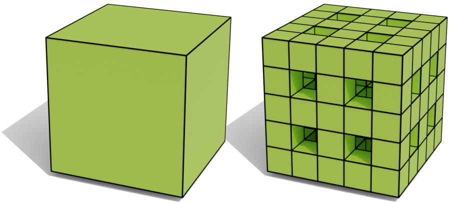
Éponge de Menger



Nœud $n16$

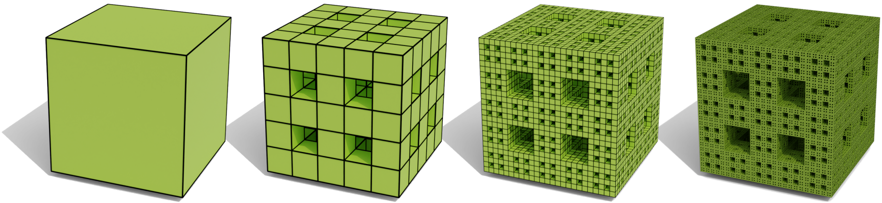
```
Point3 res = new Point3(0.0,0.0,0.0);
Point3 p0 = Point3::middle(<>_position(n0));
p0.scale(0.3333333134651184);
res.addVect(p0);
Point3 p3 = Point3::middle(<0,1,2>_position(n0));
p3.scale(0.6666666865348816);
res.addVect(p3);
return res;
```

Polycube de Menger $(2, 2, 2)$[1]



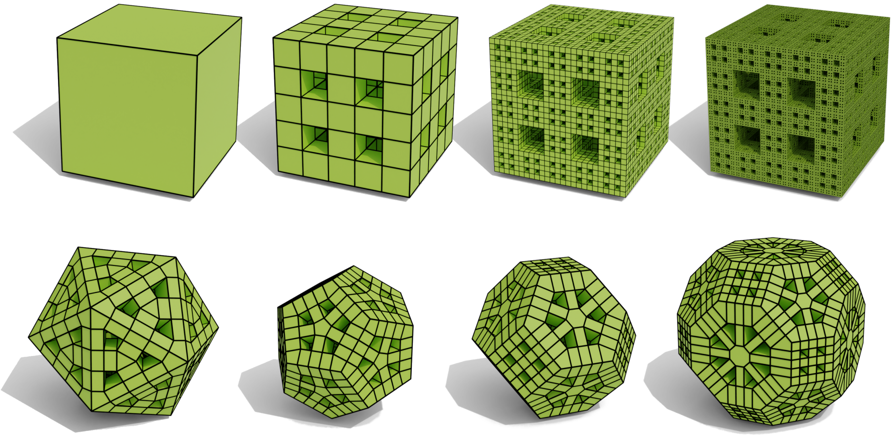
[1]Richaume et al. 2019.

Polycube de Menger $(2, 2, 2)$[1]



[1] Richaume et al. 2019.

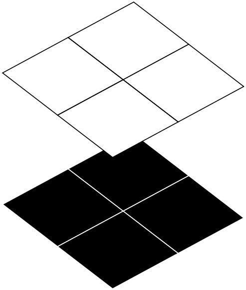
Polycube de Menger $(2, 2, 2)$[1]



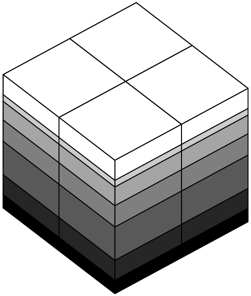
[1] Richaume et al. 2019.

Exemple inspiré de la géologie

Avant



Après



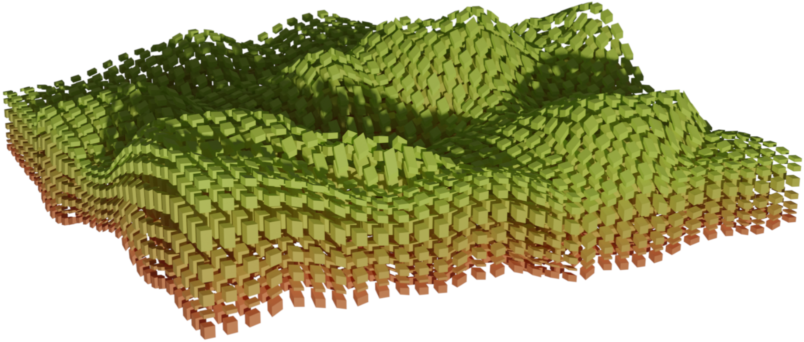
Positions et couleurs

Exemple inspiré de la géologie

Avant

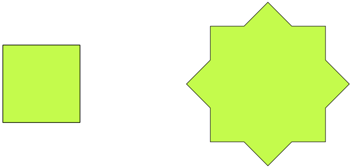
Exemple inspiré de la géologie

Après

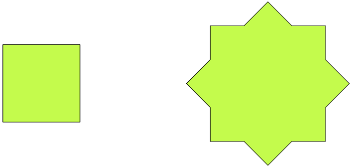


Limites

Flocon de von Koch's généré par L-systèmes

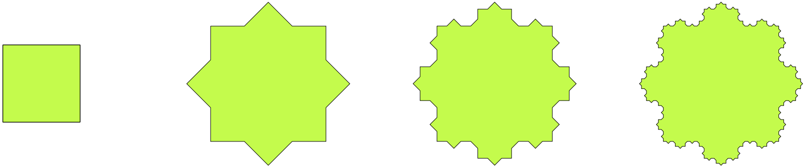


Inféré

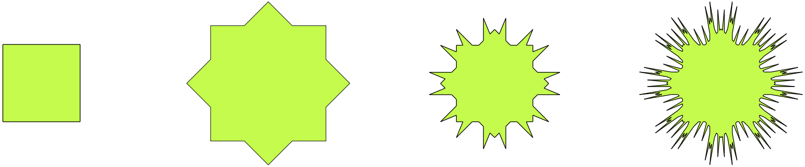


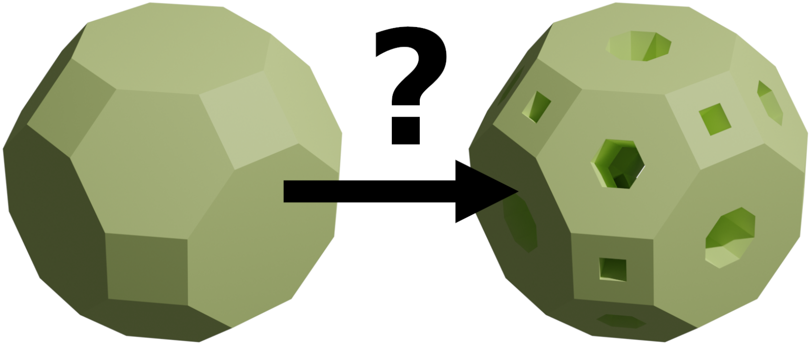
Limites

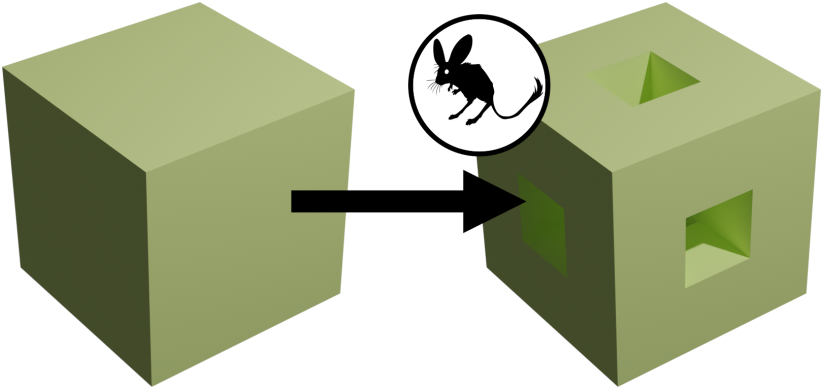
Flocon de von Koch's généré par L-systèmes



Inféré







References I

Arnould, Agnès et al. (2022). "Preserving consistency in geometric modeling with graph transformations". In: Mathematical Structures in Computer Science. DOI: 10.1017/S0960129522000226.

Bellet, Thomas et al. (2017). "Geometric Modeling: Consistency Preservation Using Two-Layered Variable Substitutions". In: Graph Transformation (ICGT 2017). Ed. by Juan de Lara et al. Vol. 10373. Lecture Notes in Computer Science. Cham: Springer International Publishing, pp. 36–53. ISBN: 978-3-319-61470-0. DOI: 10.1007/978-3-319-61470-0_3.

Damiand, Guillaume et al. (Sept. 19, 2014). Combinatorial Maps: Efficient Data Structures for Computer Graphics and Image Processing. CRC Press. 407 pp. ISBN: 978-1-4822-0652-4.

References II

Ehrig, Hartmut et al. (2006). Fundamentals of Algebraic Graph Transformation. Monographs in Theoretical Computer Science. An EATCS Series. Berlin Heidelberg: Springer-Verlag. ISBN: 978-3-540-31187-4. DOI: 10.1007/3-540-31188-2.

Heckel, Reiko et al. (2020). Graph Transformation for Software Engineers: With Applications to Model-Based Development and Domain-Specific Language Engineering. Cham: Springer International Publishing. ISBN: 978-3-030-43915-6. DOI: 10.1007/978-3-030-43916-3.

References III

Richaume, Lydie et al. (2019). "Unfolding Level 1 Menger Polycubes of Arbitrary Size With Help of Outer Faces". In: Discrete Geometry for Computer Imagery. Ed. by Michel Couprie et al. Lecture Notes in Computer Science. Cham: Springer International Publishing, pp. 457–468. ISBN: 978-3-030-14085-4. DOI: 10.1007/978-3-030-14085-4_36.

Rozenberg, Grzegorz, ed. (Feb. 1, 1997). Handbook of Graph Grammars and Computing by Graph Transformation: Volume I. Foundations. Vol. Foundations. 1 vols. USA: World Scientific Publishing Co., Inc. 545 pp. ISBN: 978-981-02-2884-2.