

Raccourcissement de courbes et décomposition de surfaces

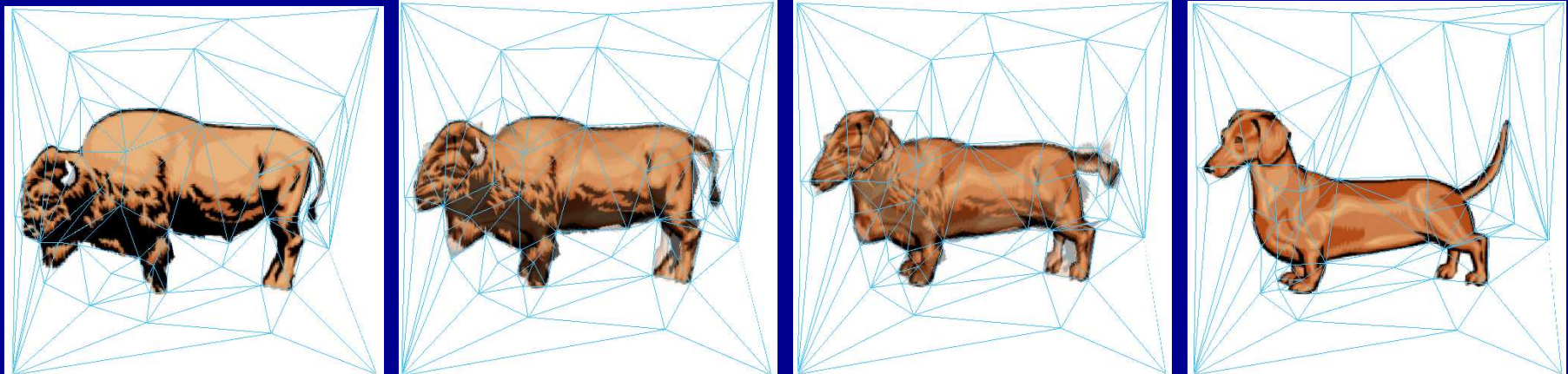
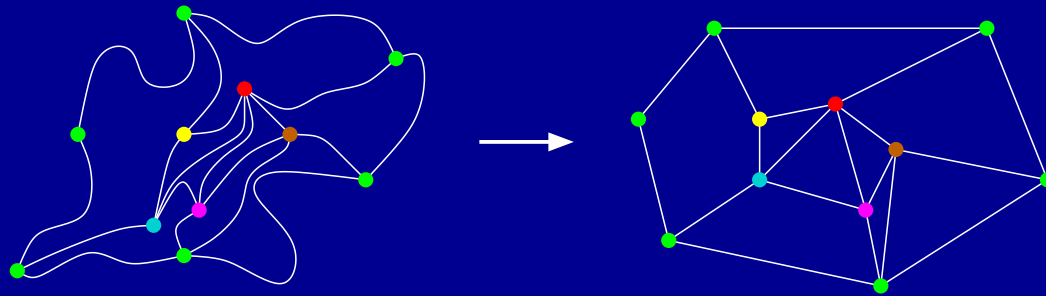
Éric Colin de Verdière

École normale supérieure (Paris) et Université Paris VII

Résultats de la thèse

- Théorème barycentrique de Tutte appliqué aux isotopies
avec Michel Pocchiola et Gert Vegter.

[1] *Computational Geometry : Theory and Applications, 2003.*



Résultats de la thèse

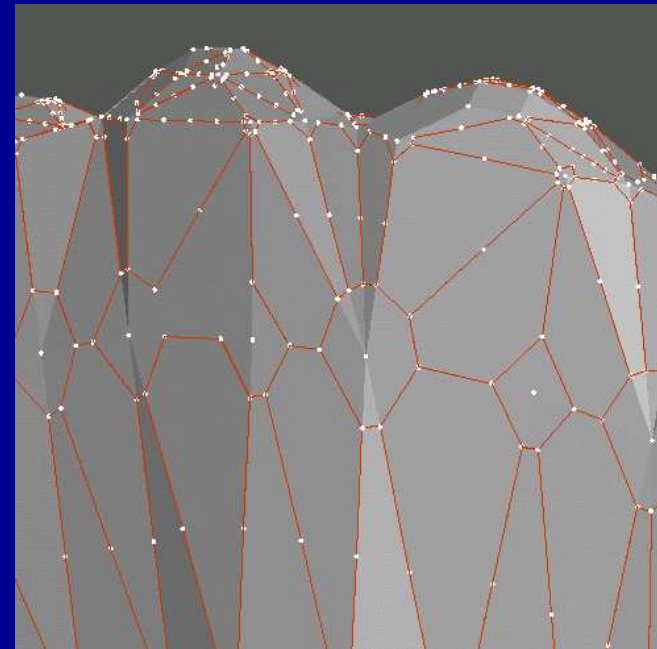
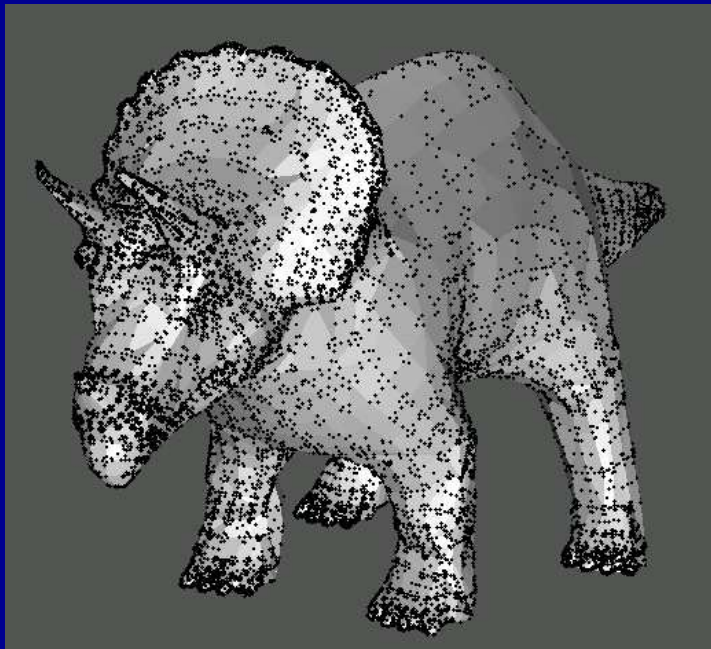
- Théorème barycentrique de Tutte appliqué aux isotopies
avec Michel Pocchiola et Gert Vegter.

[1] *Computational Geometry : Theory and Applications, 2003.*

- Triangulations de Delaunay conformes 3D
avec David Cohen-Steiner et Mariette Yvinec.

[2] *Symposium on Computational Geometry, 2002 ;*

[2'] *Computational Geometry : Theory and Applications, à paraître.*



Résultats de la thèse

- Théorème barycentrique de Tutte appliqué aux isotopies
avec Michel Pocchiola et Gert Vegter.

[1] *Computational Geometry : Theory and Applications, 2003.*

- Triangulations de Delaunay conformes 3D
avec David Cohen-Steiner et Mariette Yvinec.

[2] *Symposium on Computational Geometry, 2002 ;*

[2'] *Computational Geometry : Theory and Applications, à paraître.*

■ Optimisation de courbes sur des surfaces

- ◆ *avec Francis Lazarus :*

[3] *Symposium on Foundations of Computer Science, 2002 ;*

[4] *Symposium on Graph Drawing, 2003 ;*

- ◆ *dans cette thèse :*

- extension des résultats de [3] ;
- simplification des démonstrations de [3] ;
- simplification pour une étape de [4].

Décomposer, raccourcir, déformer

Topologie algorithmique des surfaces.

■ **Décomposer** topologiquement des surfaces ;

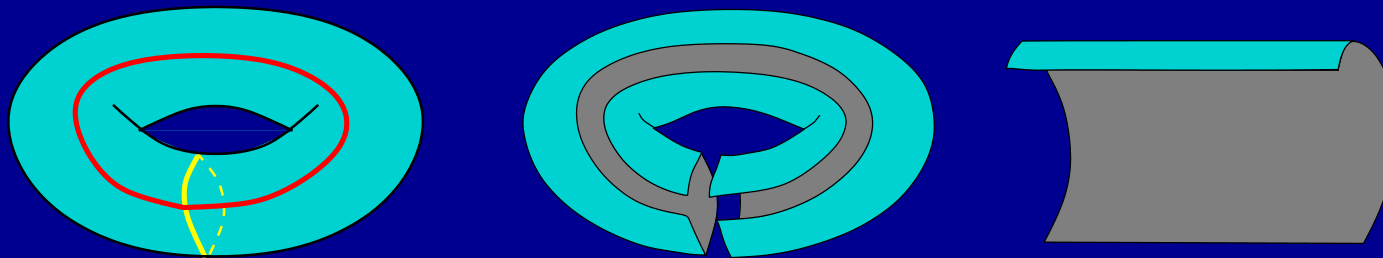
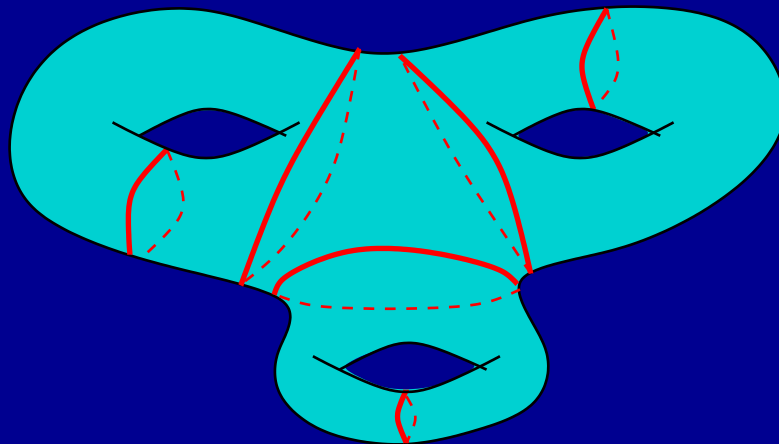
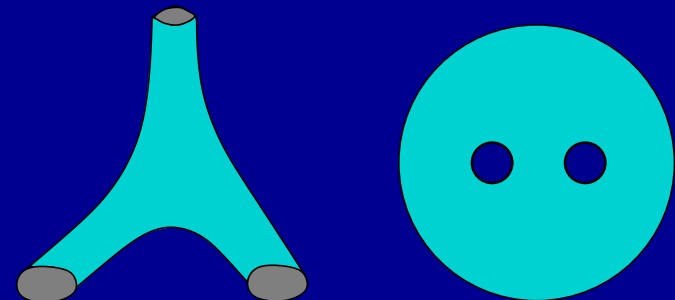


schéma polygonal



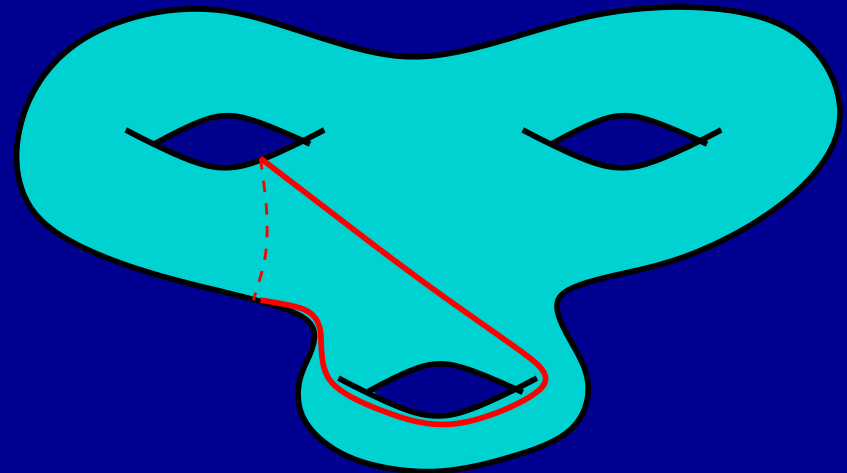
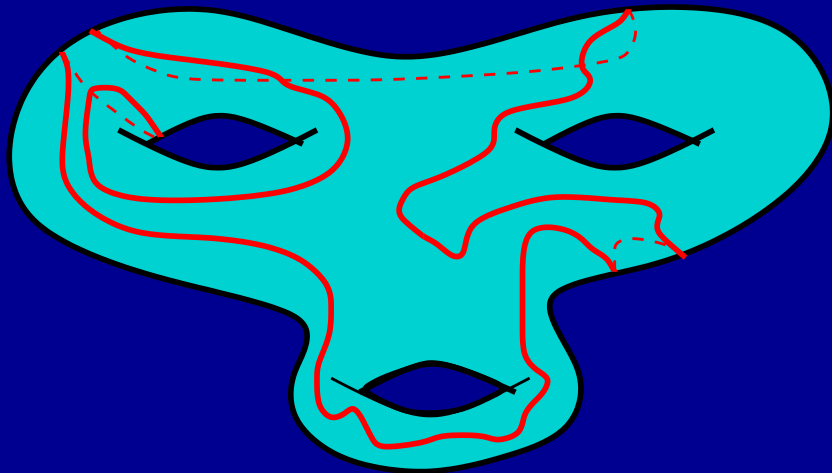
décomposition en pantalons



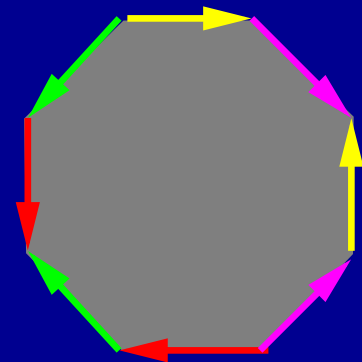
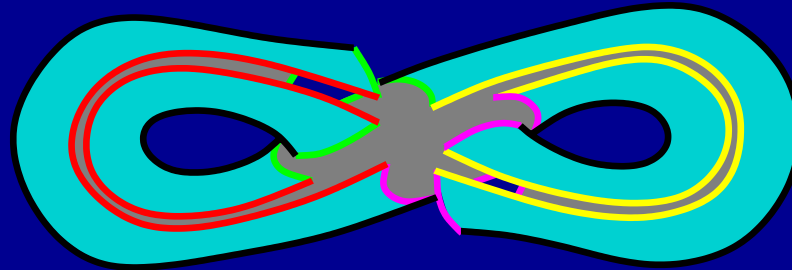
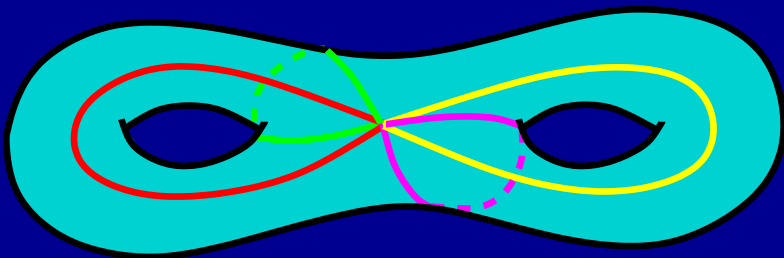
■ **raccourcir** des courbes sur des surfaces par **déformation**.

Motivations

- spécialisation du problème de calcul de plus courts chemins ;
- représentant « canonique » d'une classe d'homotopie ;

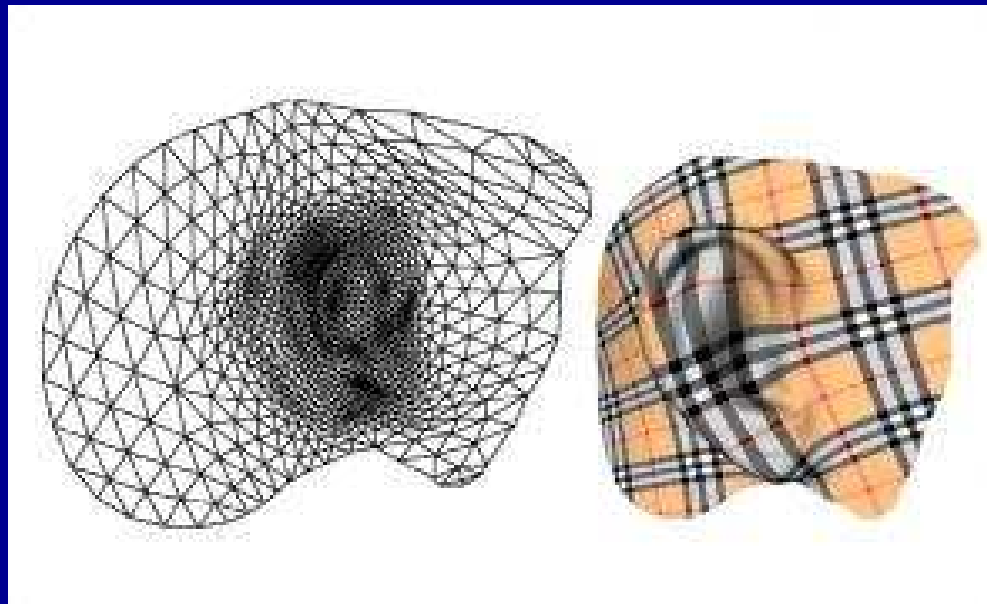


- calcul d'homéomorphismes.



Motivations en infographie

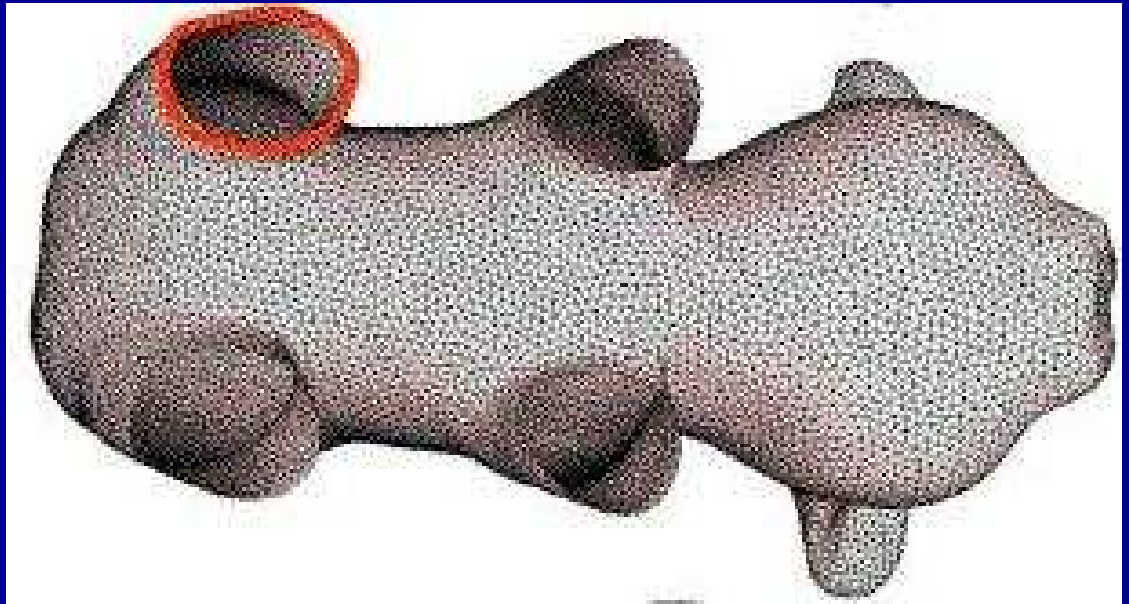
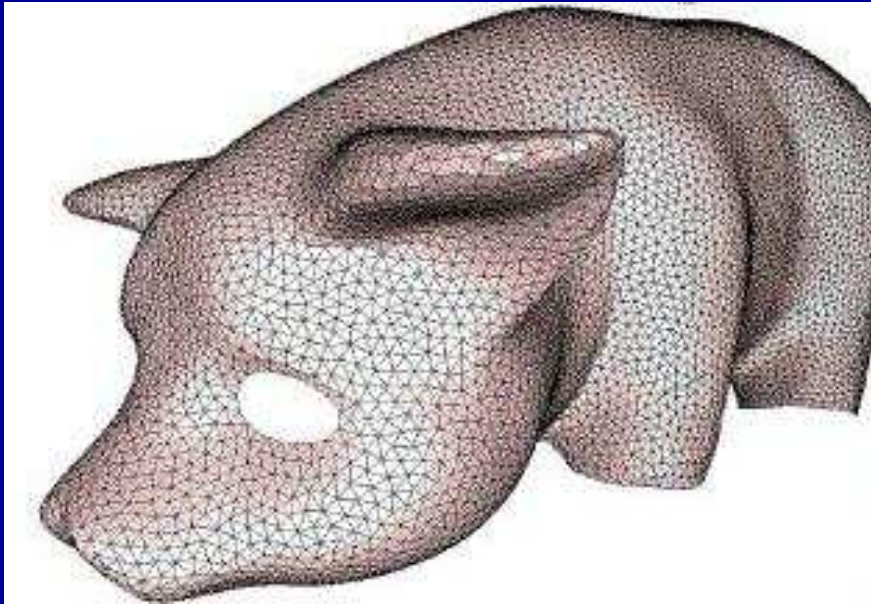
- CAO, modélisation géométrique, dessin 3D ;
- paramétrage (visualisation, plaquage de texture, maillage et calcul numérique, remaillage) ;
- analyse multi-échelle ;
- compression géométrique.



[Desbrun, Meyer, Alliez, 2002]

Motivations en infographie

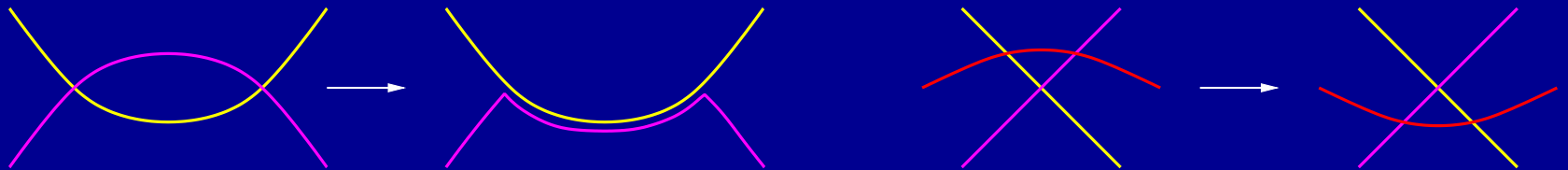
- CAO, modélisation géométrique, dessin 3D ;
- paramétrage (visualisation, plaquage de texture, maillage et calcul numérique, remaillage) ;
- analyse multi-échelle ;
- compression géométrique.



[Alliez, Colin de Verdière, Devillers, Isenburg, 2003]

Travaux antérieurs

- Topologie algébrique classique (début du XX^e siècle) ;
- ♦ calcul efficace de schémas polygonaux canoniques
[Vegter, Yap, 1990], [Lazarus, Pocchiola, Vegter, Verroust, 2001] ;
- ♦ calcul du plus court schéma polygonal sur le graphe sommets-arêtes d'une surface polyédrale
[Erickson, Har-Peled, 2002] ;
- ♦ simplicité d'un plus court cycle homotope à un cycle simple donné [Hass, Scott, 1982] ;
- ♦ décroisement de courbes, mouvements de Reidemeister
[Hass, Scott, 1985], [de Graaf, Schrijver, 1997] ;



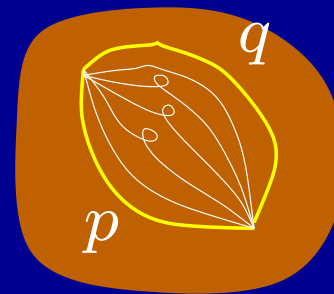
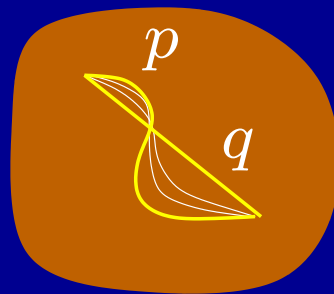
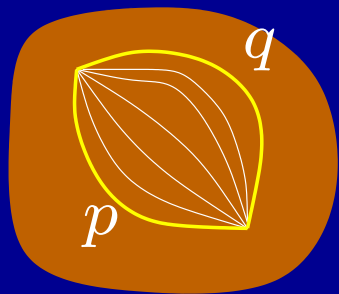
- calcul de plus courtes courbes homotopes dans le plan
[Hershberger, Snoeyink, 1994], [Efrat, Kobourov, Lubiw 2002], [Bespamyatnikh, 2003].

Plongements de graphes et de cycles

Optimisation

- d'un plongement de graphe ;
- d'un plongement de cycles (famille de cycles simples et deux à deux disjoints).

Trouver le plus court plongement homotope (ou isotope) au plongement donné.

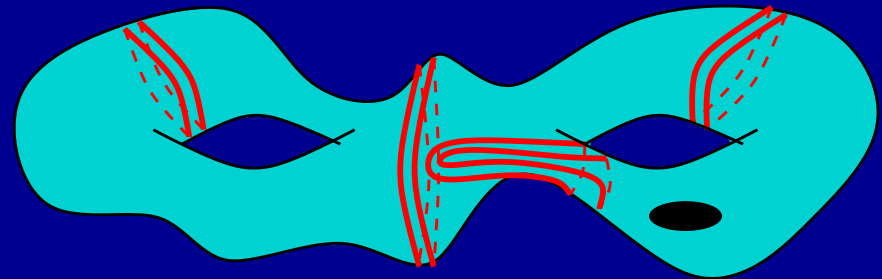
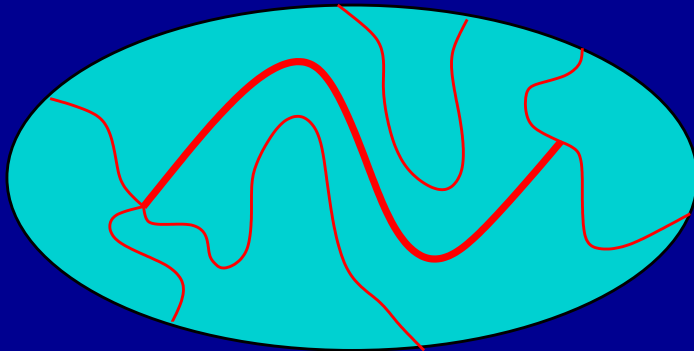


pas une isotopie

Aperçu

$S = (s_1, \dots, s_N)$ **système de découpage** décomposant la surface en surfaces topologiquement élémentaires.

■ **Algorithme** glouton, itératif, simple ;



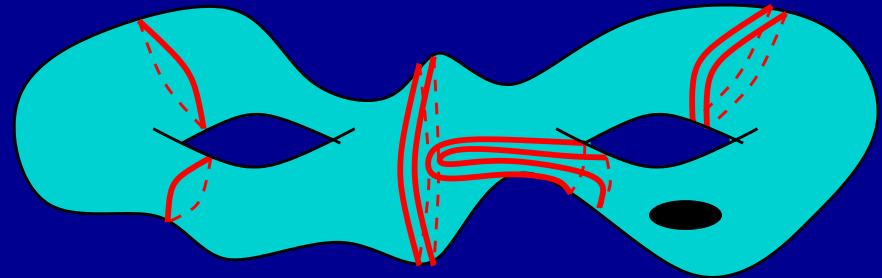
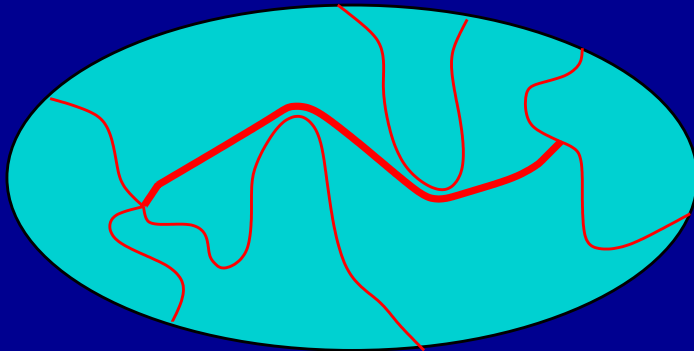
■ **résultat** : après itération, chaque courbe s_i est **individuellement** aussi courte que possible dans une classe d'homotopie ou d'isotopie ;

■ **conséquence** : complétion, optimisation, extraction.

Aperçu

$S = (s_1, \dots, s_N)$ **système de découpage** décomposant la surface en surfaces topologiquement élémentaires.

■ **Algorithme** glouton, itératif, simple ;



■ **résultat** : après itération, chaque courbe s_i est **individuellement** aussi courte que possible dans une classe d'homotopie ou d'isotopie ;

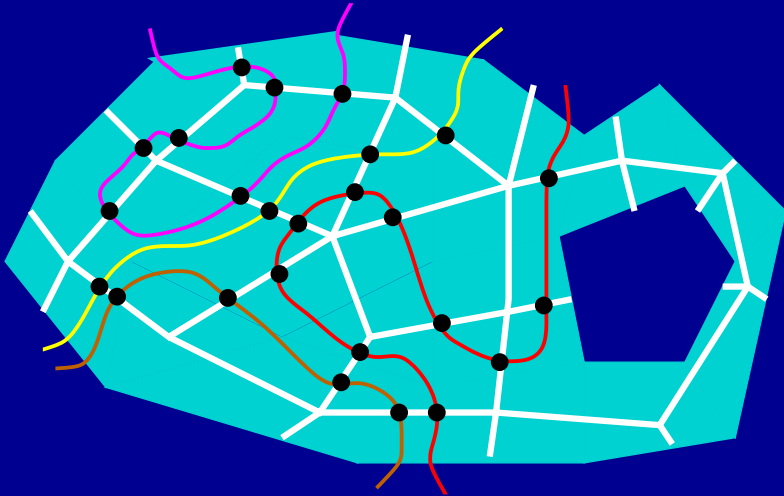
■ **conséquence** : complétion, optimisation, extraction.

Plan

1. Cadre de l'étude
2. Optimisation de plongements de graphe
3. Optimisation de plongements de cycles
4. Étude de la complexité

1. Cadre de l'étude

Notion de longueur

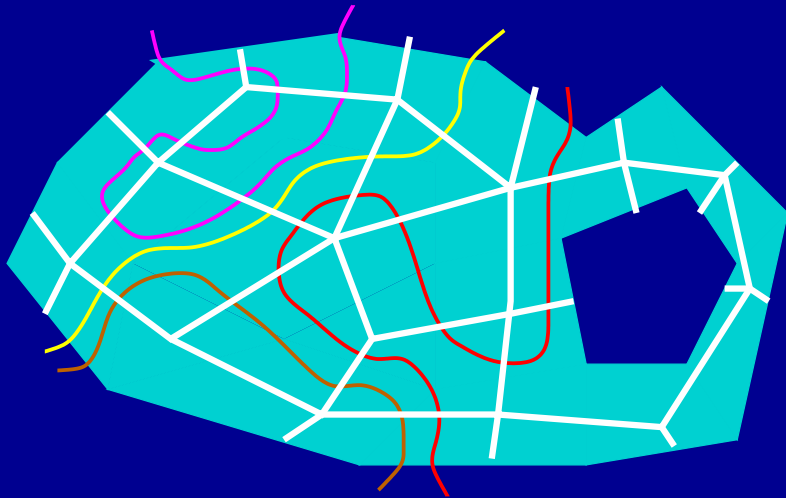


- Surface \mathcal{M} connexe, compacte, orientable, év. à bords ;
- graphe \mathcal{B} pondéré plongé sur \mathcal{M} ;
- longueur $|c|$ d'une courbe c : somme des poids des arêtes de \mathcal{B} croisées par c .

Courbes **régulières** par rapport à \mathcal{B} .

La position exacte des courbes à l'intérieur d'une face n'importe pas.

Notion de longueur

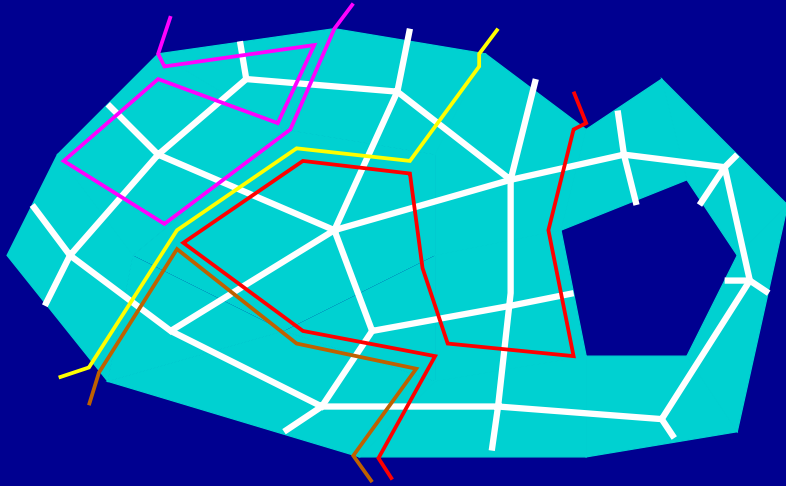


- Surface \mathcal{M} connexe, compacte, orientable, év. à bords ;
- graphe \mathcal{B} pondéré plongé sur \mathcal{M} ;
- longueur $|c|$ d'une courbe c : somme des poids des arêtes de \mathcal{B} croisées par c .

Courbes **régulières** par rapport à \mathcal{B} .

La position exacte des courbes à l'intérieur d'une face n'importe pas.

Notion de longueur

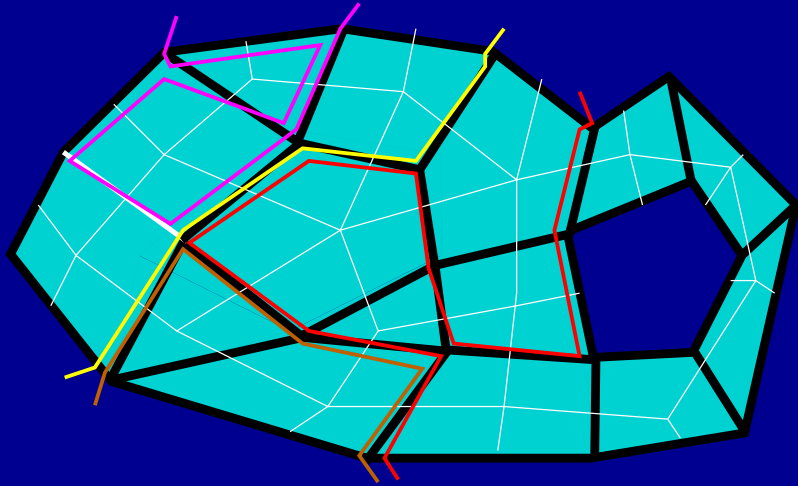


- Surface \mathcal{M} connexe, compacte, orientable, év. à bords ;
- graphe \mathcal{B} pondéré plongé sur \mathcal{M} ;
- longueur $|c|$ d'une courbe c : somme des poids des arêtes de \mathcal{B} croisées par c .

Courbes **régulières** par rapport à \mathcal{B} .

La position exacte des courbes à l'intérieur d'une face n'importe pas.

Notion de longueur

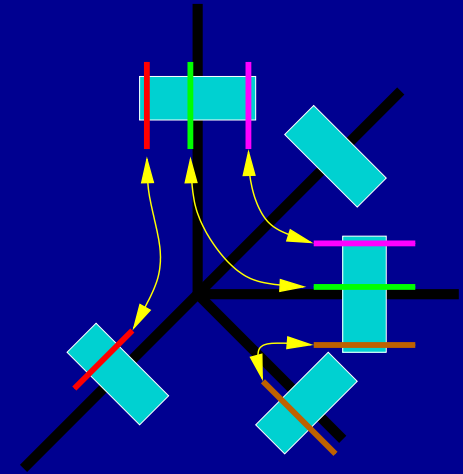
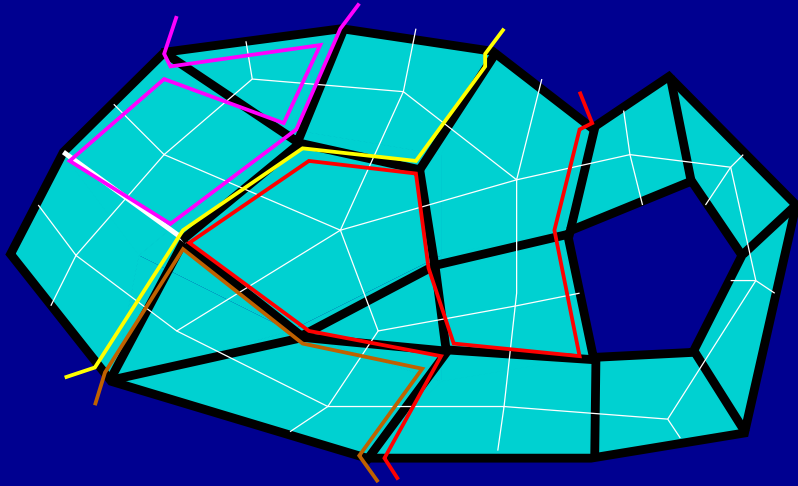


- Surface \mathcal{M} connexe, compacte, orientable, év. à bords ;
- graphe \mathcal{B} pondéré plongé sur \mathcal{M} ;
- longueur $|c|$ d'une courbe c : somme des poids des arêtes de \mathcal{B} croisées par c .

Courbes **régulières** par rapport à \mathcal{B} .

La position exacte des courbes à l'intérieur d'une face n'importe pas.

Notion de longueur

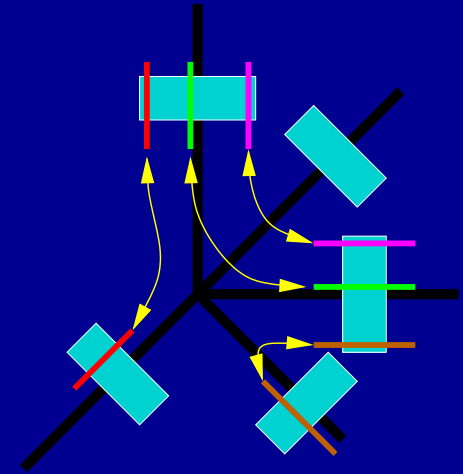
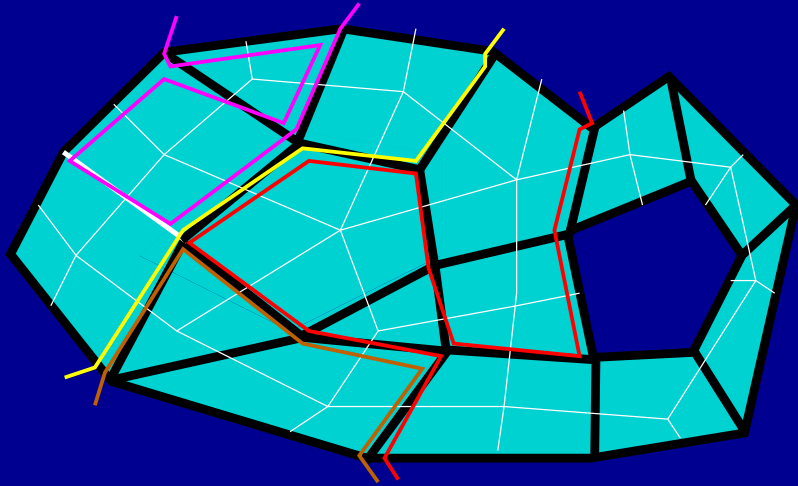


- Surface \mathcal{M} connexe, compacte, orientable, év. à bords ;
- graphe \mathcal{B} pondéré plongé sur \mathcal{M} ;
- longueur $|c|$ d'une courbe c : somme des poids des arêtes de \mathcal{B} croisées par c .

Courbes **régulières** par rapport à \mathcal{B} .

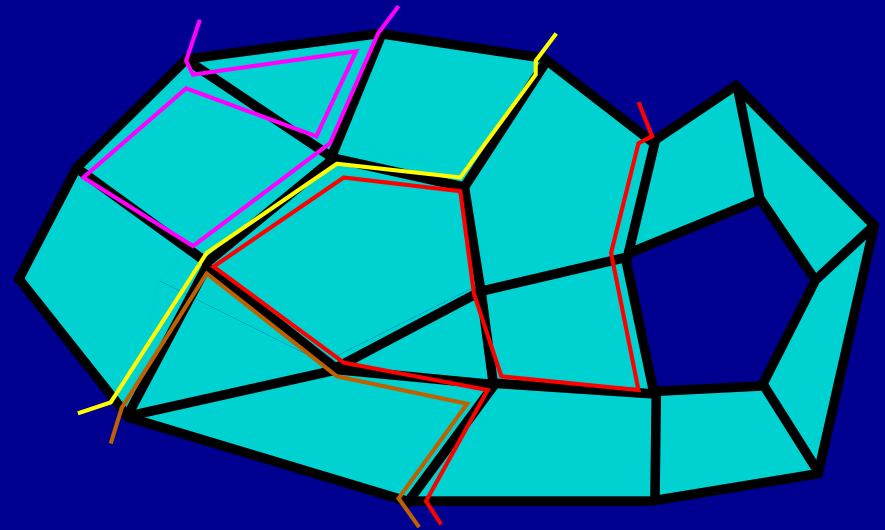
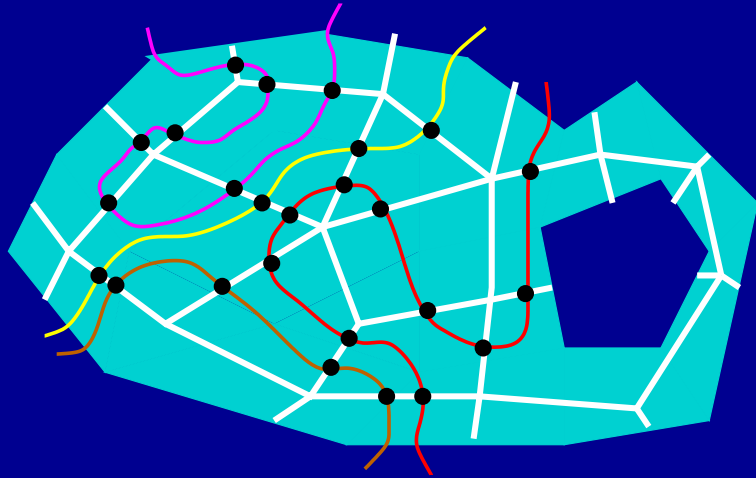
La position exacte des courbes à l'intérieur d'une face n'importe pas.

Notion de longueur



- \mathcal{M} polyédrale, de graphe sommets-arêtes \mathcal{N} pondéré ;
- courbes « longeant » le graphe \mathcal{N} sans se « croiser » ;
- structure de données : chemins « ordonnés » dans \mathcal{N} ;
- longueur : calculée dans le graphe \mathcal{N} .

Dualité



- cadre topologique (nombre de croisements avec le graphe \mathcal{B}) : démonstrations ;
- cadre algorithmique (opérations utilisant le graphe \mathcal{N}) : algorithmes et complexité.

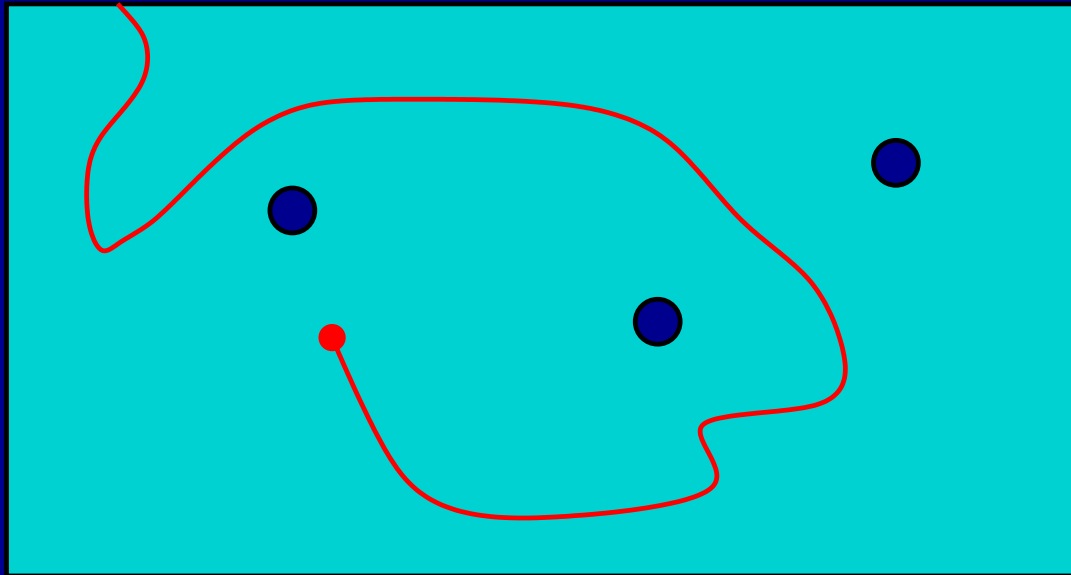
Intérêt :

- parfois souhaitable (remaillage, calcul approché) ;
- techniques susceptibles d'extensions.

2. Plongements de graphe

Complétion

$s = (s_1, \dots, s_n)$ plongement de graphe.



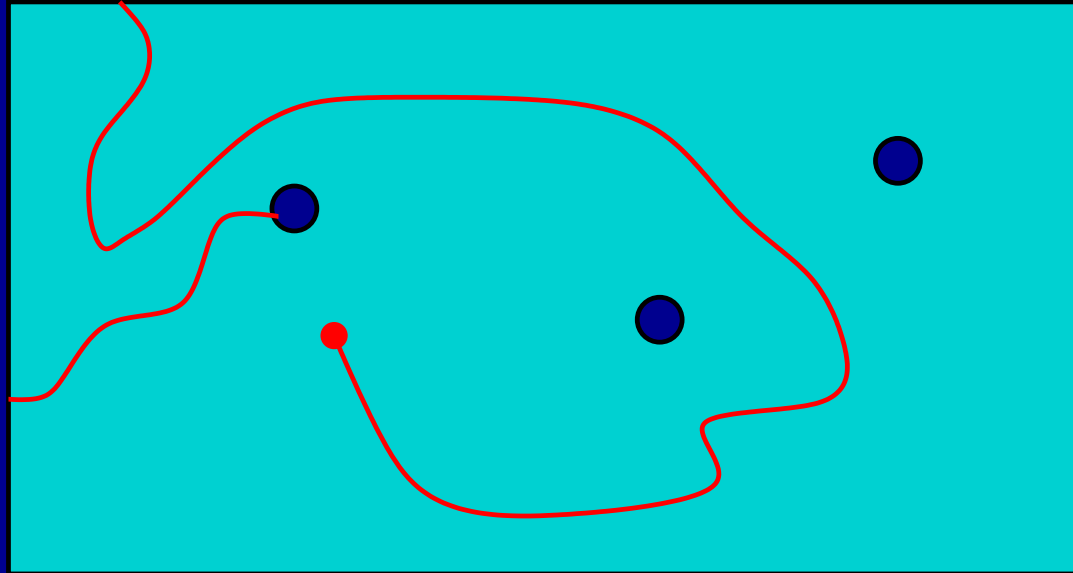
Complétion en un plongement de graphe $S = (s_1, \dots, s_N)$ tel que, pour tout i , la face de $S \setminus s_i$ contenant s_i est un disque, séparé en deux par s_i .

C'est un **système de découpage par graphe**.

Proposition. C'est possible sans ajouter de sommets à l'intérieur de \mathcal{M} (si $n \geq 1$).

Complétion

$s = (s_1, \dots, s_n)$ plongement de graphe.



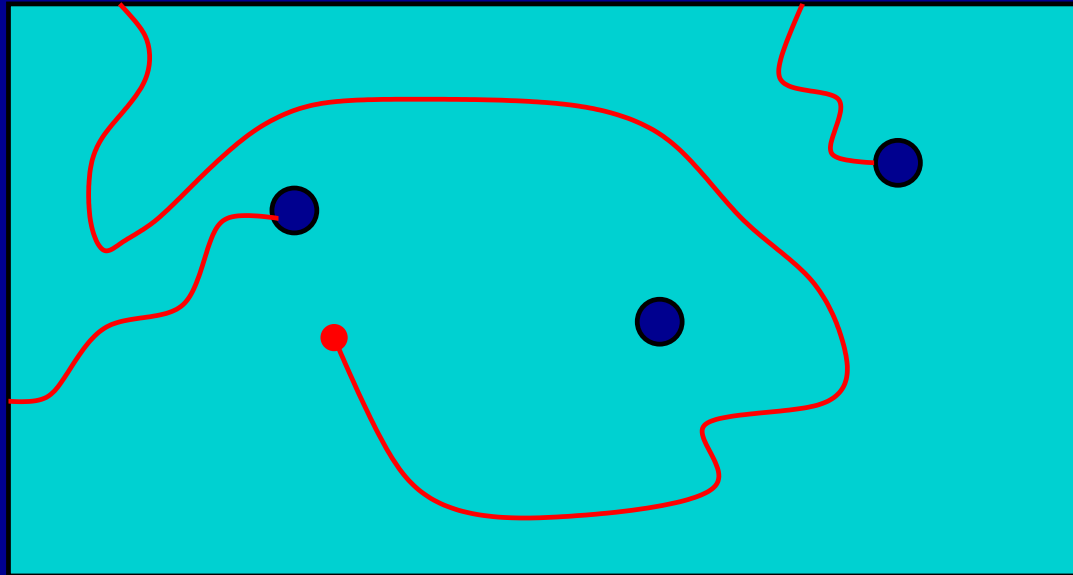
Complétion en un plongement de graphe $S = (s_1, \dots, s_N)$ tel que, pour tout i , la face de $S \setminus s_i$ contenant s_i est un disque, séparé en deux par s_i .

C'est un **système de découpage par graphe**.

Proposition. C'est possible sans ajouter de sommets à l'intérieur de \mathcal{M} (si $n \geq 1$).

Complétion

$s = (s_1, \dots, s_n)$ plongement de graphe.



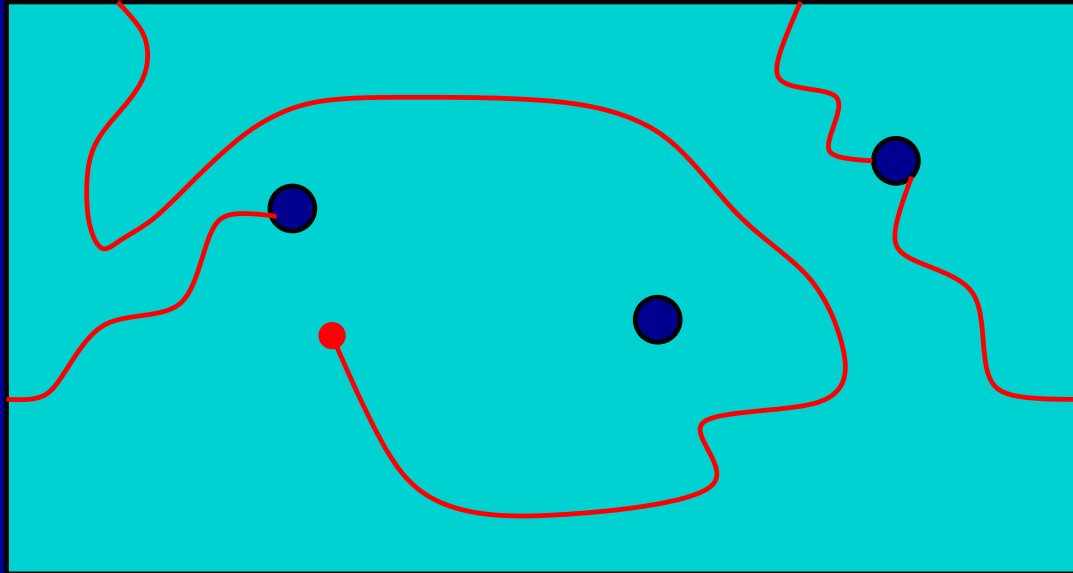
Complétion en un plongement de graphe $S = (s_1, \dots, s_N)$ tel que, pour tout i , la face de $S \setminus s_i$ contenant s_i est un disque, séparé en deux par s_i .

C'est un **système de découpage par graphe**.

Proposition. C'est possible sans ajouter de sommets à l'intérieur de \mathcal{M} (si $n \geq 1$).

Complétion

$s = (s_1, \dots, s_n)$ plongement de graphe.



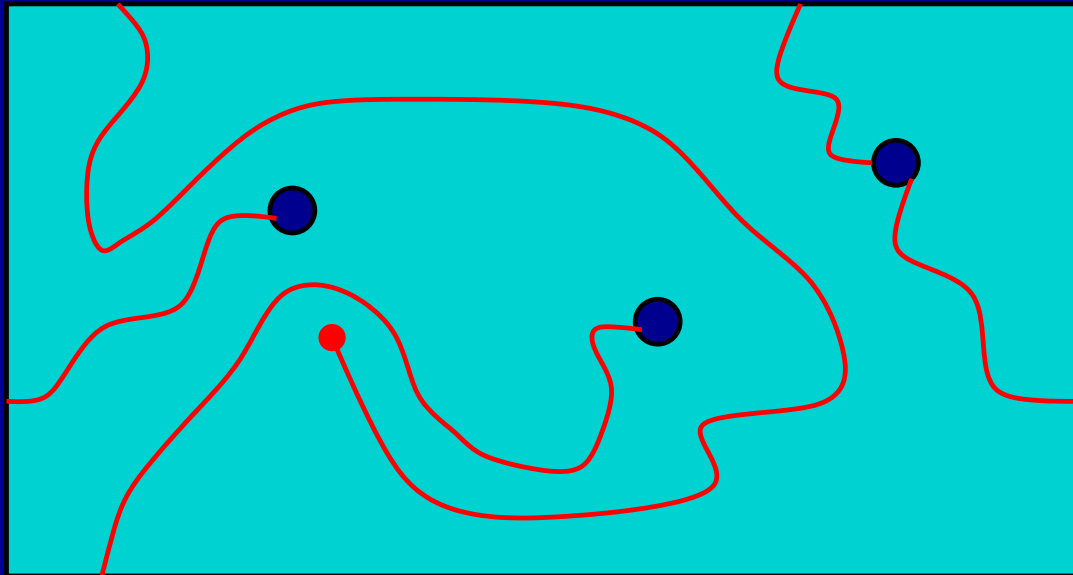
Complétion en un plongement de graphe $S = (s_1, \dots, s_N)$ tel que, pour tout i , la face de $S \setminus s_i$ contenant s_i est un disque, séparé en deux par s_i .

C'est un **système de découpage par graphe**.

Proposition. C'est possible sans ajouter de sommets à l'intérieur de \mathcal{M} (si $n \geq 1$).

Complétion

$s = (s_1, \dots, s_n)$ plongement de graphe.



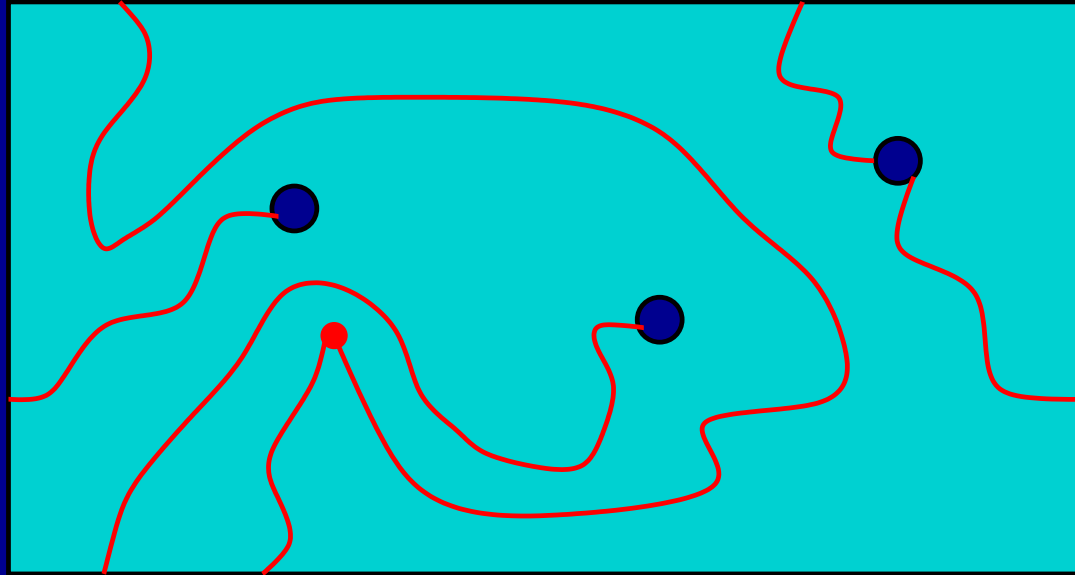
Complétion en un plongement de graphe $S = (s_1, \dots, s_N)$ tel que, pour tout i , la face de $S \setminus s_i$ contenant s_i est un disque, séparé en deux par s_i .

C'est un **système de découpage par graphe**.

Proposition. C'est possible sans ajouter de sommets à l'intérieur de \mathcal{M} (si $n \geq 1$).

Complétion

$s = (s_1, \dots, s_n)$ plongement de graphe.



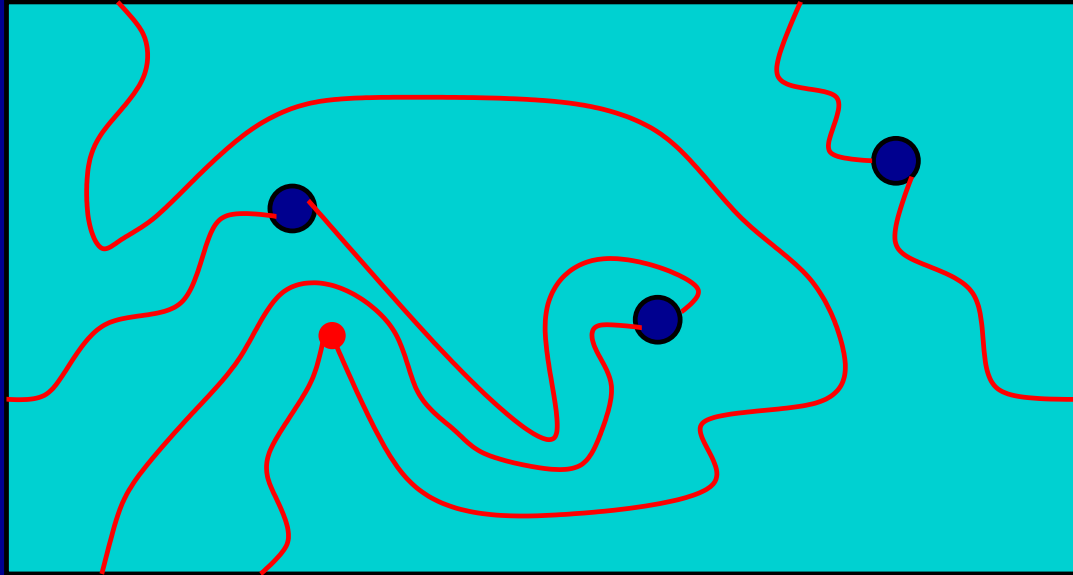
Complétion en un plongement de graphe $S = (s_1, \dots, s_N)$ tel que, pour tout i , la face de $S \setminus s_i$ contenant s_i est un disque, séparé en deux par s_i .

C'est un **système de découpage par graphe**.

Proposition. C'est possible sans ajouter de sommets à l'intérieur de \mathcal{M} (si $n \geq 1$).

Complétion

$s = (s_1, \dots, s_n)$ plongement de graphe.



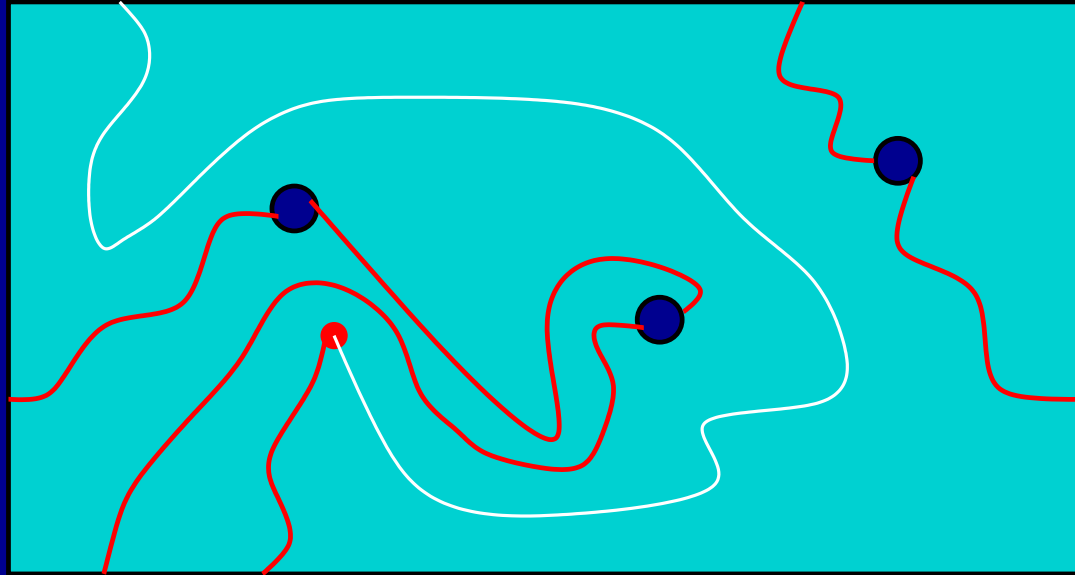
Complétion en un plongement de graphe $S = (s_1, \dots, s_N)$ tel que, pour tout i , la face de $S \setminus s_i$ contenant s_i est un disque, séparé en deux par s_i .

C'est un **système de découpage par graphe**.

Proposition. C'est possible sans ajouter de sommets à l'intérieur de \mathcal{M} (si $n \geq 1$).

Complétion

$s = (s_1, \dots, s_n)$ plongement de graphe.



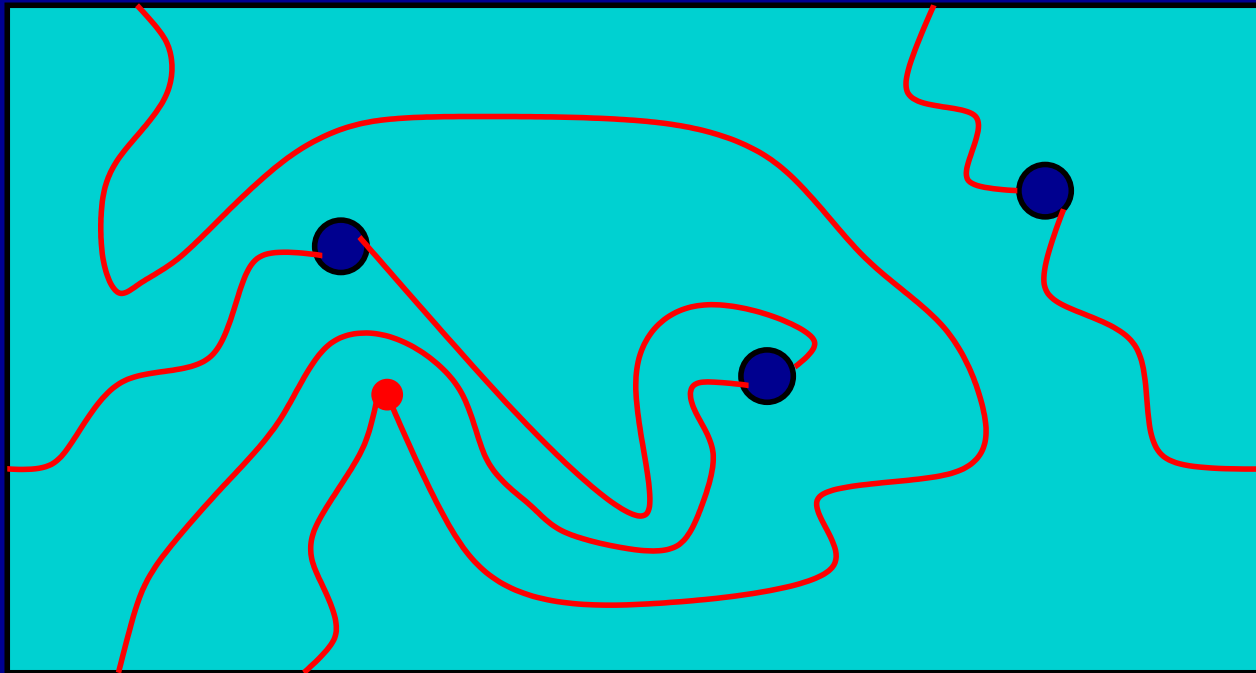
Complétion en un plongement de graphe $S = (s_1, \dots, s_N)$ tel que, pour tout i , la face de $S \setminus s_i$ contenant s_i est un disque, séparé en deux par s_i .

C'est un **système de découpage par graphe**.

Proposition. C'est possible sans ajouter de sommets à l'intérieur de \mathcal{M} (si $n \geq 1$).

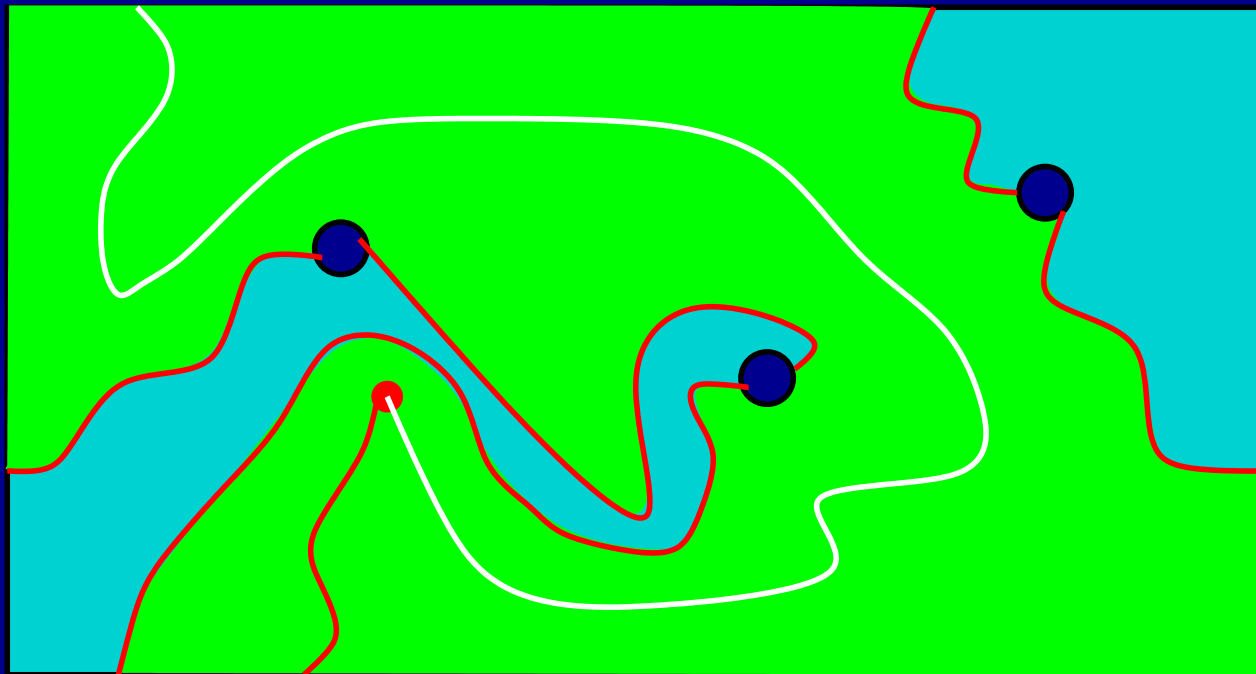
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: raccourcir s_i dans le disque de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



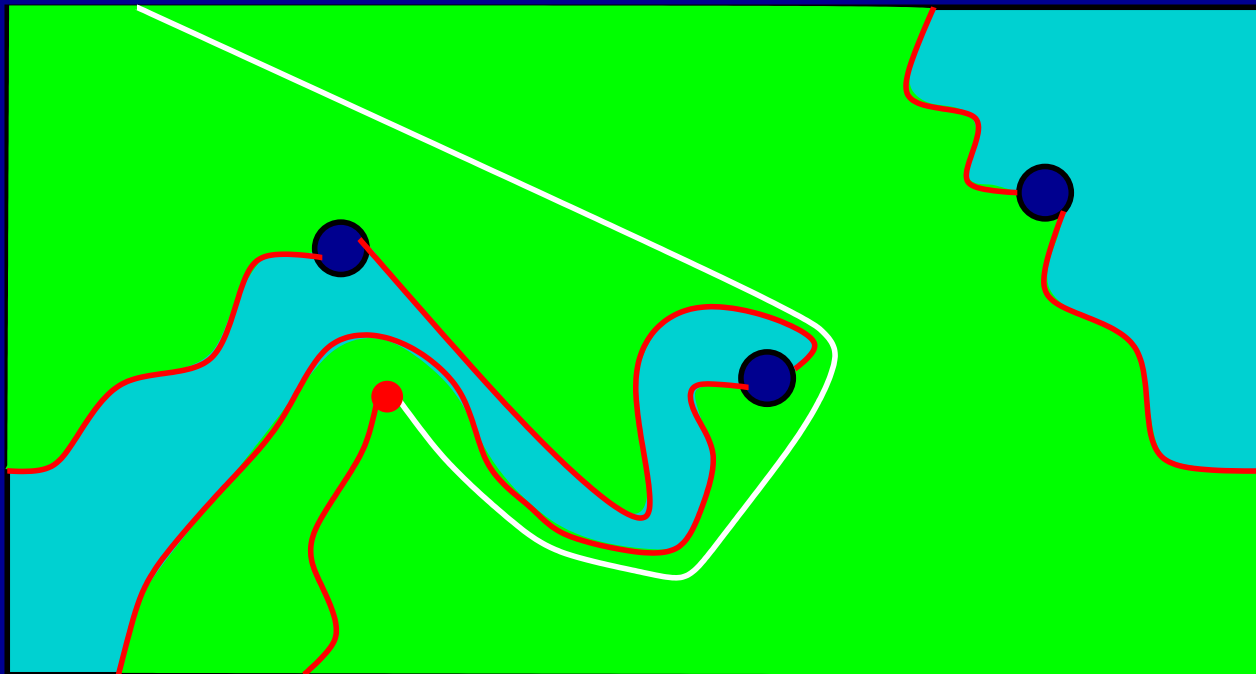
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: raccourcir s_i dans le disque de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: raccourcir s_i dans le disque de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: raccourcir s_i dans le disque de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: raccourcir s_i dans le disque de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: raccourcir s_i dans le disque de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: raccourcir s_i dans le disque de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: raccourcir s_i dans le disque de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: raccourcir s_i dans le disque de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



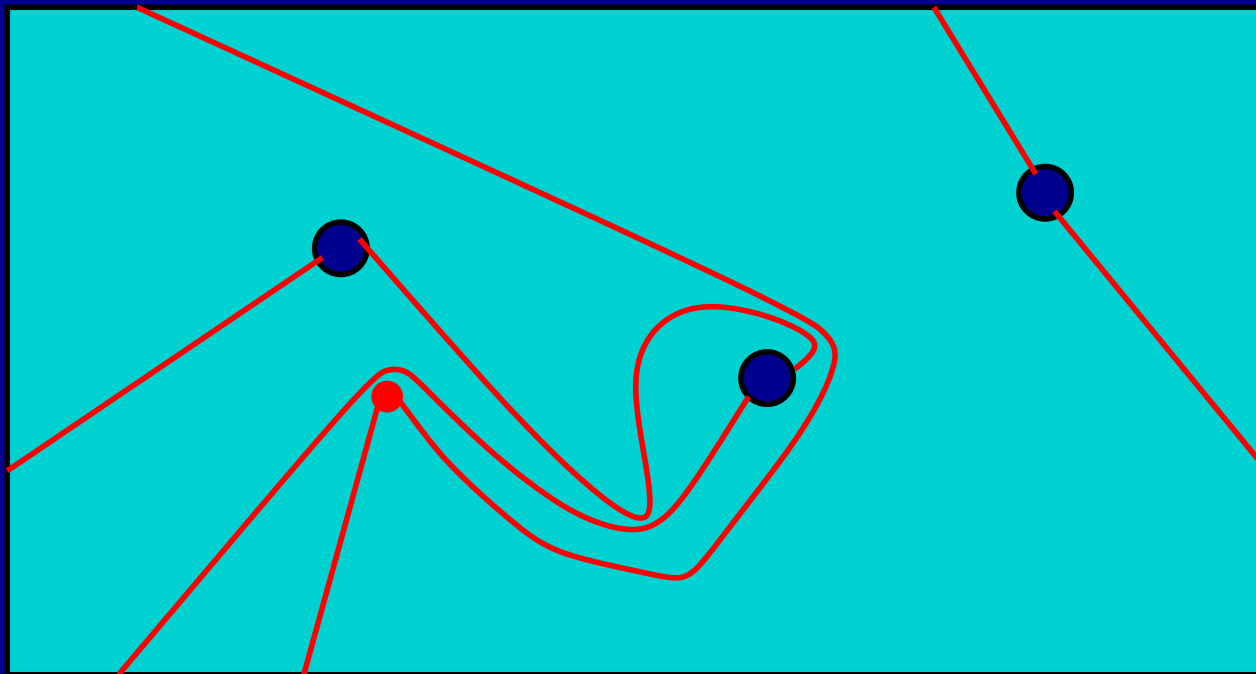
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: raccourcir s_i dans le disque de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



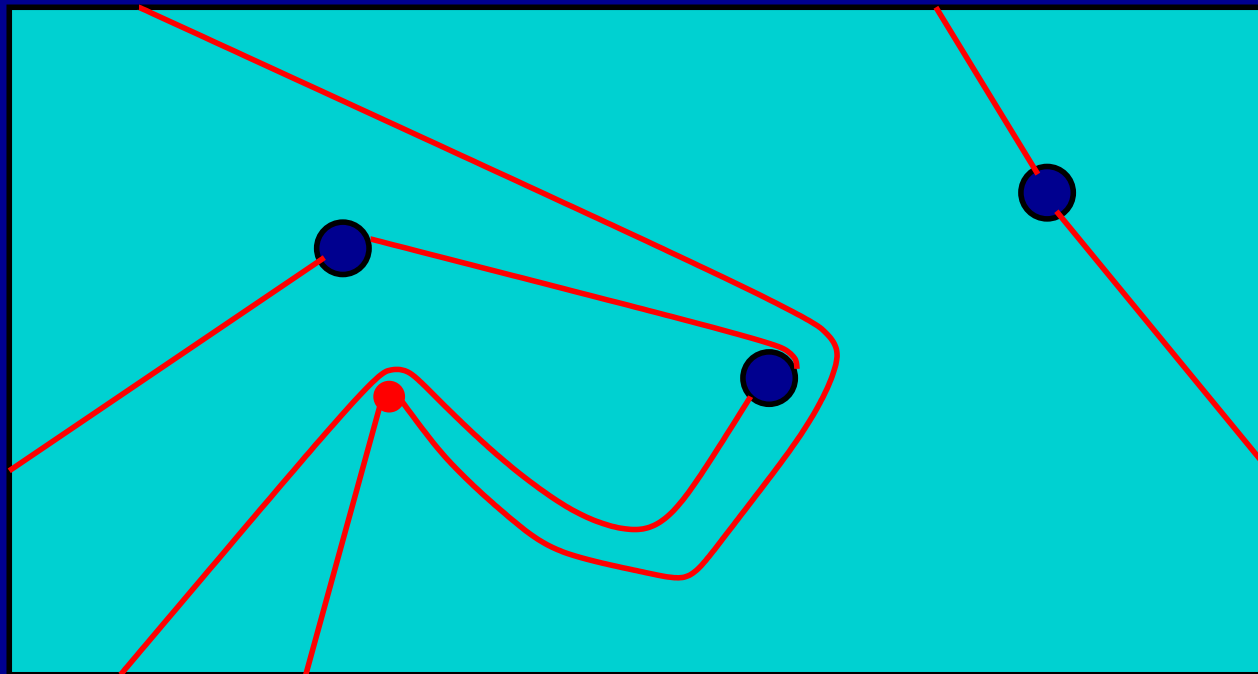
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: raccourcir s_i dans le disque de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



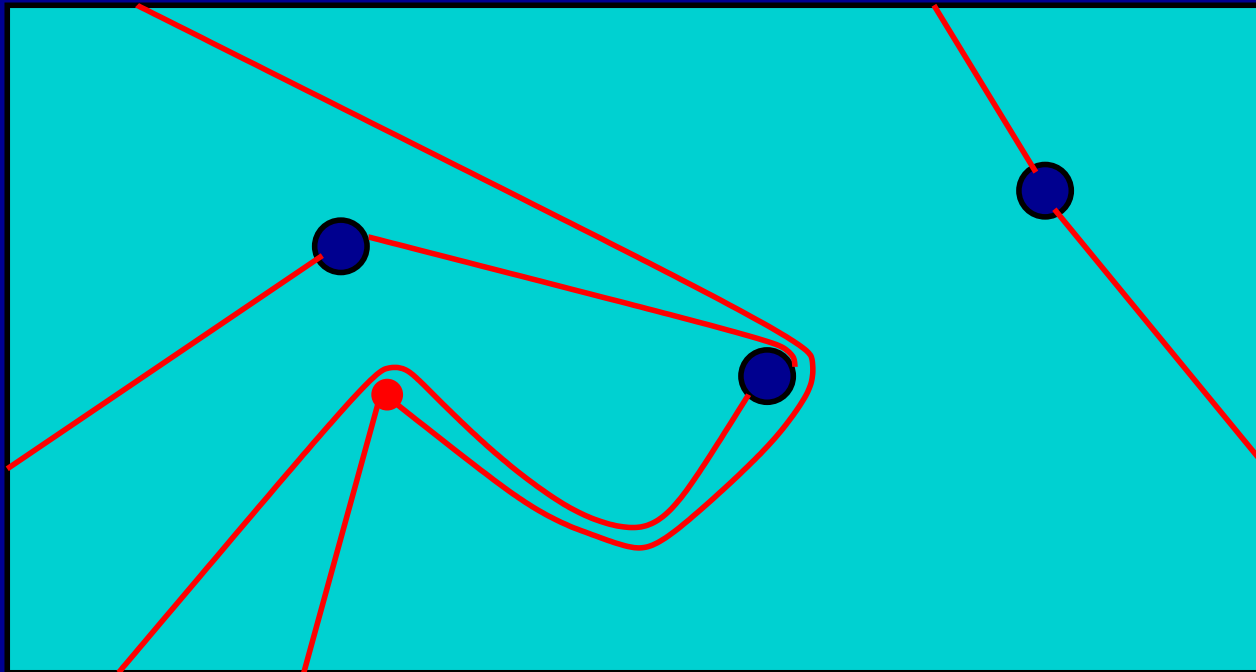
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: raccourcir s_i dans le disque de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



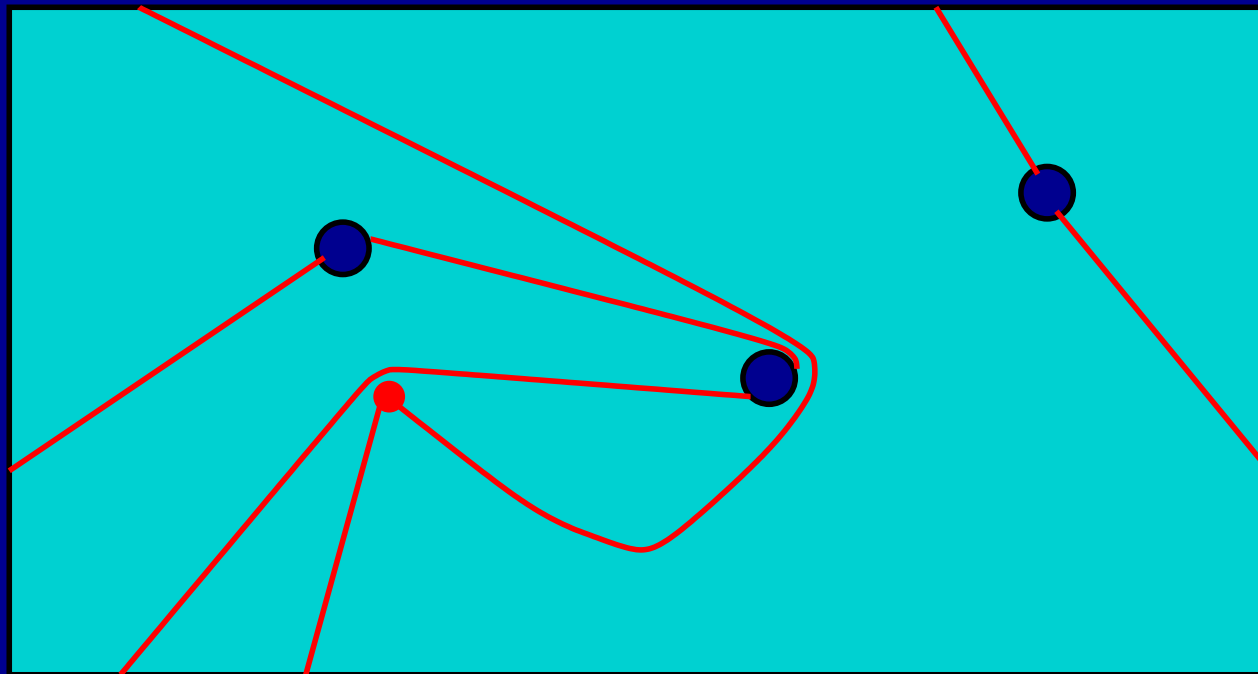
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: raccourcir s_i dans le disque de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



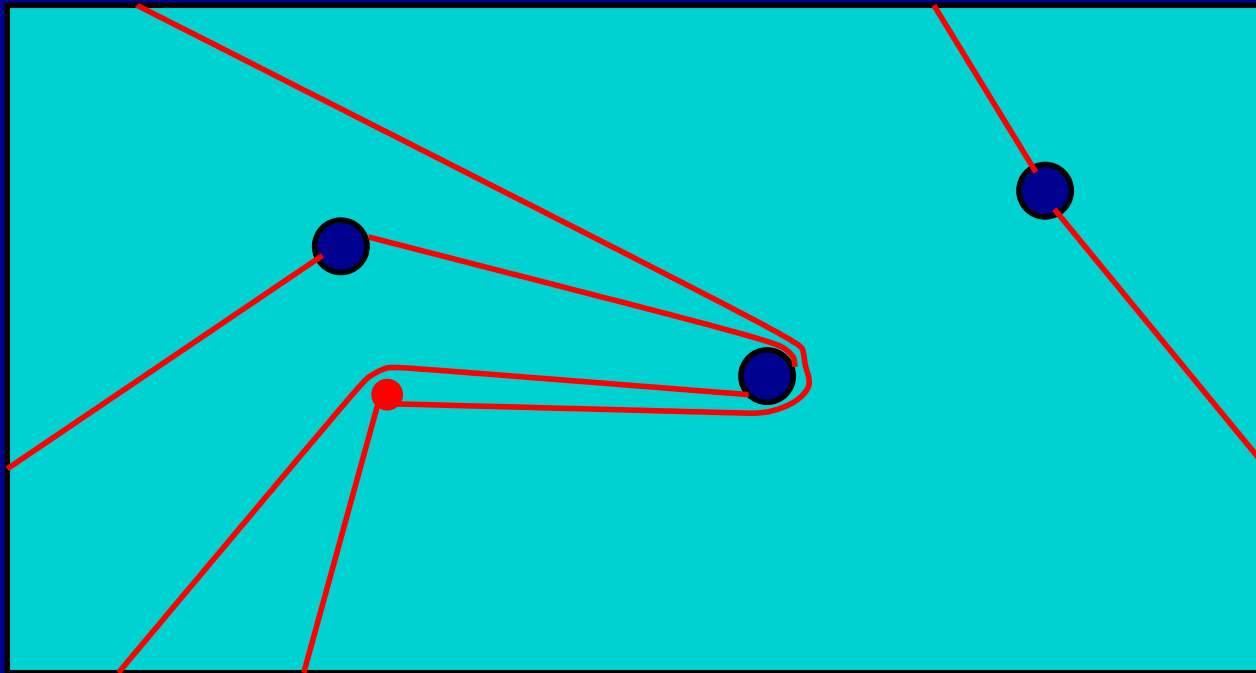
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: raccourcir s_i dans le disque de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: raccourcir s_i dans le disque de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



Résultat – cas particulier

Théorème (I). Si tous les sommets sont sur le bord de \mathcal{M} :

- Stabilité atteinte : $\exists m, |\text{rac}^{m+1}(S)| = |\text{rac}^m(S)|$;
- soit $S' = \text{rac}^m(S)$. Pour tout $i \in [1 \dots N]$, s'_i est un plus court chemin parmi tous les chemins homotopes à s_i .

Conséquence. On peut, pour tout i , remplacer s_i par un plus court chemin homotope s'_i de sorte que les s'_i soient simples et deux à deux disjoints.

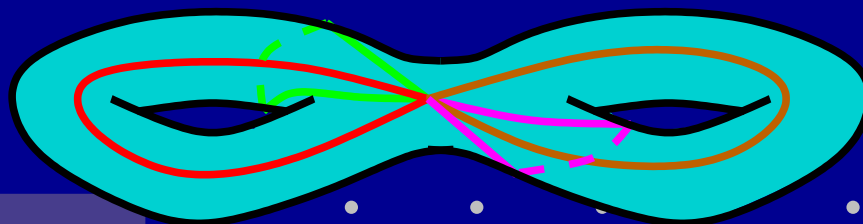
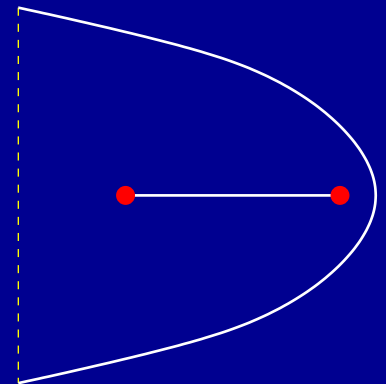
Résultat – cas général

Théorème (II).

- $\exists m, |\text{rac}^{m+1}(S)| = |\text{rac}^m(S)|$;
- soit $S' = \text{rac}^m(S)$. Pour tout $i \in [1 \dots N]$, s'_i est un plus court chemin parmi tous les chemins simples isotopes à s_i dans \mathcal{M} privé des sommets de S non extrémités de s_i .

Conséquences :

- s' est de longueur minimale parmi les plongements de graphe **isotopes à sommets fixés** à s ;
- si s est un **système fondamental de lacets**, alors, pour tout i , s'_i est de longueur minimale parmi les lacets simples homotopes à s_i (avec [Epstein, 1966]).



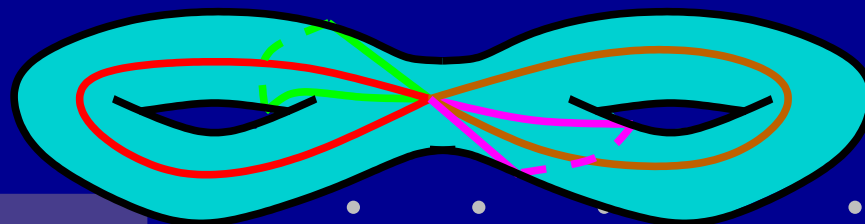
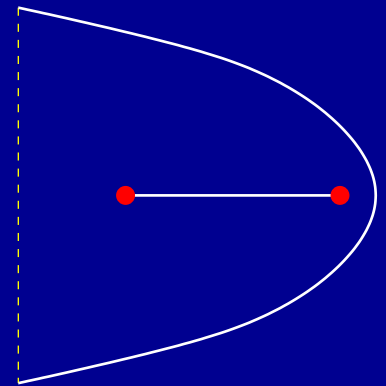
Résultat – cas général

Théorème (II).

- $\exists m, |\text{rac}^{m+1}(S)| = |\text{rac}^m(S)|$;
- soit $S' = \text{rac}^m(S)$. Pour tout $i \in [1 \dots N]$, s'_i est un plus court chemin parmi tous les chemins simples isotopes à s_i dans \mathcal{M} privé des sommets de s non extrémités de s_i .

Conséquences :

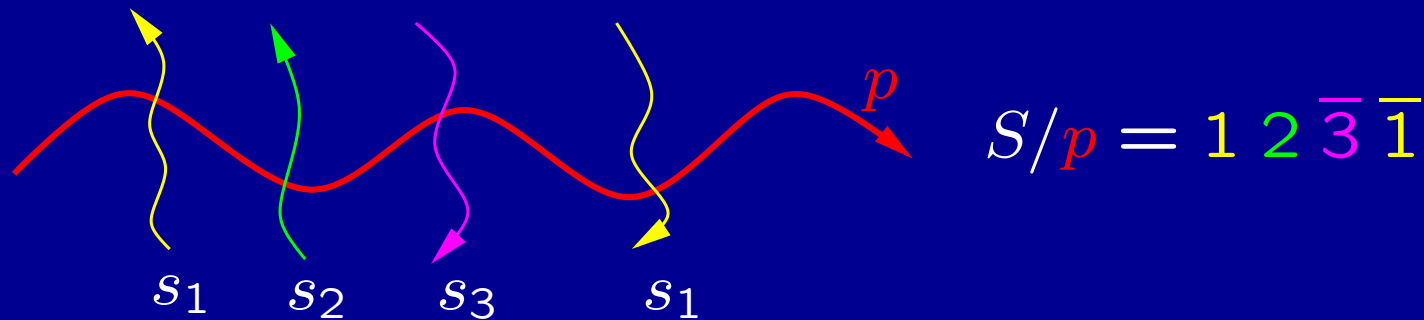
- s' est de longueur minimale parmi les plongements de graphe **isotopes à sommets fixés** à s ;
- si s est un **système fondamental de lacets**, alors, pour tout i , s'_i est de longueur minimale parmi les lacets simples homotopes à s_i (avec [Epstein, 1966]).



Mot des croisements

S : chemins simples et deux à deux disjoints dont les extrémités sont sur $\partial\mathcal{M}$.

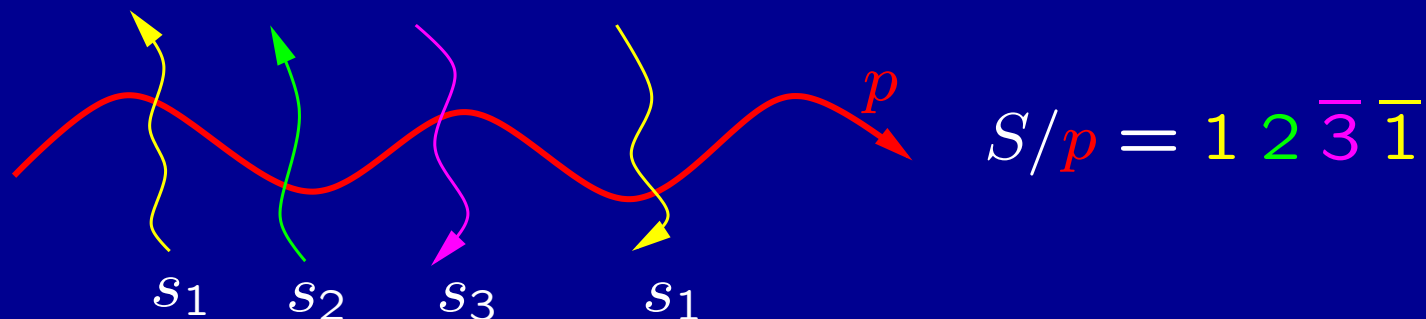
Mot des croisements S/p d'un chemin p avec S : liste des croisements.



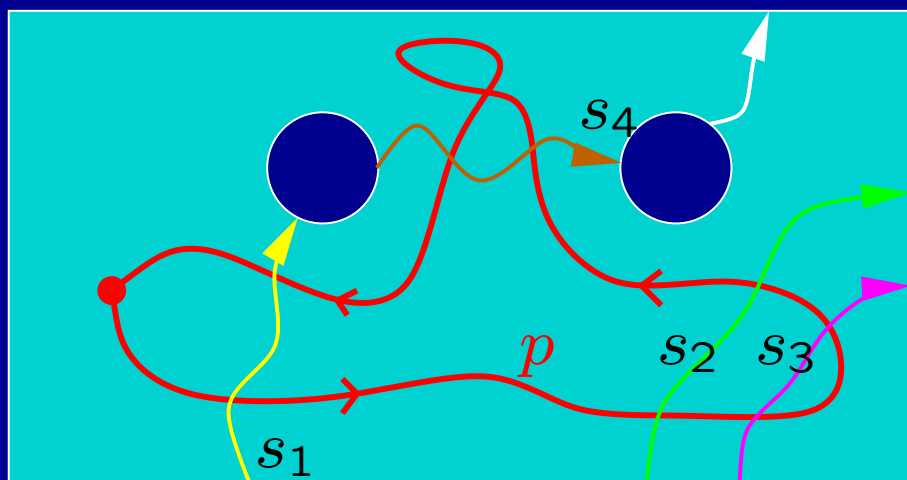
Mot des croisements

S : chemins simples et deux à deux disjoints dont les extrémités sont sur $\partial\mathcal{M}$.

Mot des croisements S/p d'un chemin p avec S : liste des croisements.



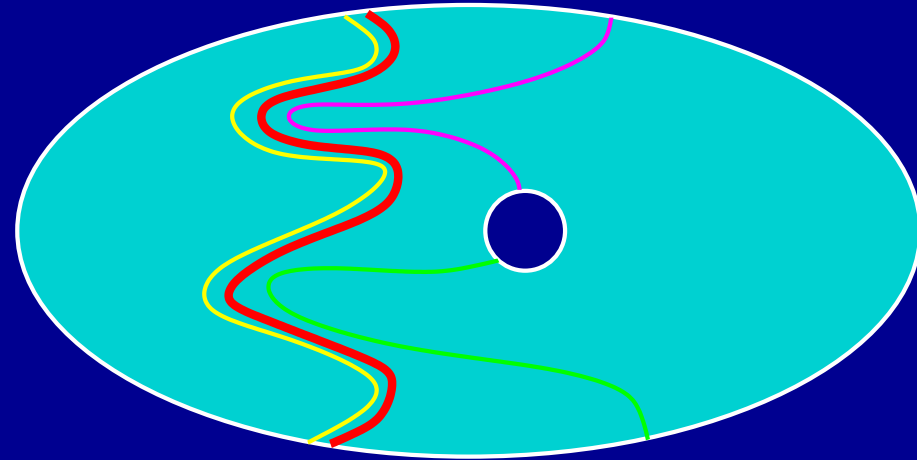
Si p est un chemin fermé contractile, alors S/p est parenthésé.



$$S/p = 1\ 2\ \underbrace{3\ \bar{3}}\ \underbrace{\bar{2}\ \bar{4}\ 4}\ \bar{1}$$

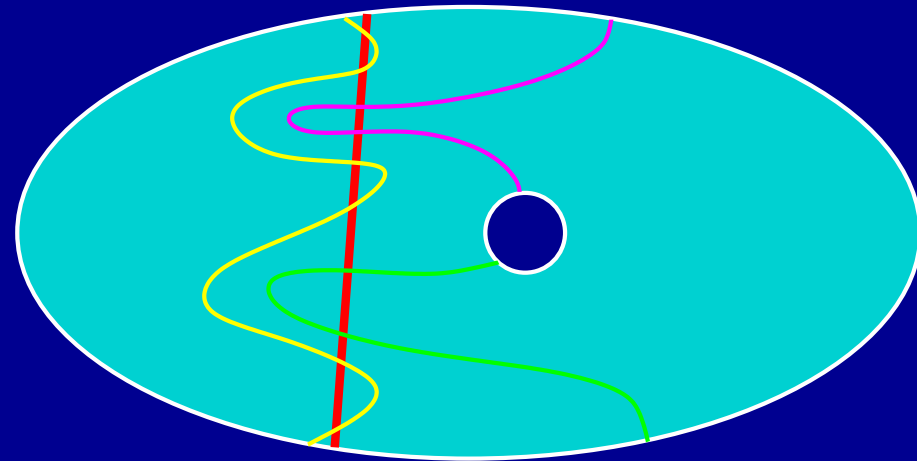
Démonstration de (I)

- $i \in [1 \dots N]$;
- t_i « longeant » s_i ;
- raccourcir t_i autant que possible dans sa classe d'homotopie.
 t_i n'est pas forcément simple.



Démonstration de (I)

- $i \in [1 \dots N]$;
- t_i « longeant » s_i ;
- raccourcir t_i autant que possible dans sa classe d'homotopie.
 t_i n'est pas forcément simple.

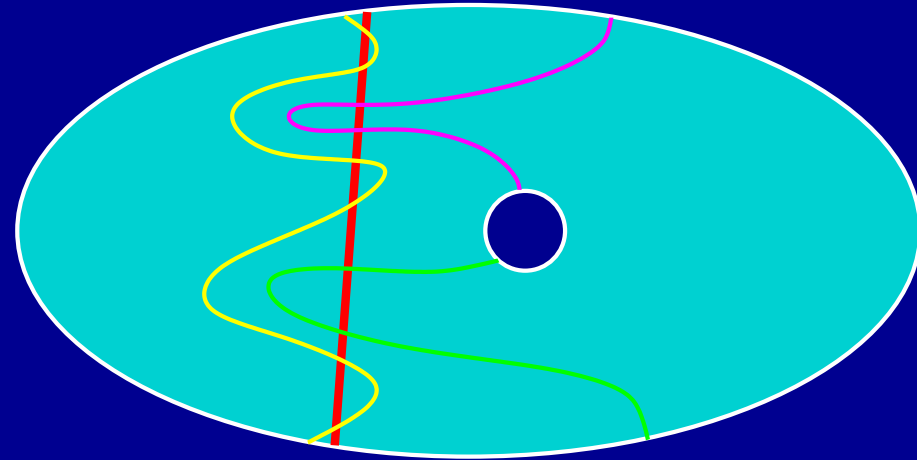


$$S/t_i = 1 \bar{1} 2 \bar{2} 1 \bar{1} 3 \bar{3} 1 \bar{1}$$

But : montrer qu'à une certaine étape, on a $|s_i| = |t_i|$.

Démonstration de (I)

- $i \in [1 \dots N]$;
- t_i « longeant » s_i ;
- raccourcir t_i autant que possible dans sa classe d'homotopie.
 t_i n'est pas forcément simple.



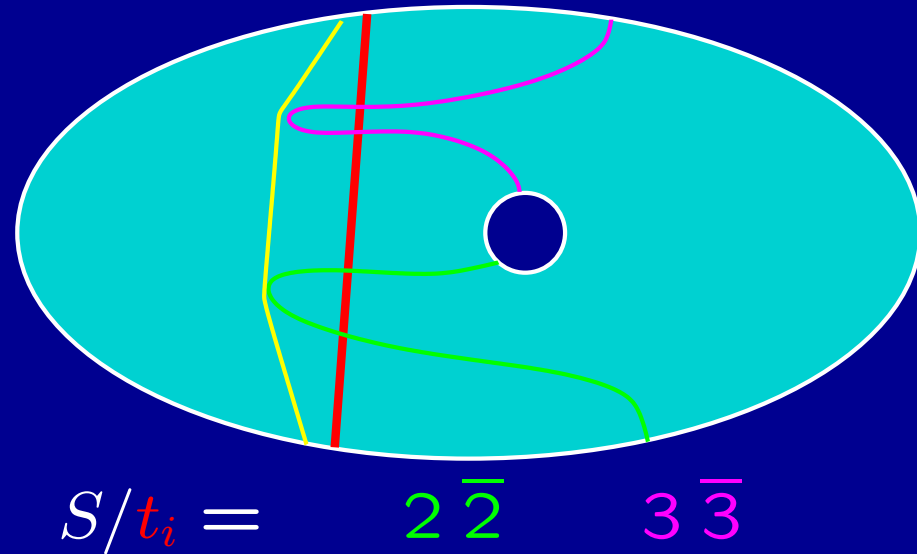
$$S/t_i = 1 \bar{1} 2 \bar{2} 1 \bar{1} 3 \bar{3} 1 \bar{1}$$

But : montrer qu'à une certaine étape, on a $|s_i| = |t_i|$.

- S/t_i est parenthésé ;
- chaque application de rac_j supprime les $j\bar{j}$ et $\bar{j}j$:
 $(\text{rac}_j(S))/t_i = \text{red}_j(S/t_i)$;
- si S et t_i sont disjoints, après une itération supplémentaire, le $i^{\text{ème}}$ chemin et t_i ont même longueur.

Démonstration de (I)

- $i \in [1 \dots N]$;
- t_i « longeant » s_i ;
- raccourcir t_i autant que possible dans sa classe d'homotopie.
 t_i n'est pas forcément simple.

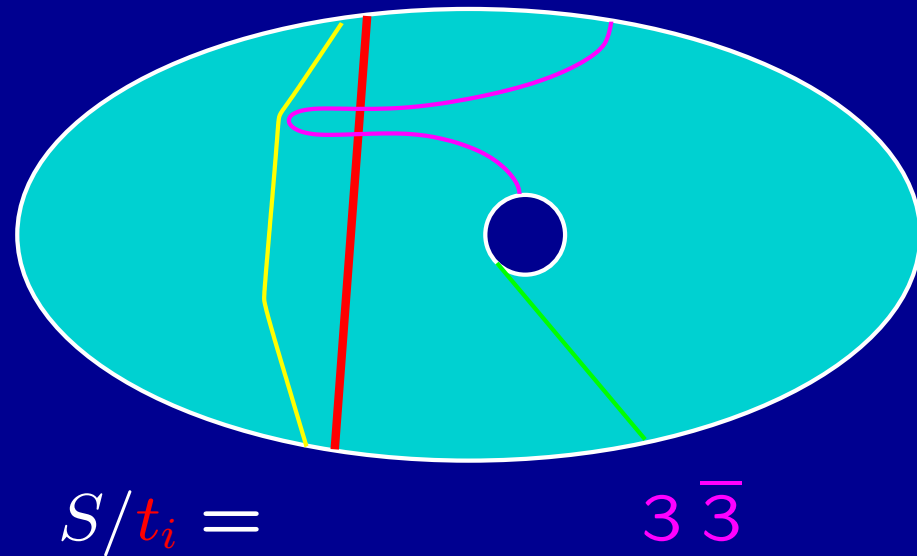


But : montrer qu'à une certaine étape, on a $|s_i| = |t_i|$.

- S/t_i est parenthésé ;
- chaque application de rac_j supprime les $j\bar{j}$ et $\bar{j}j$:
 $(\text{rac}_j(S))/t_i = \text{red}_j(S/t_i)$;
- si S et t_i sont disjoints, après une itération supplémentaire, le $i^{\text{ème}}$ chemin et t_i ont même longueur.

Démonstration de (I)

- $i \in [1 \dots N]$;
- t_i « longeant » s_i ;
- raccourcir t_i autant que possible dans sa classe d'homotopie.
 t_i n'est pas forcément simple.

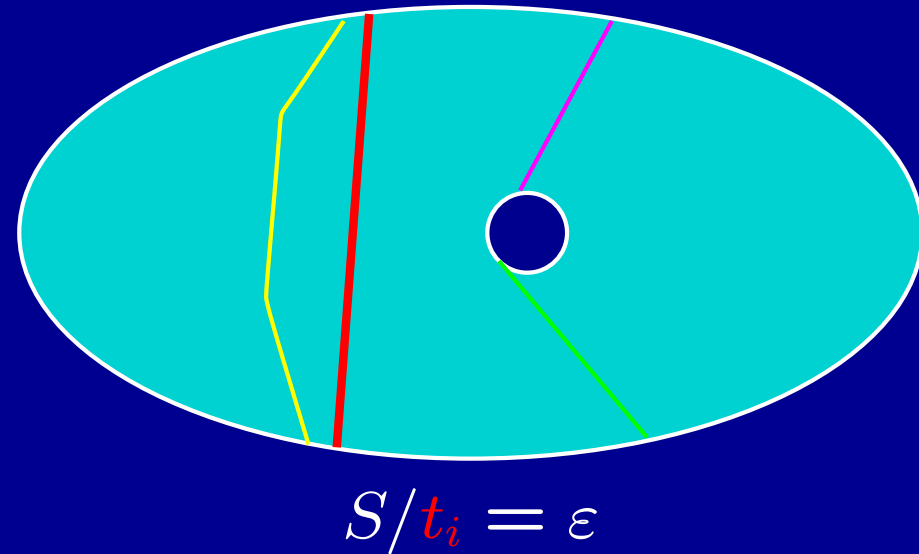


But : montrer qu'à une certaine étape, on a $|s_i| = |t_i|$.

- S/t_i est parenthésé ;
- chaque application de rac_j supprime les $j\bar{j}$ et $\bar{j}j$:
 $(\text{rac}_j(S))/t_i = \text{red}_j(S/t_i)$;
- si S et t_i sont disjoints, après une itération supplémentaire, le $i^{\text{ème}}$ chemin et t_i ont même longueur.

Démonstration de (I)

- $i \in [1 \dots N]$;
- t_i « longeant » s_i ;
- raccourcir t_i autant que possible dans sa classe d'homotopie.
 t_i n'est pas forcément simple.

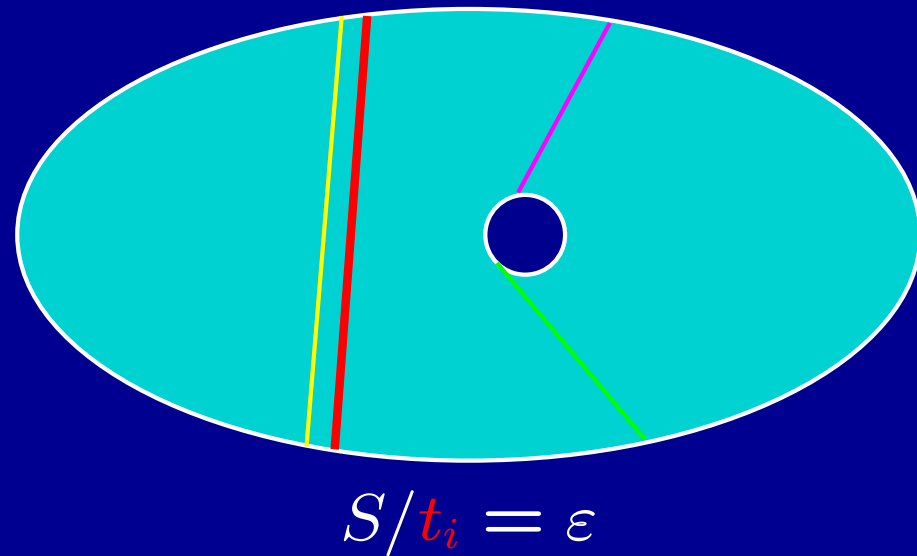


But : montrer qu'à une certaine étape, on a $|s_i| = |t_i|$.

- S/t_i est parenthésé ;
- chaque application de rac_j supprime les $j\bar{j}$ et $\bar{j}j$:
 $(\text{rac}_j(S))/t_i = \text{red}_j(S/t_i)$;
- si S et t_i sont disjoints, après une itération supplémentaire, le $i^{\text{ème}}$ chemin et t_i ont même longueur.

Démonstration de (I)

- $i \in [1 \dots N]$;
- t_i « longeant » s_i ;
- raccourcir t_i autant que possible dans sa classe d'homotopie.
 t_i n'est pas forcément simple.

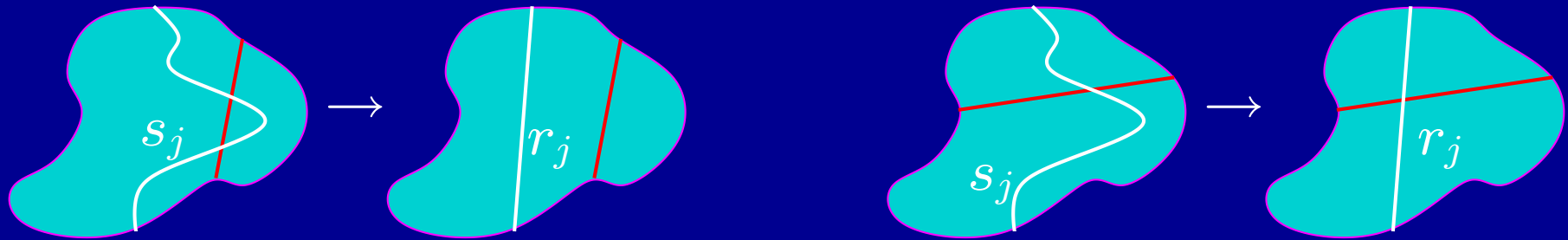


But : montrer qu'à une certaine étape, on a $|s_i| = |t_i|$.

- S/t_i est parenthésé ;
- chaque application de rac_j supprime les $j\bar{j}$ et $\bar{j}j$:
 $(\text{rac}_j(S))/t_i = \text{red}_j(S/t_i)$;
- si S et t_i sont disjoints, après une itération supplémentaire, le $i^{\text{ème}}$ chemin et t_i ont même longueur.

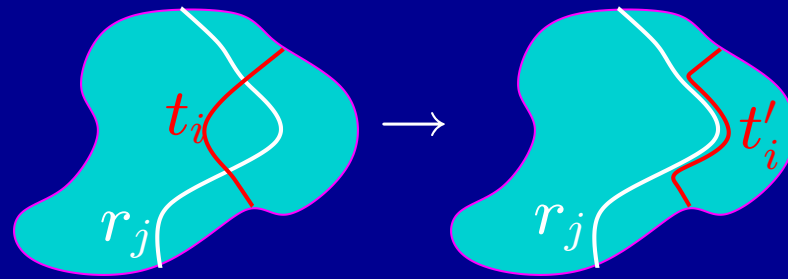
Démonstration de (I), suite et fin

- Si $R = \text{rac}_j(S)$ alors : $\text{red}_j(R/t_i) = \text{red}_j(S/t_i)$.

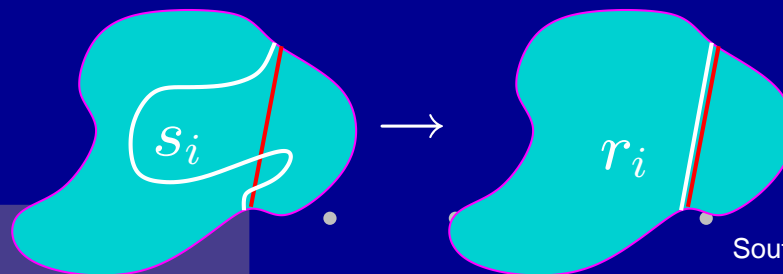


- Il existe t'_i ayant les mêmes propriétés que t_i tel que :

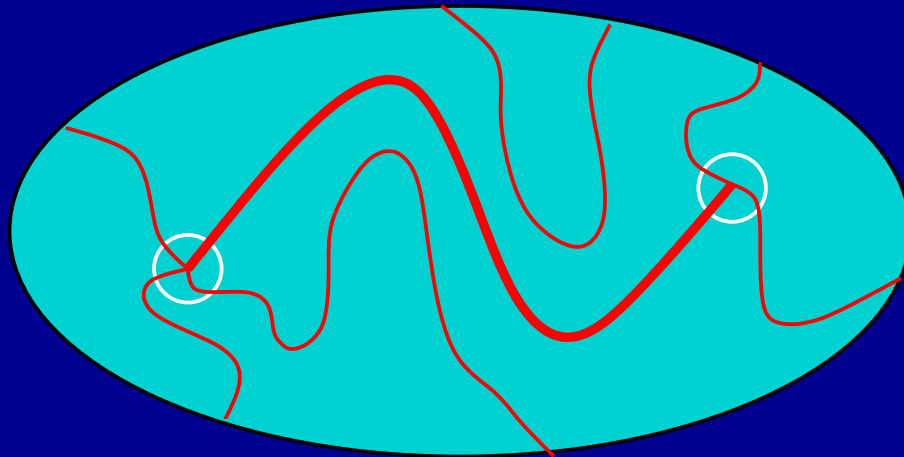
$$R/t'_i = \text{red}_j(S/t_i).$$



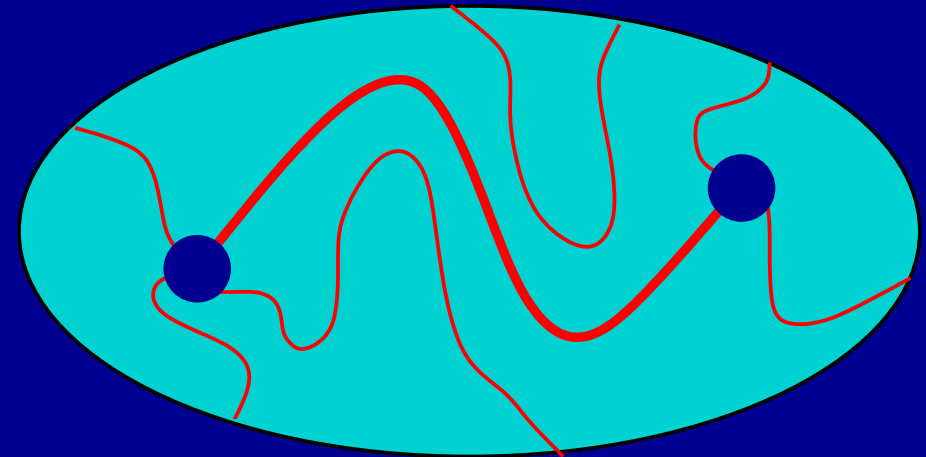
- À partir d'une certaine étape, S et t_i sont disjoints (quitte à changer t_i). Après une itération supplémentaire : $|s_i| = |t_i|$.



Idée de la démonstration de (II)



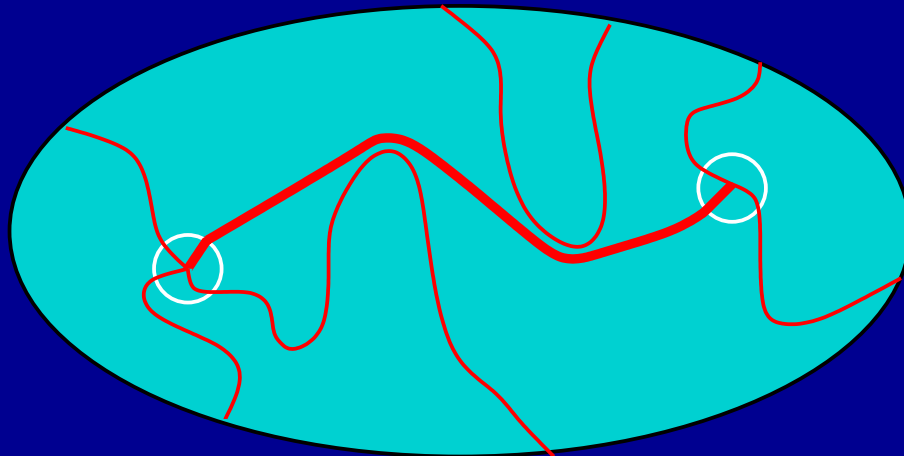
\mathcal{M}, S



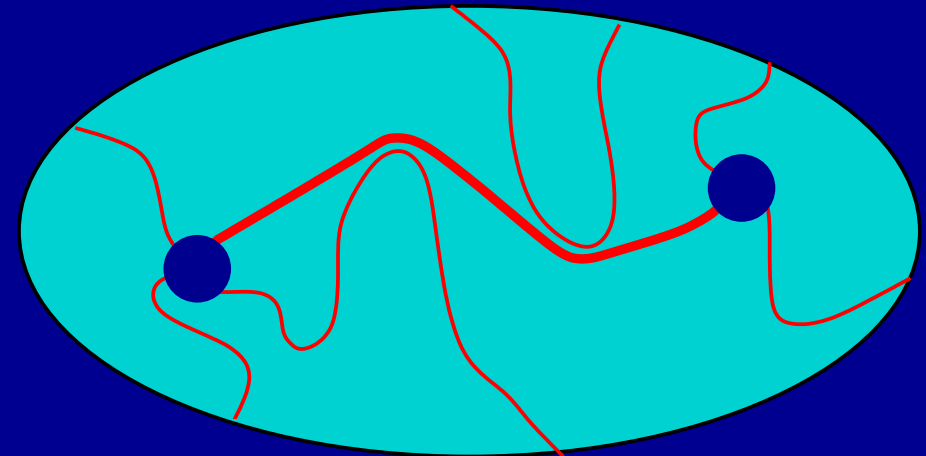
$\widehat{\mathcal{M}}, \widehat{S}$

- Petits disques autour des sommets de S ;
- les optimisations de S et \widehat{S} sont similaires ;
- on applique le théorème (I) à \widehat{S} : \widehat{s}'_i est optimal ;
- on se ramène à s'_i .

Idée de la démonstration de (II)



\mathcal{M}, S



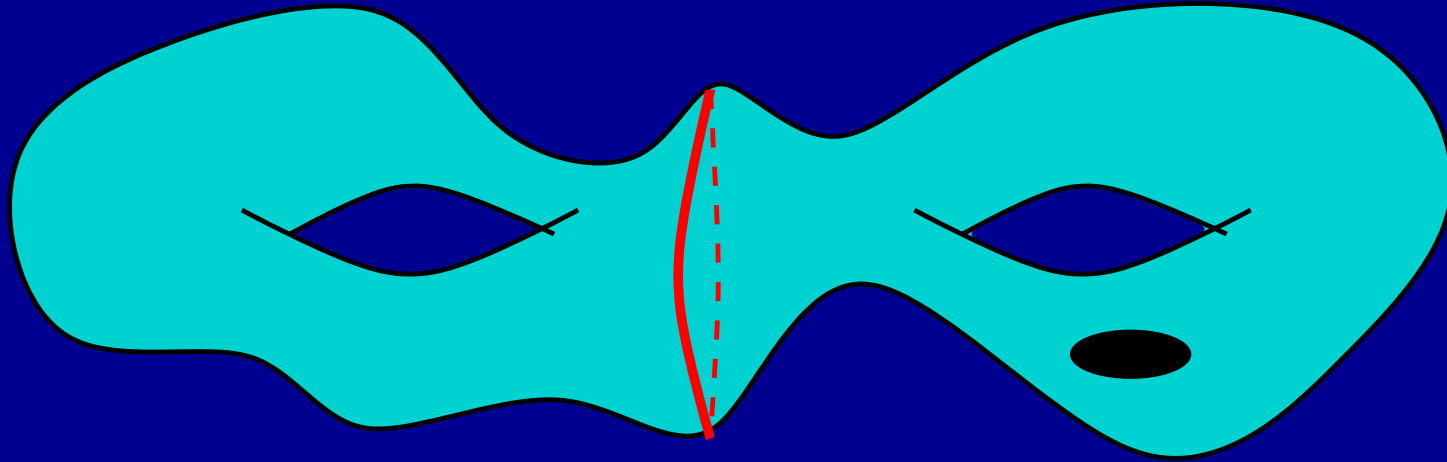
$\widehat{\mathcal{M}}, \widehat{S}$

- Petits disques autour des sommets de S ;
- les optimisations de S et \widehat{S} sont similaires ;
- on applique le théorème (I) à \widehat{S} : \widehat{s}'_i est optimal ;
- on se ramène à s'_i .

3. Plongements de cycles

Complétion

$s = (s_1, \dots, s_n)$ plongement de cycles.



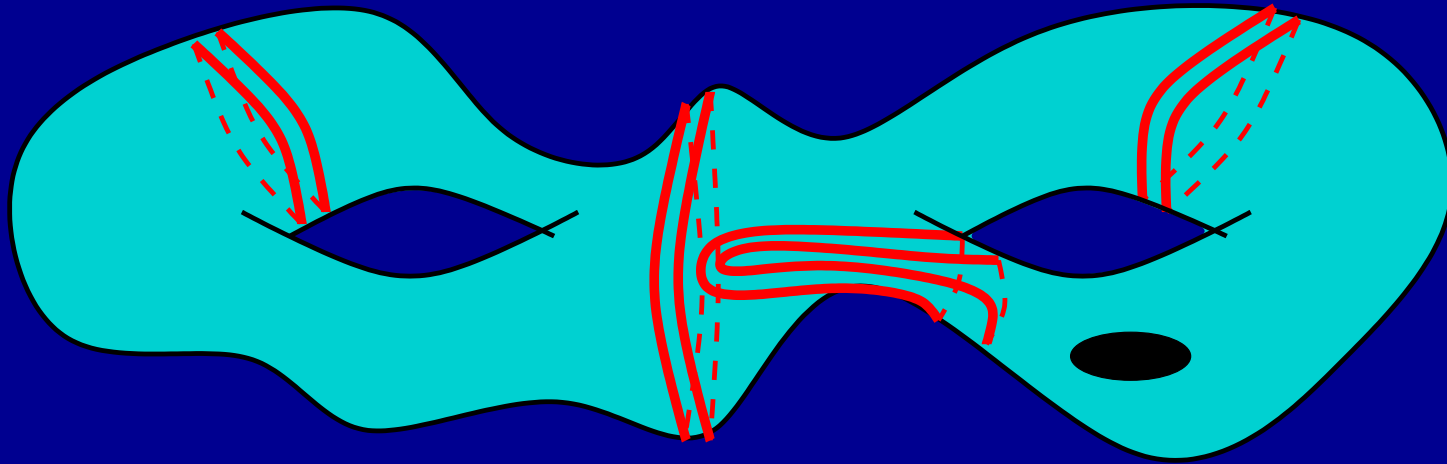
Complétion en un plongement de cycles S tel que, pour tout i , la face de $S \setminus s_i$ contenant s_i est un cylindre ou un pantalon, séparé en cylindres ou pantalons par s_i .

C'est un **système de découpage par cycles**.

Proposition. C'est toujours possible.

Complétion

$s = (s_1, \dots, s_n)$ plongement de cycles.



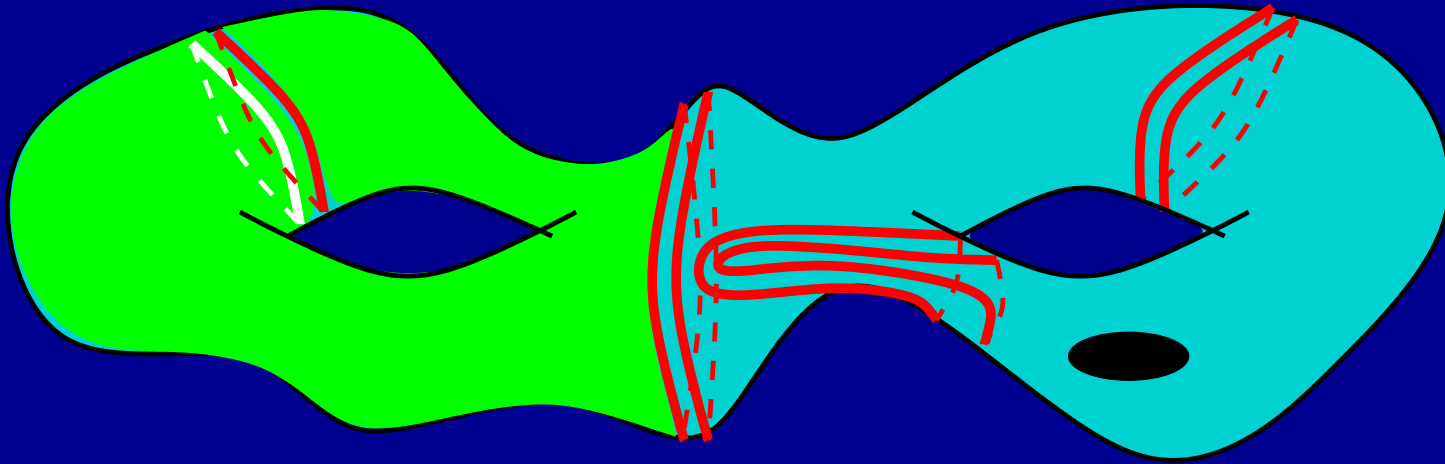
Complétion en un plongement de cycles S tel que, pour tout i , la face de $S \setminus s_i$ contenant s_i est un cylindre ou un pantalon, séparé en cylindres ou pantalons par s_i .

C'est un **système de découpage par cycles**.

Proposition. C'est toujours possible.

Complétion

$s = (s_1, \dots, s_n)$ plongement de cycles.



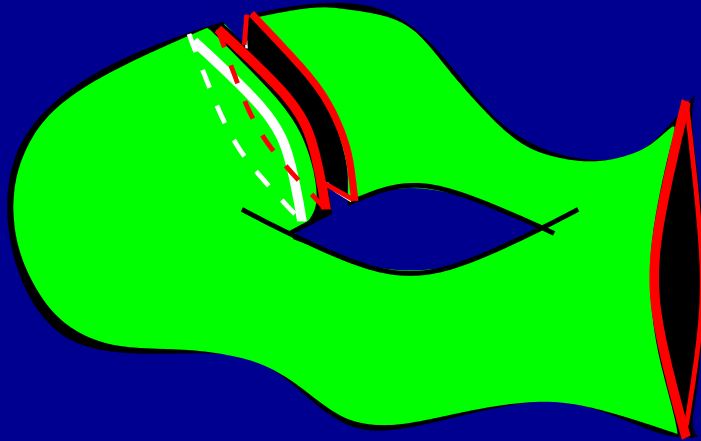
Complétion en un plongement de cycles S tel que, pour tout i , la face de $S \setminus s_i$ contenant s_i est un cylindre ou un pantalon, séparé en cylindres ou pantalons par s_i .

C'est un **système de découpage par cycles**.

Proposition. C'est toujours possible.

Complétion

$s = (s_1, \dots, s_n)$ plongement de cycles.



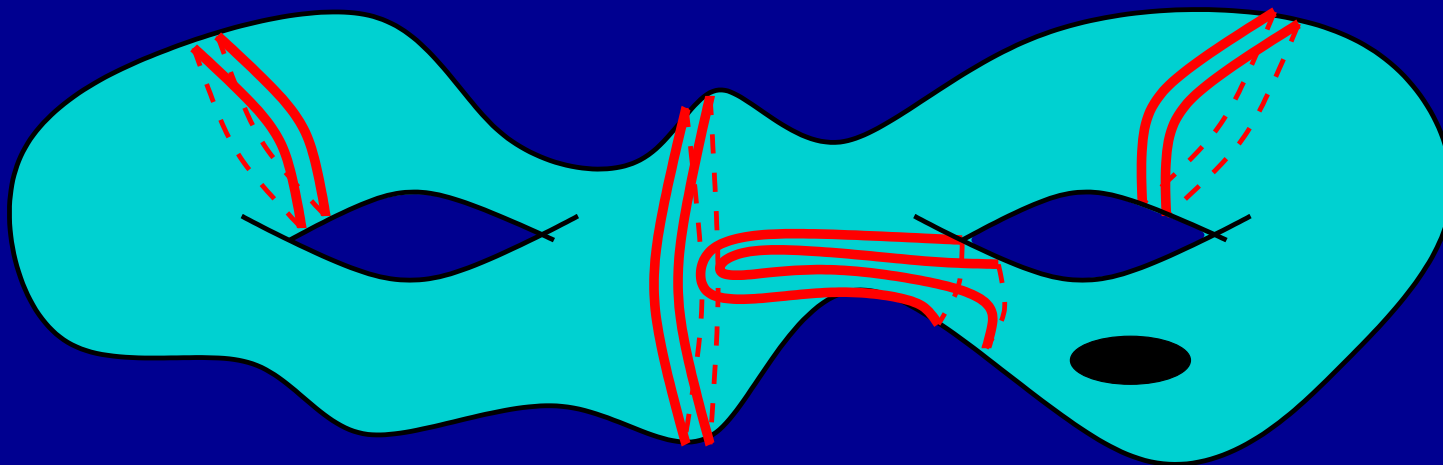
Complétion en un plongement de cycles S tel que, pour tout i , la face de $S \setminus s_i$ contenant s_i est un cylindre ou un pantalon, séparé en cylindres ou pantalons par s_i .

C'est un **système de découpage par cycles**.

Proposition. C'est toujours possible.

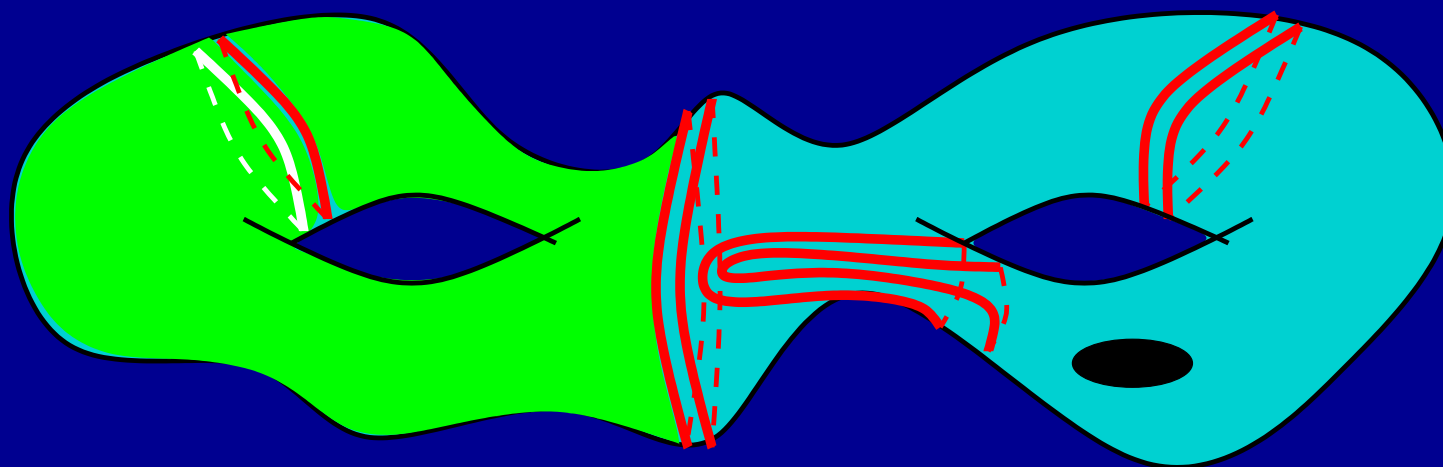
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



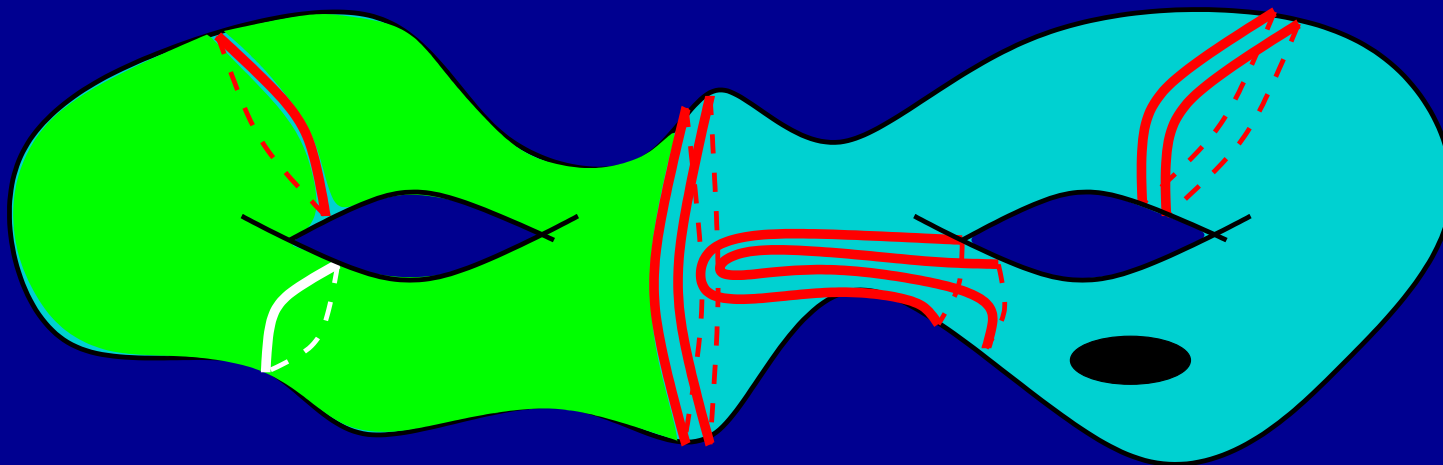
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



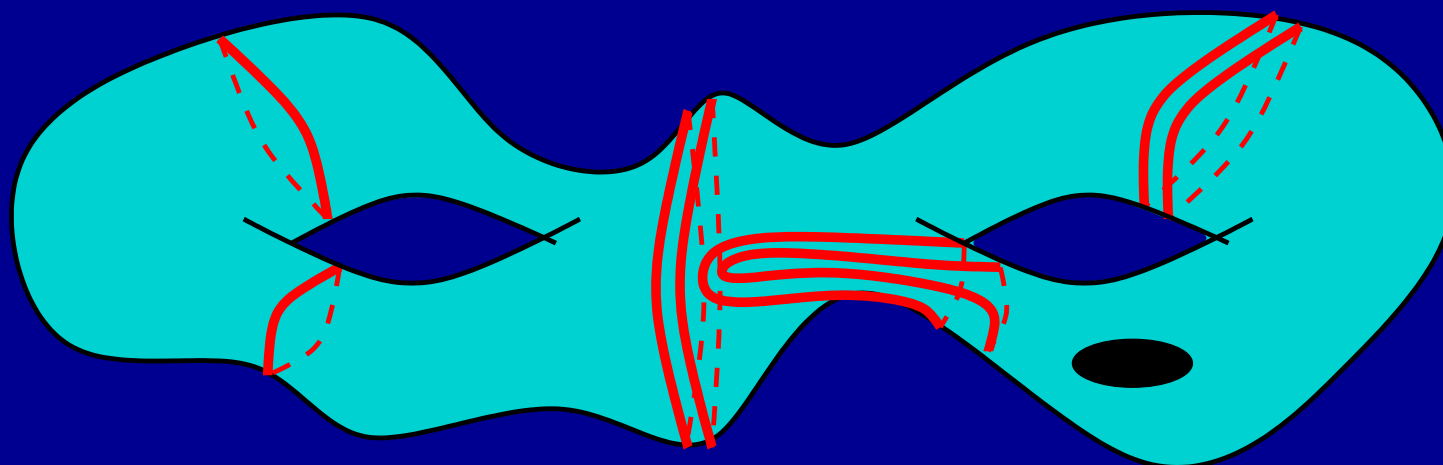
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



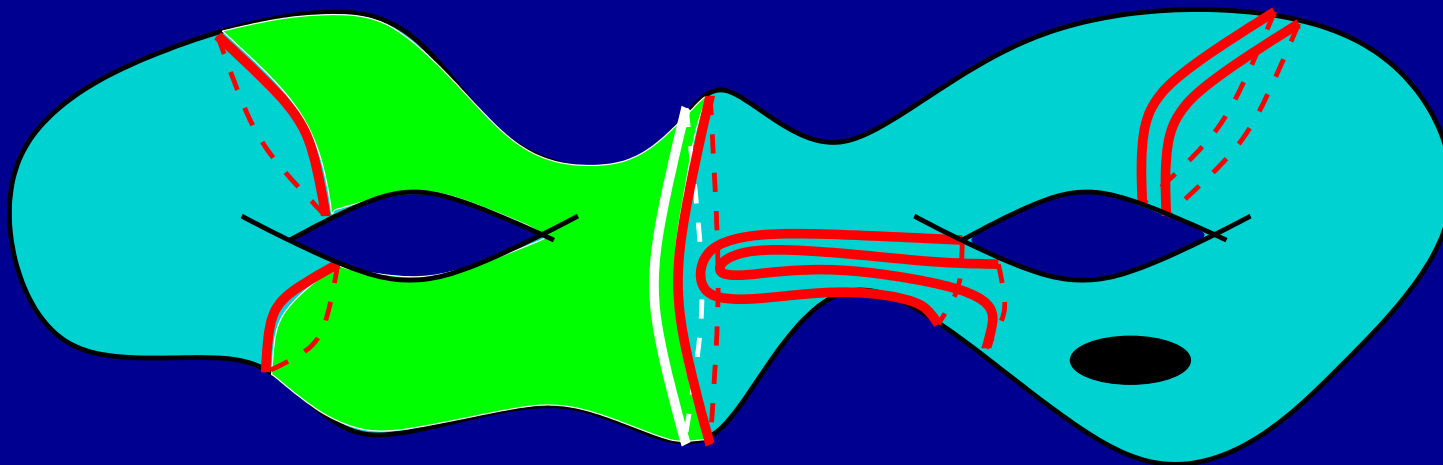
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



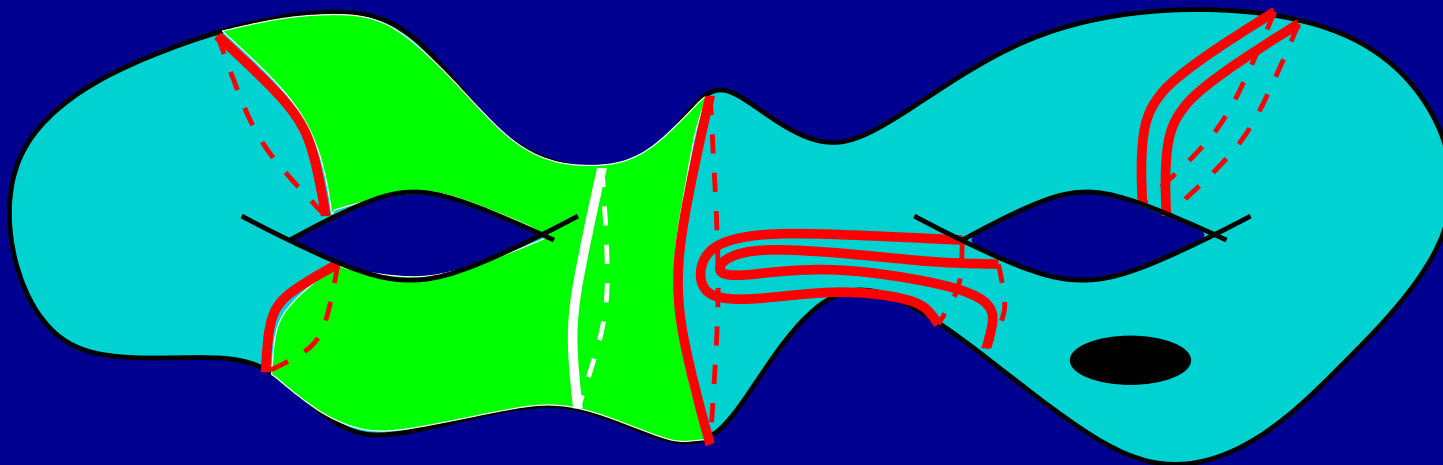
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



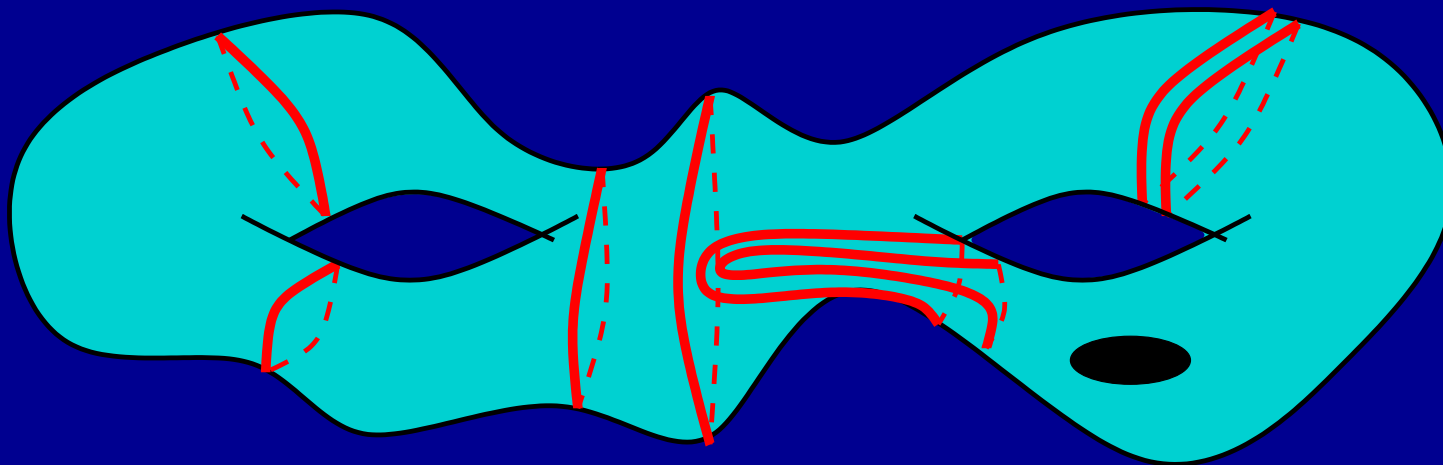
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



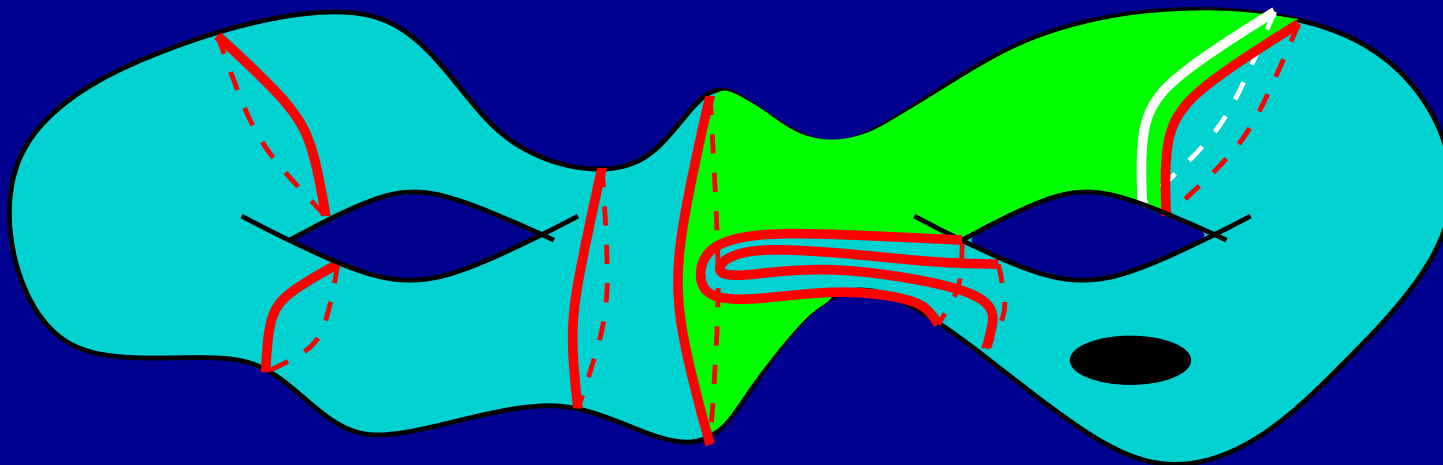
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



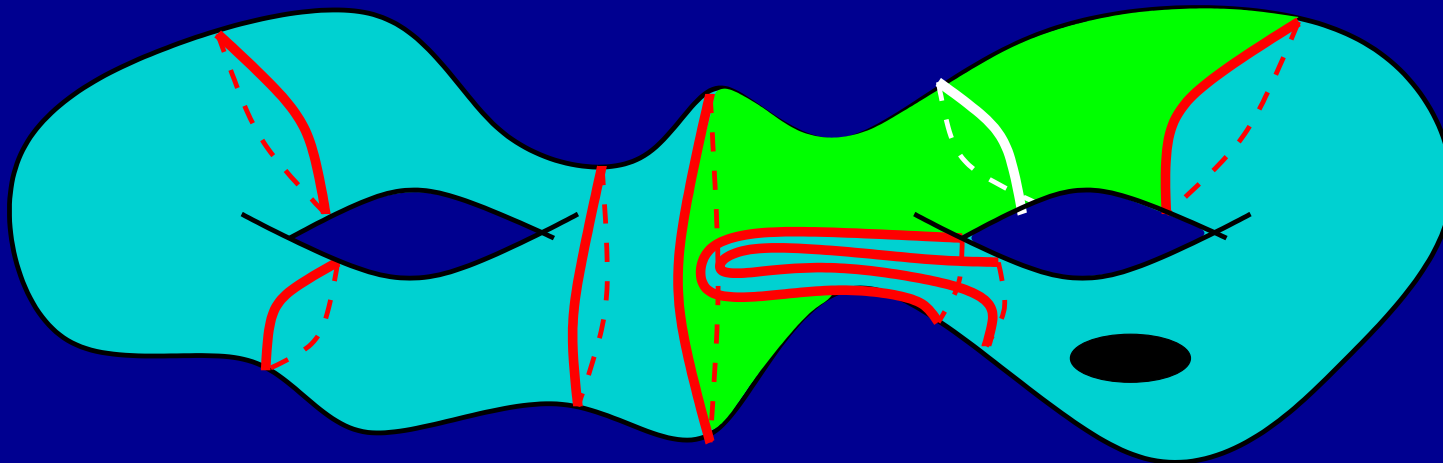
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



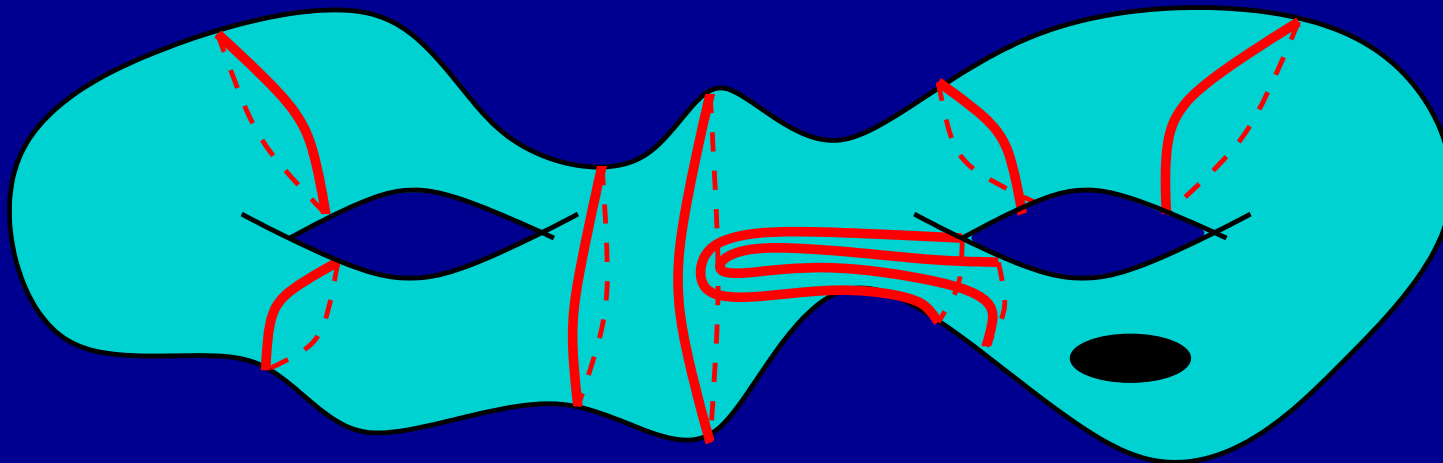
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



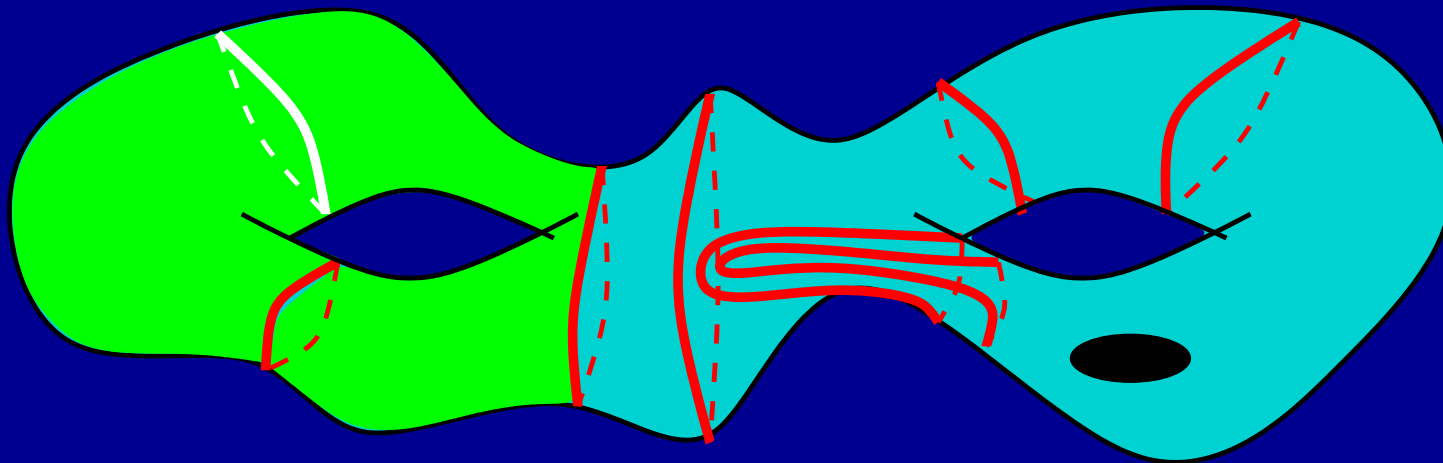
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



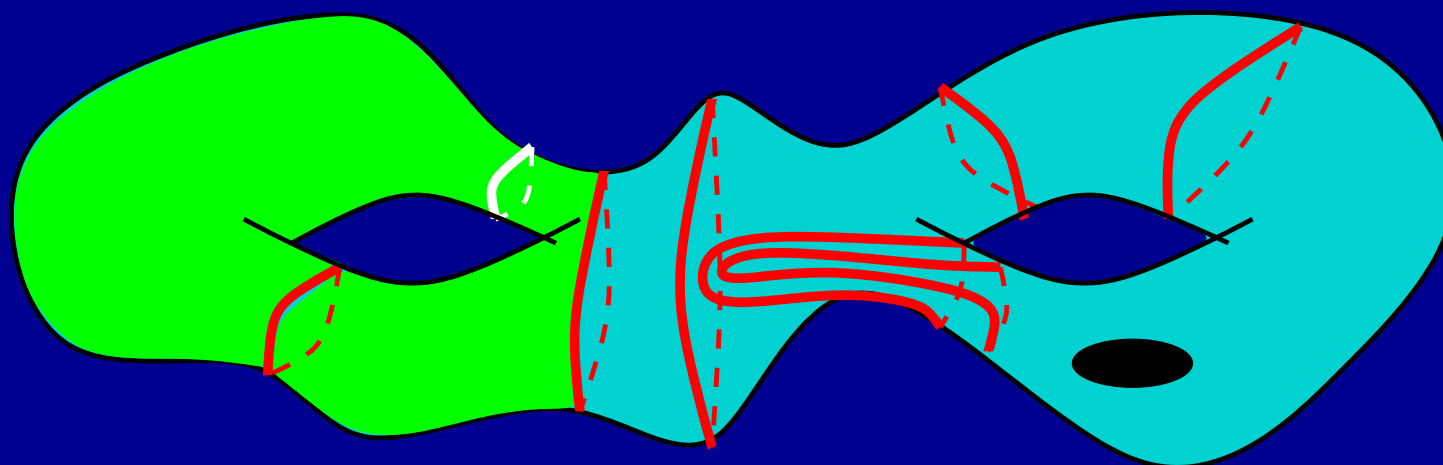
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



Optimisation et extraction

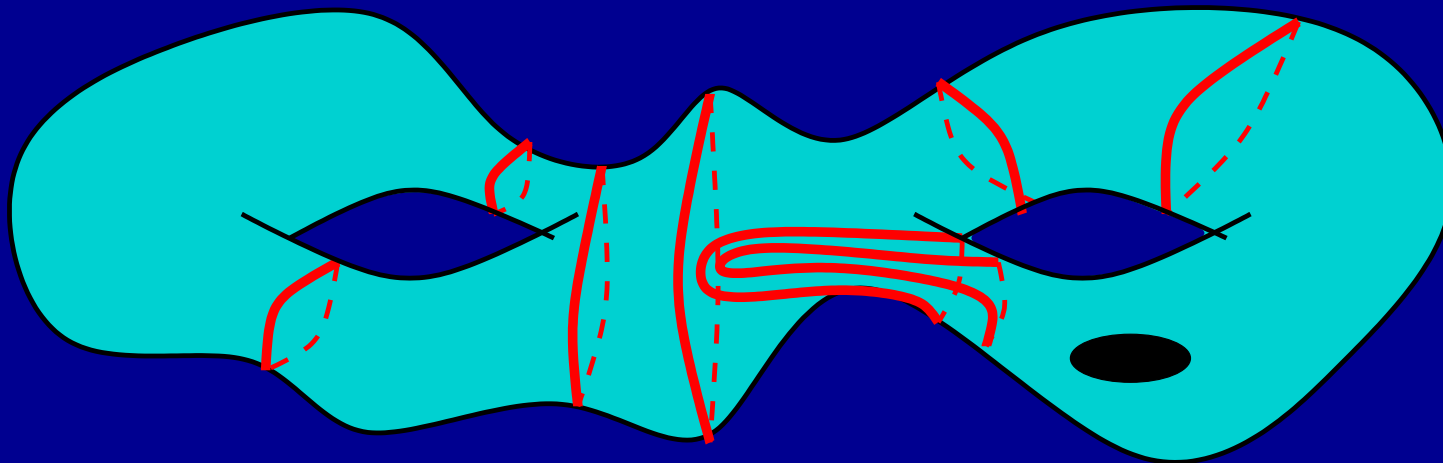
- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



$$s_{1\dots n} \xrightarrow{\text{complétion}} S_{1\dots N} \xrightarrow{\text{optimisation}} S'_{1\dots N} \xrightarrow{\text{extraction}} s'_{1\dots n}$$

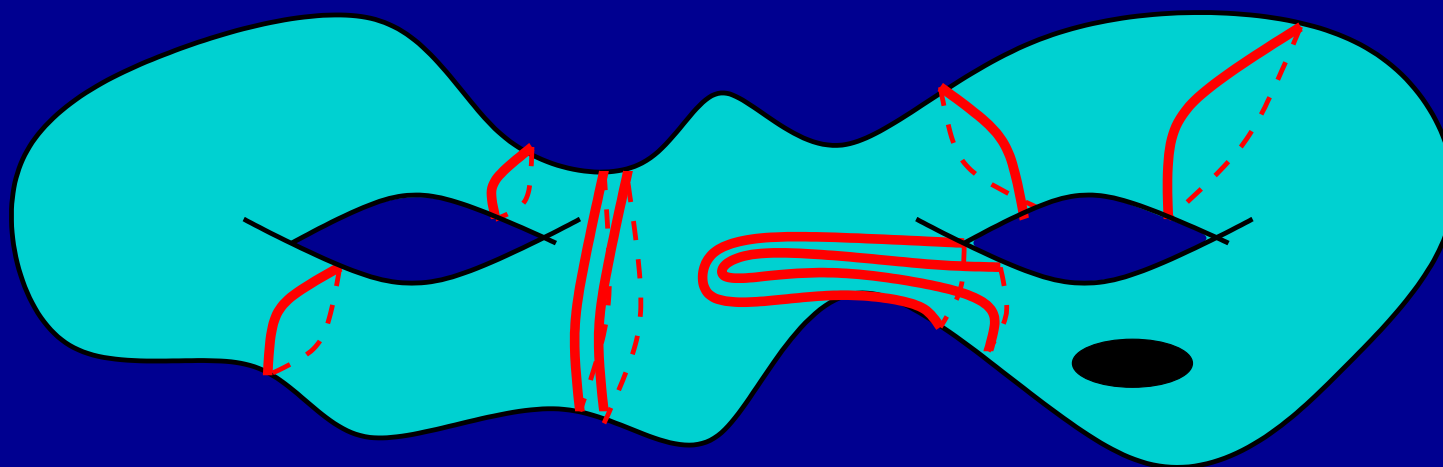
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



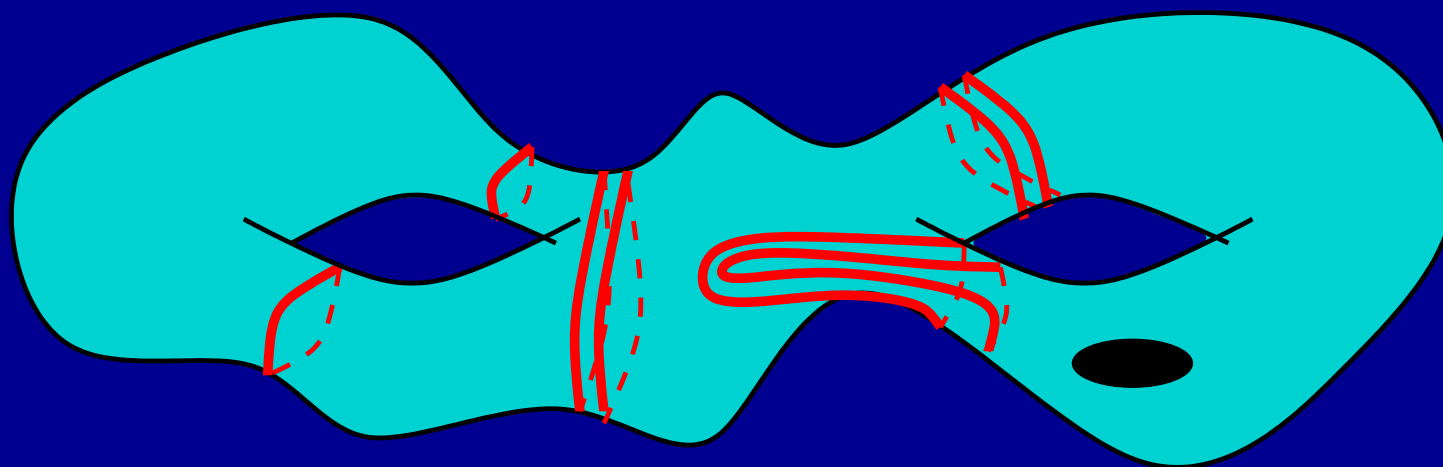
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



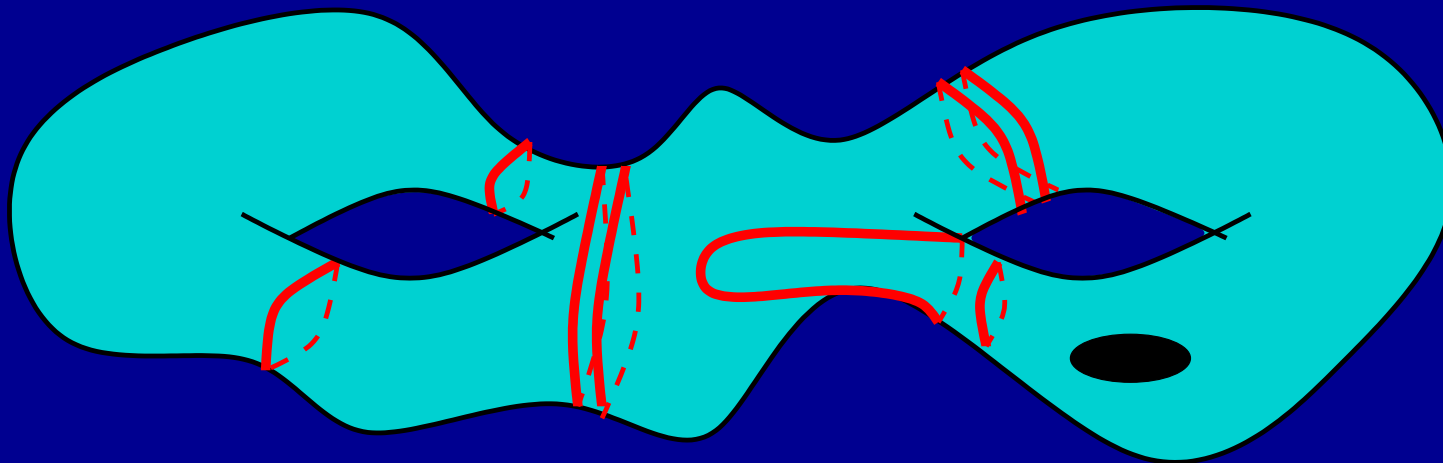
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



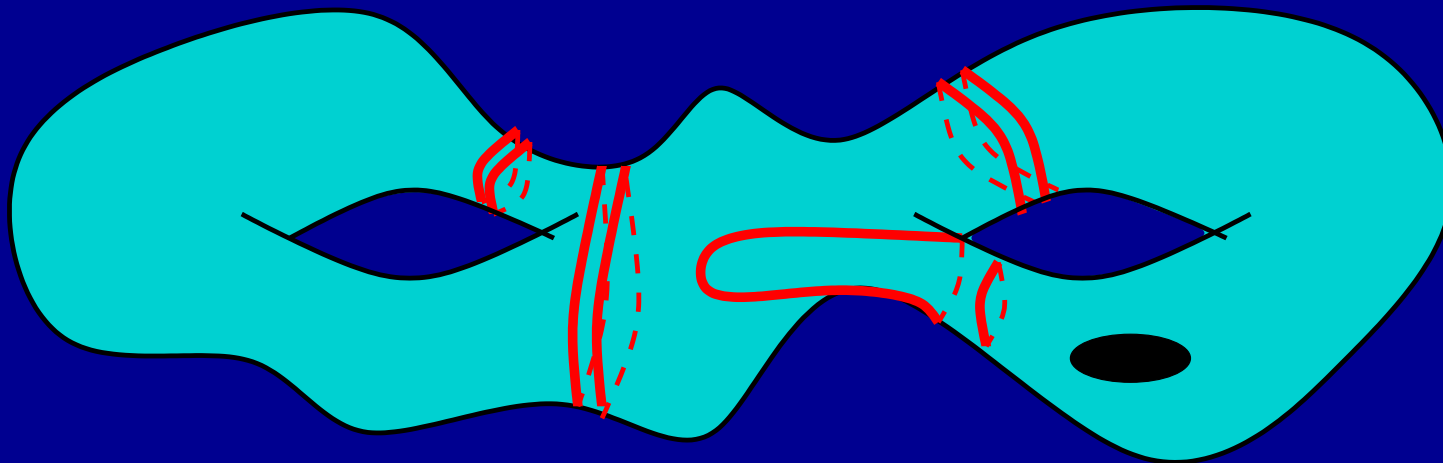
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



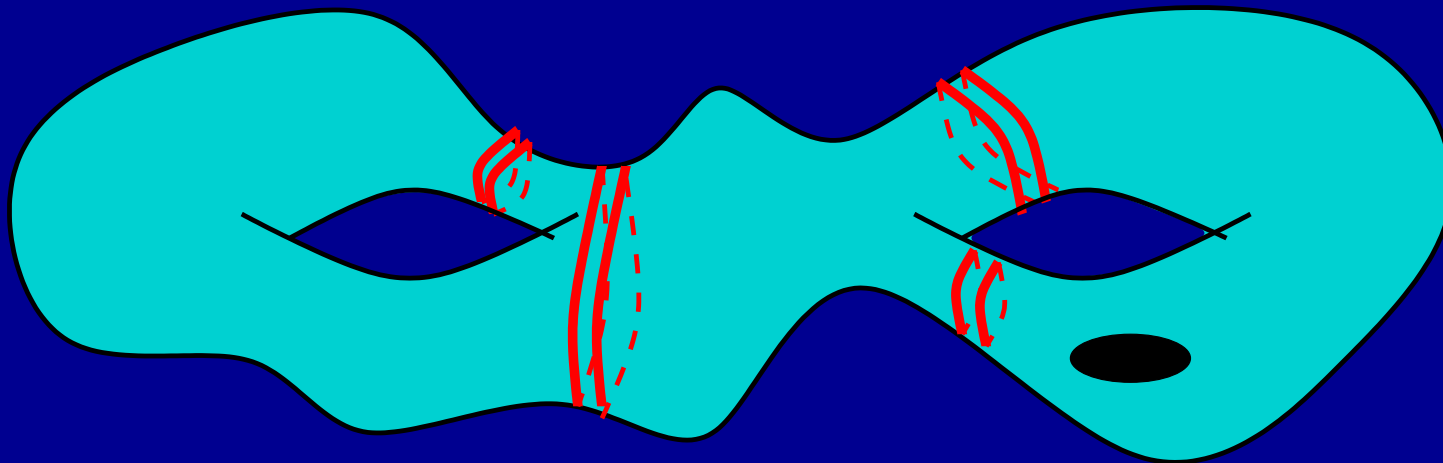
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



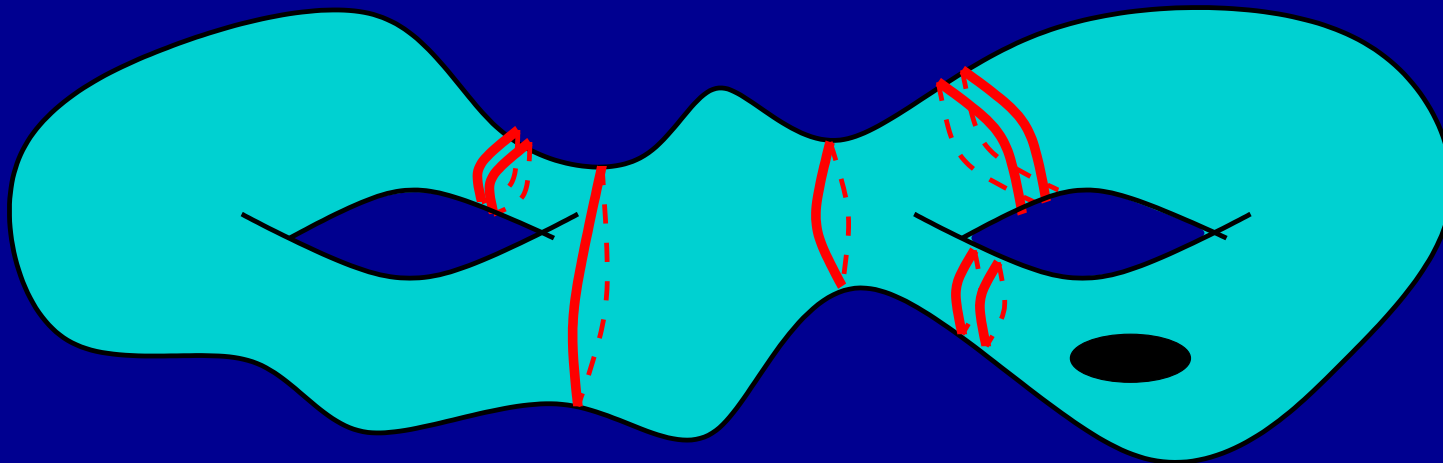
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



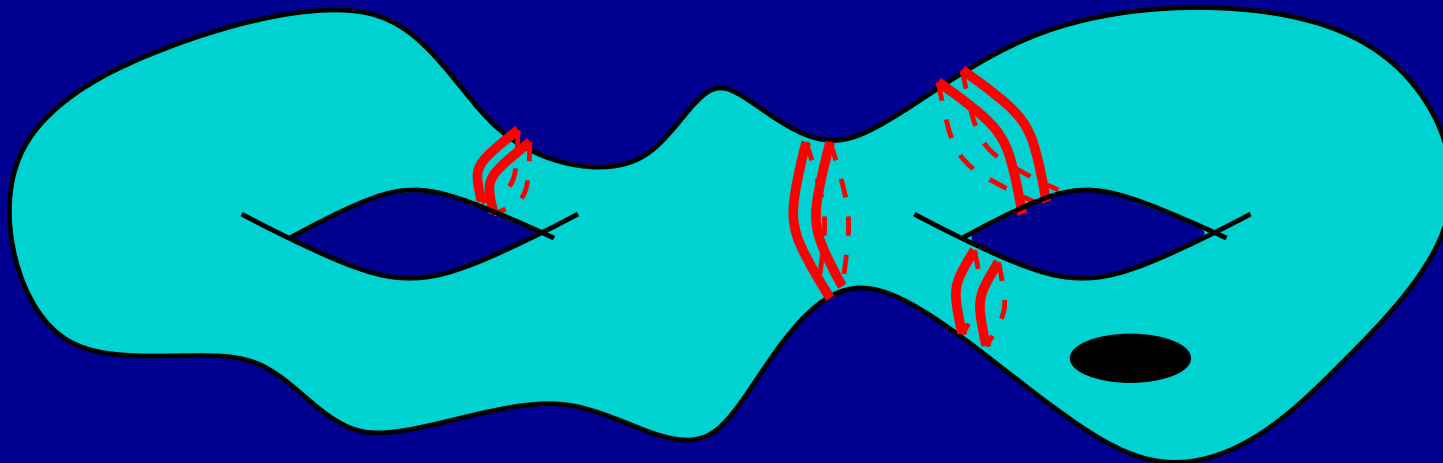
Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



Optimisation et extraction

- Étape élémentaire $\text{rac}_i(S)$: remplacer s_i par un plus court cycle simple homotope dans le cylindre ou pantalon de $S \setminus s_i$ dans lequel il se trouve.
- Étape principale $\text{rac} = \text{rac}_N \circ \text{rac}_{N-1} \circ \dots \circ \text{rac}_1$.
- Algorithme : itérer rac jusqu'à stabilité des longueurs.



Résultat

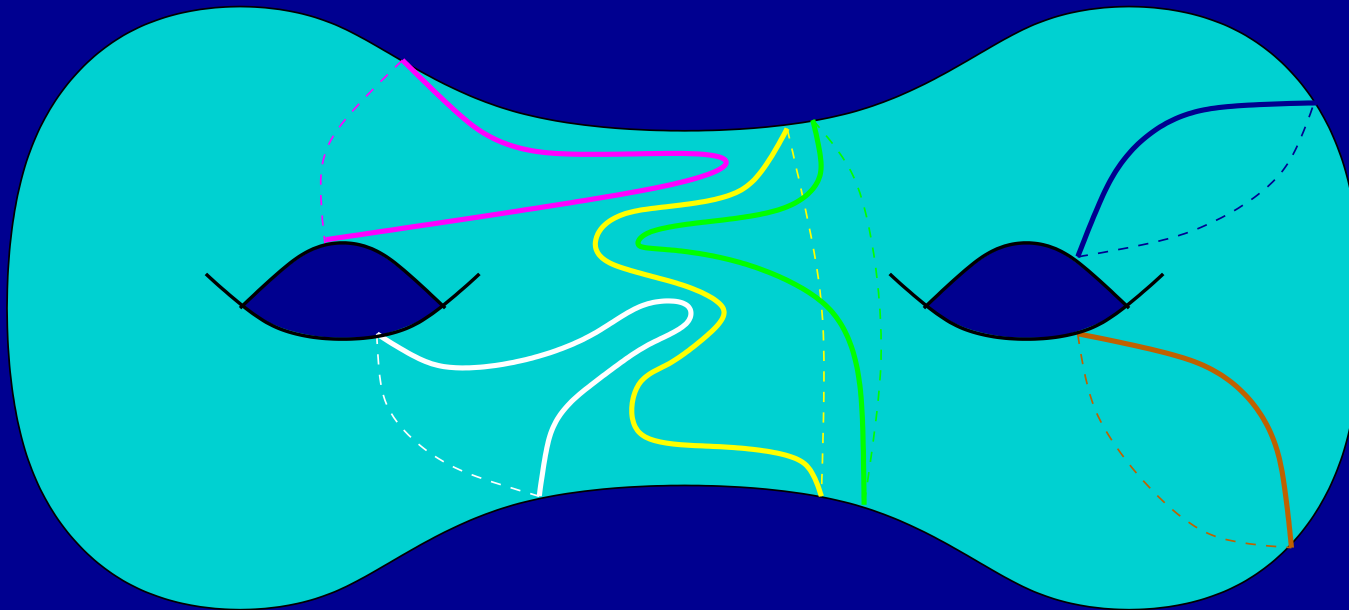
Théorème (III).

- $\exists m, |\text{rac}^{m+1}(S)| = |\text{rac}^m(S)|$;
- soit $S' = \text{rac}^m(S)$. Pour tout $i \in [1 \dots N]$, s'_i est un plus court cycle homotope à s_i .

Conséquence. On peut, pour tout i , remplacer s_i par un plus court cycle homotope s'_i de sorte que les s'_i soient simples et deux à deux disjoints.

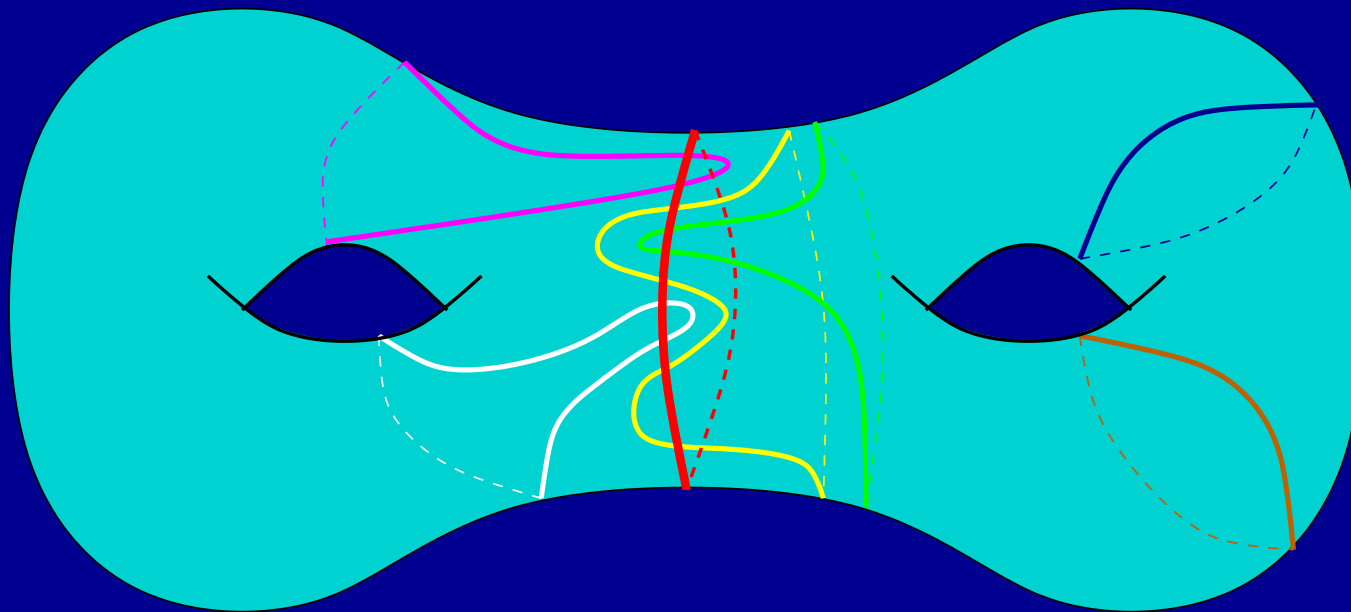
Idée naïve pour montrer (III)

- $i \in [1 \dots N]$;
- t_i homotope à s_i , aussi court que possible ; t_i n'est pas forcément simple ;



Idée naïve pour montrer (III)

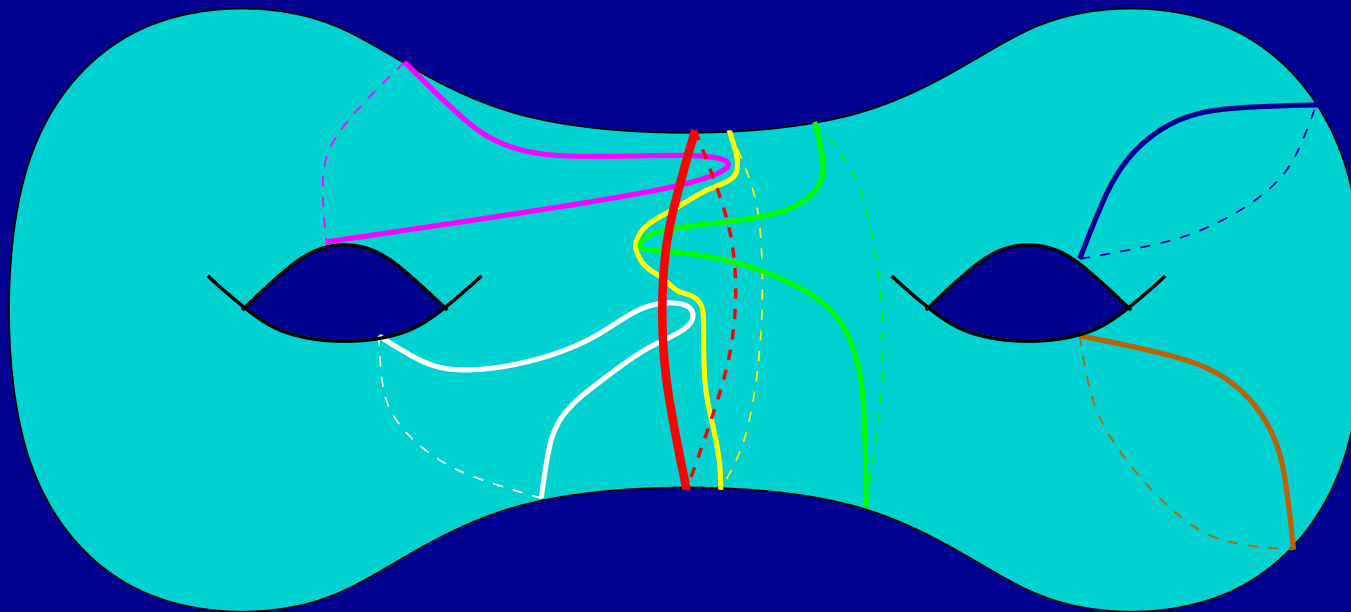
- $i \in [1 \dots N]$;
- t_i homotope à s_i , aussi court que possible ; t_i n'est pas forcément simple ;
- le mot des croisements S/t_i est parenthésé ;
- il décroît par suppression de $j\bar{j}$ ou $\bar{j}j$.



1 $\bar{1}$ 2 $\bar{2}$ 144 $\bar{1}$ 3 $\bar{3}$

Idée naïve pour montrer (III)

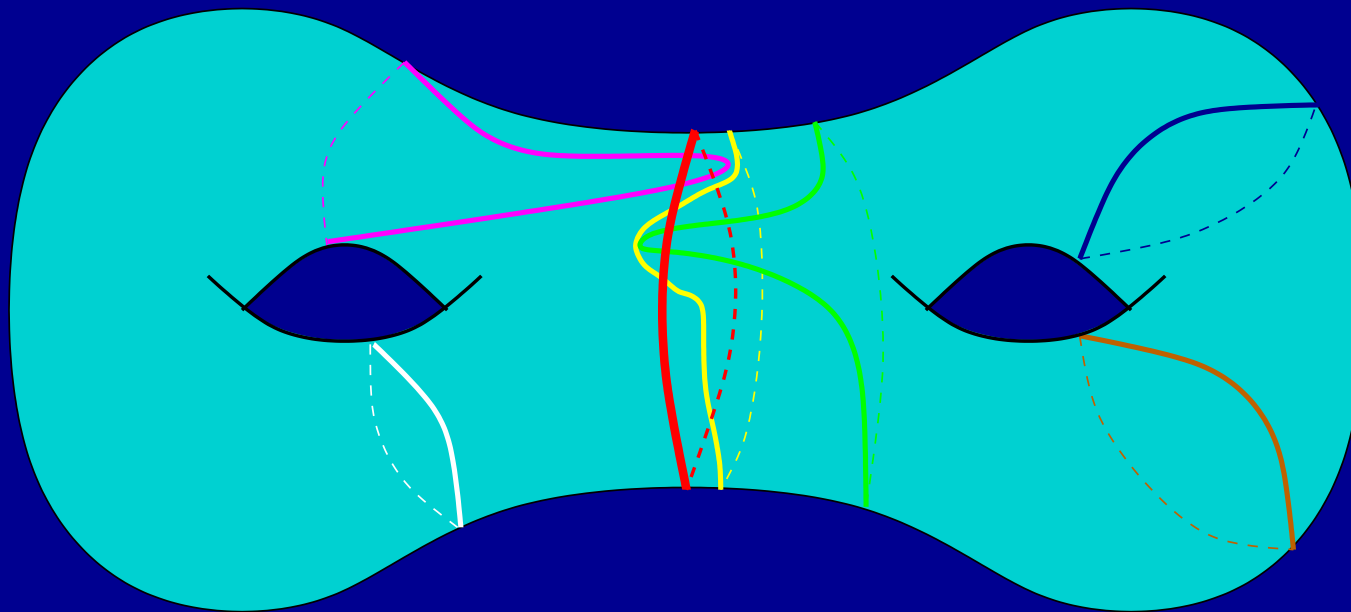
- $i \in [1 \dots N]$;
- t_i homotope à s_i , aussi court que possible ; t_i n'est pas forcément simple ;
- le mot des croisements S/t_i est parenthésé ;
- il décroît par suppression de $j\bar{j}$ ou $\bar{j}j$.



2 $\bar{2}$ 144 $\bar{1}$ 3 $\bar{3}$

Idée naïve pour montrer (III)

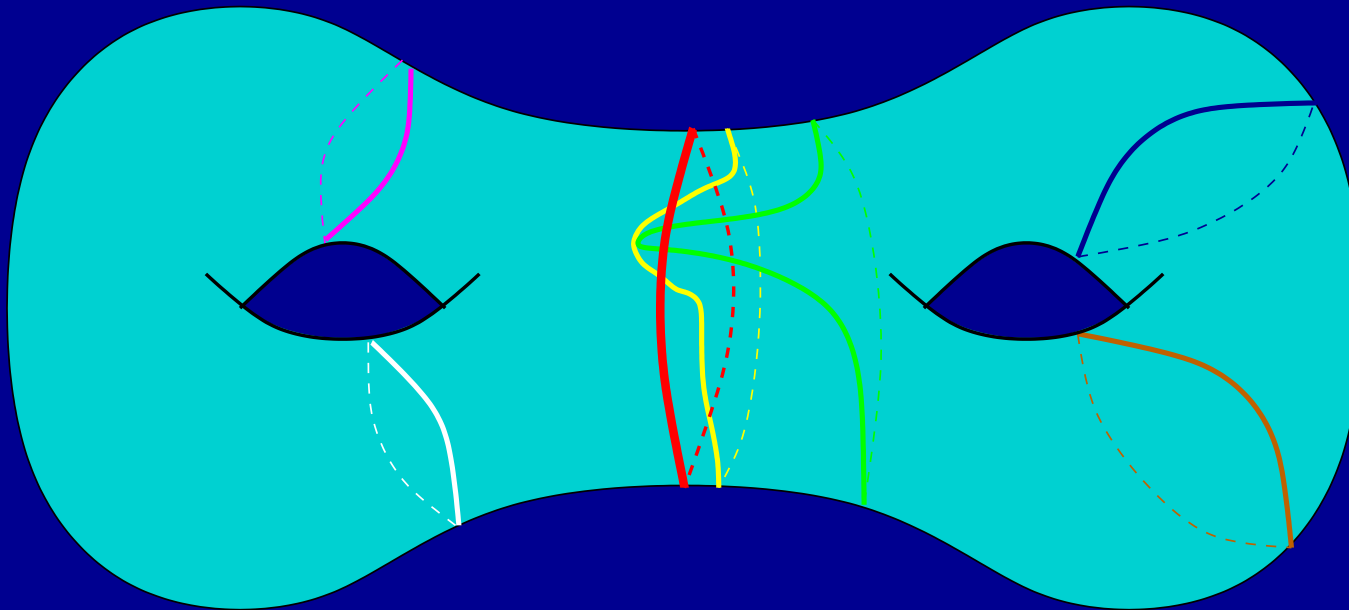
- $i \in [1 \dots N]$;
- t_i homotope à s_i , aussi court que possible ; t_i n'est pas forcément simple ;
- le mot des croisements S/t_i est parenthésé ;
- il décroît par suppression de $j\bar{j}$ ou $\bar{j}j$.



144133

Idée naïve pour montrer (III)

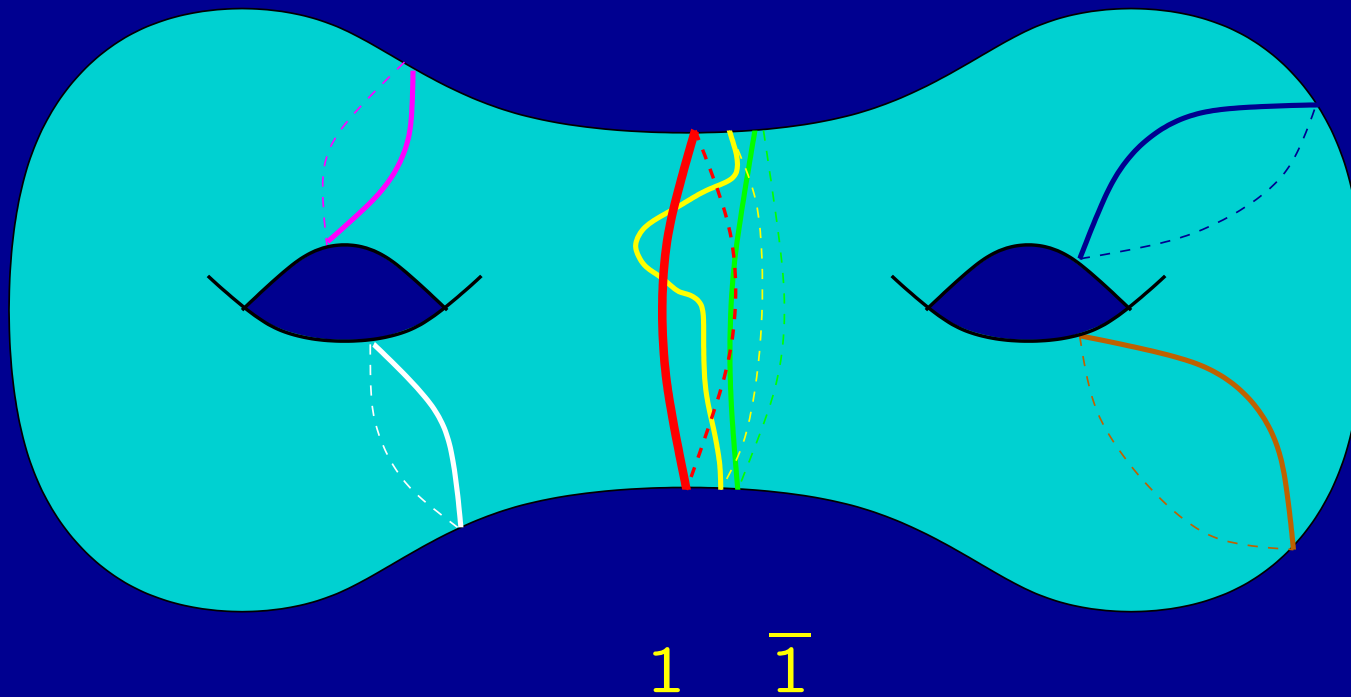
- $i \in [1 \dots N]$;
- t_i homotope à s_i , aussi court que possible ; t_i n'est pas forcément simple ;
- le mot des croisements S/t_i est parenthésé ;
- il décroît par suppression de $j\bar{j}$ ou $\bar{j}j$.



1441

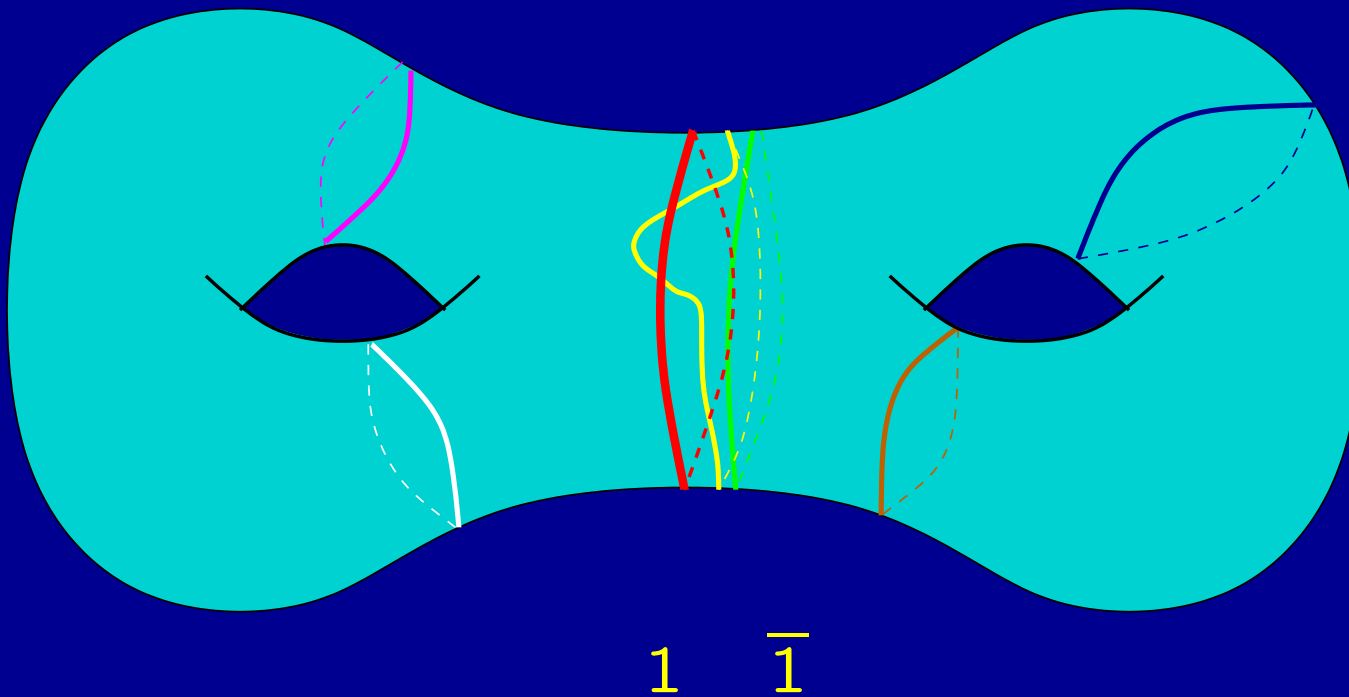
Idée naïve pour montrer (III)

- $i \in [1 \dots N]$;
- t_i homotope à s_i , aussi court que possible ; t_i n'est pas forcément simple ;
- le mot des croisements S/t_i est parenthésé ;
- il décroît par suppression de $j\bar{j}$ ou $\bar{j}j$.



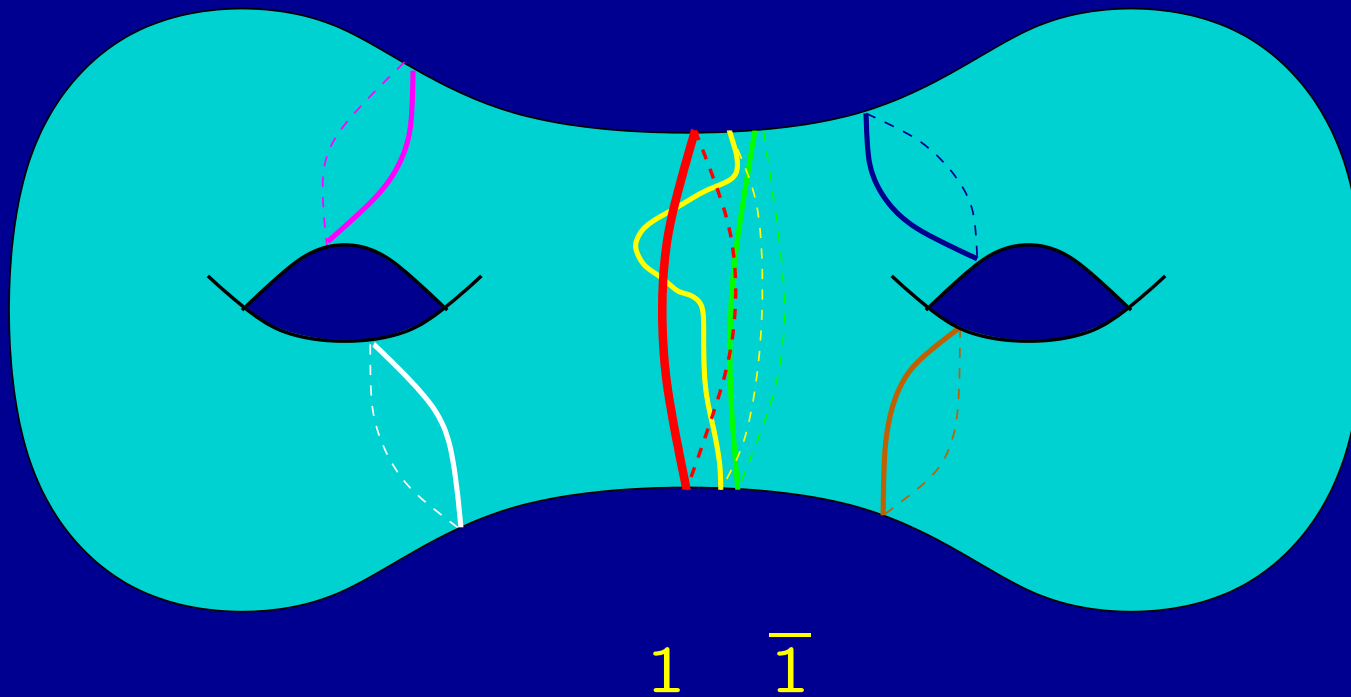
Idée naïve pour montrer (III)

- $i \in [1 \dots N]$;
- t_i homotope à s_i , aussi court que possible ; t_i n'est pas forcément simple ;
- le mot des croisements S/t_i est parenthésé ;
- il décroît par suppression de $j\bar{j}$ ou $\bar{j}j$.



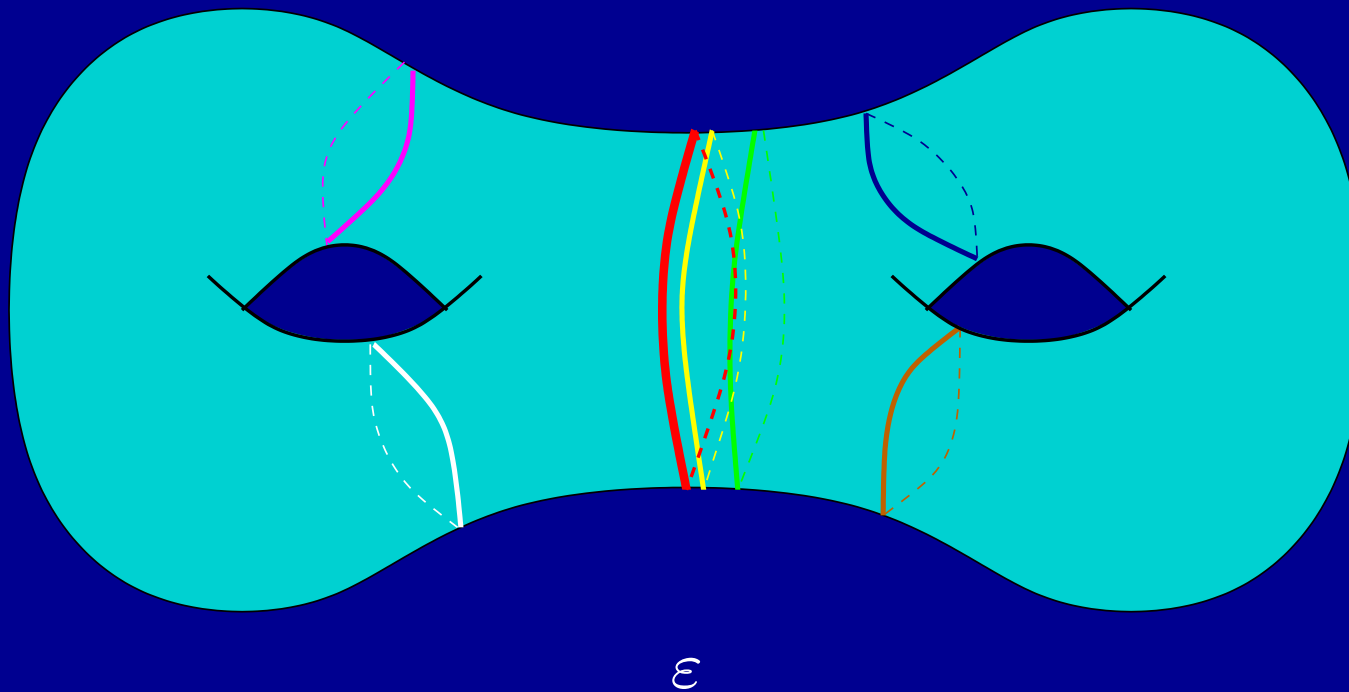
Idée naïve pour montrer (III)

- $i \in [1 \dots N]$;
- t_i homotope à s_i , aussi court que possible ; t_i n'est pas forcément simple ;
- le mot des croisements S/t_i est parenthésé ;
- il décroît par suppression de $j\bar{j}$ ou $\bar{j}j$.



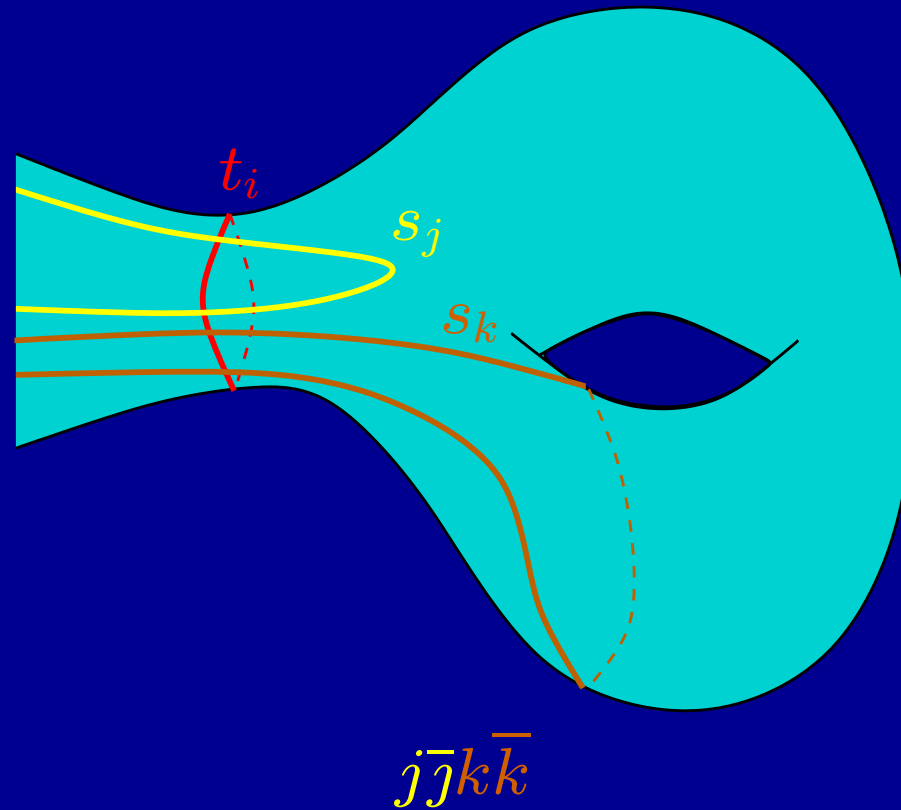
Idée naïve pour montrer (III)

- $i \in [1 \dots N]$;
- t_i homotope à s_i , aussi court que possible ; t_i n'est pas forcément simple ;
- le mot des croisements S/t_i est parenthésé ;
- il décroît par suppression de $j\bar{j}$ ou $\bar{j}j$.



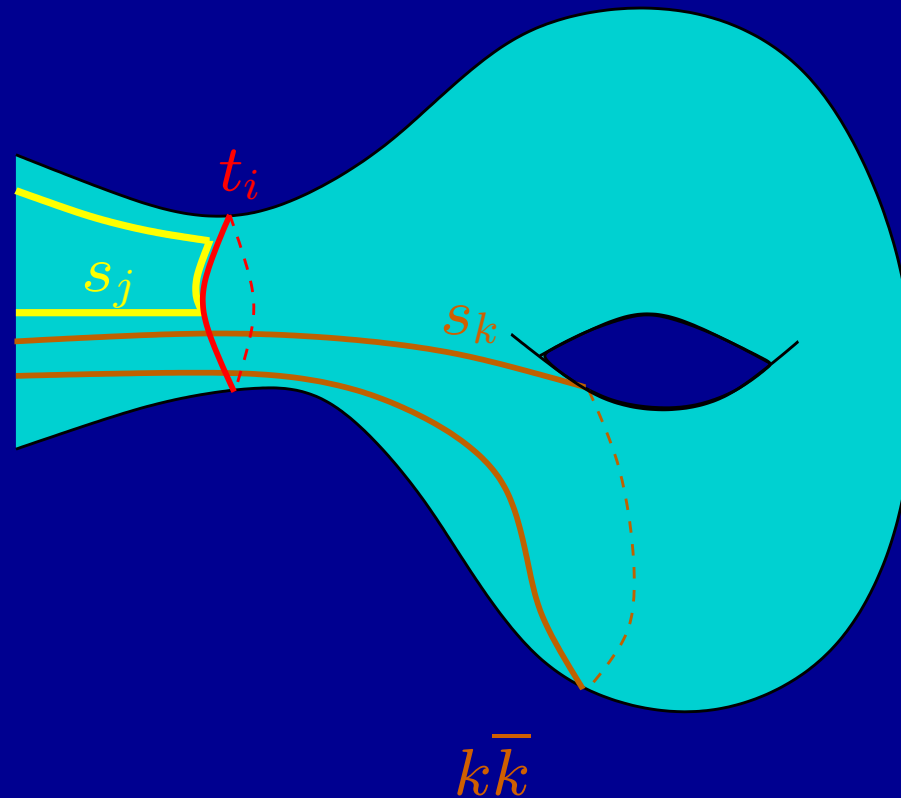
En fait...

... tout est plus compliqué.



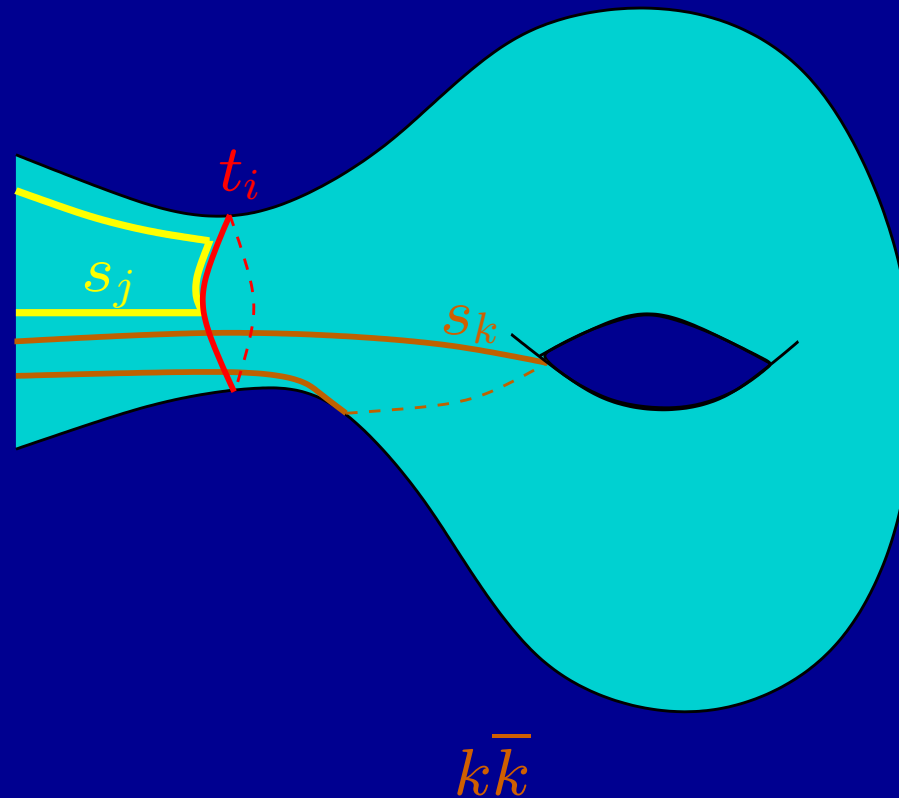
En fait...

... tout est plus compliqué.



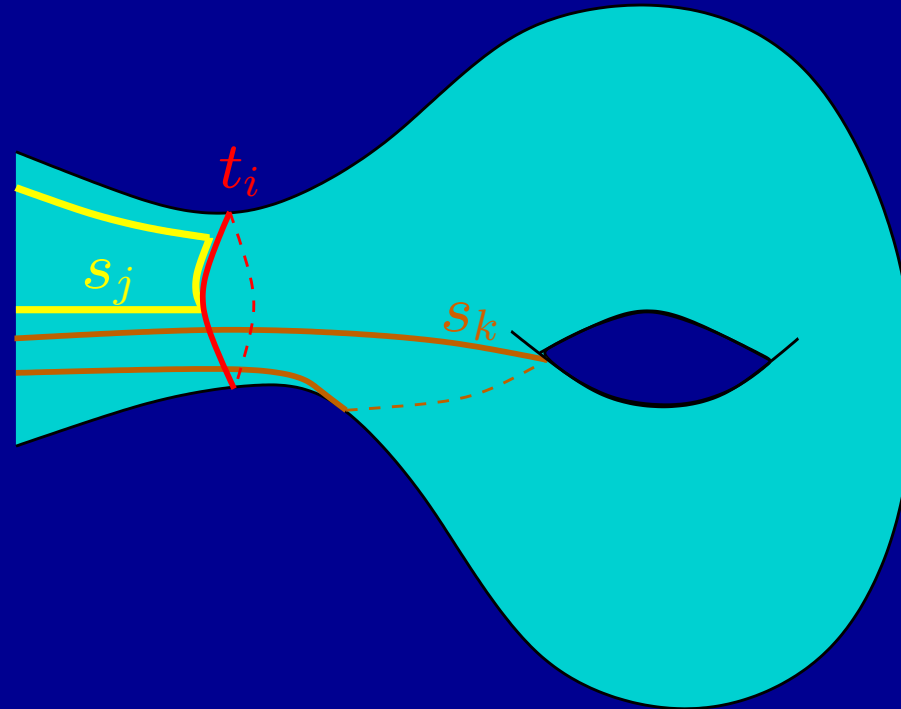
En fait...

... tout est plus compliqué.



En fait...

... tout est plus compliqué.



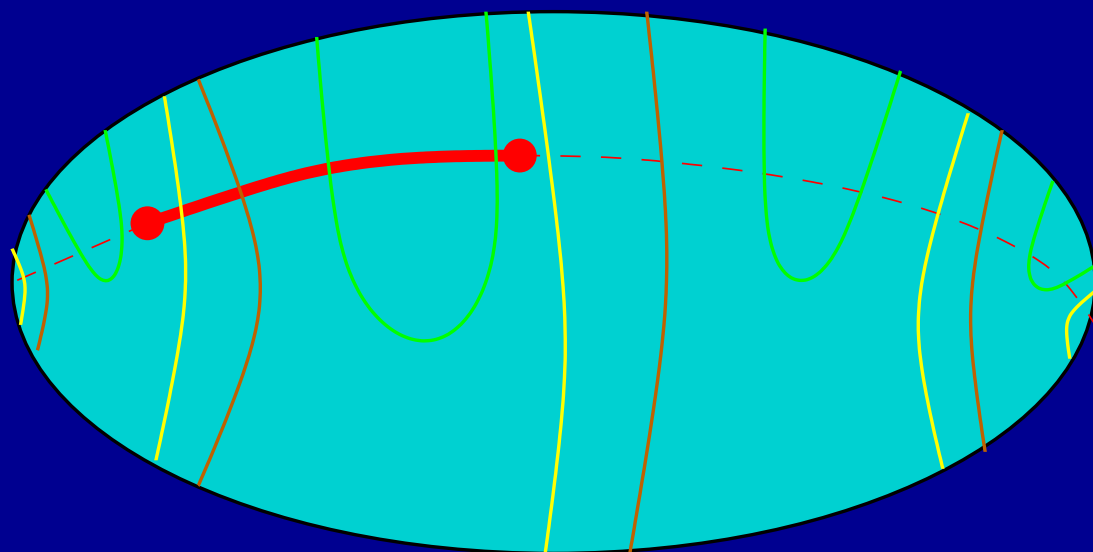
Il faut distinguer selon les classes d'homotopie.

$$j^\alpha \bar{j}^\alpha k^\beta \bar{k}^\gamma$$

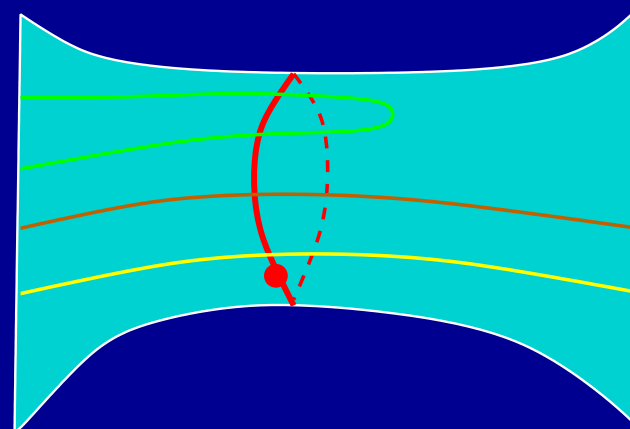
Outils pour montrer (III)

Mots des croisements dans le **revêtement universel** de la surface : $[S/\tilde{t}_i]$ est l'ensemble des mots des croisements entre relevés de S et « périodes » d'un relevé \tilde{t}_i de t_i .

$$j^\alpha . w \simeq w . j^{\tau(\alpha)}$$



$$j^\alpha k^\beta \ell^\gamma \bar{\ell}^\gamma$$

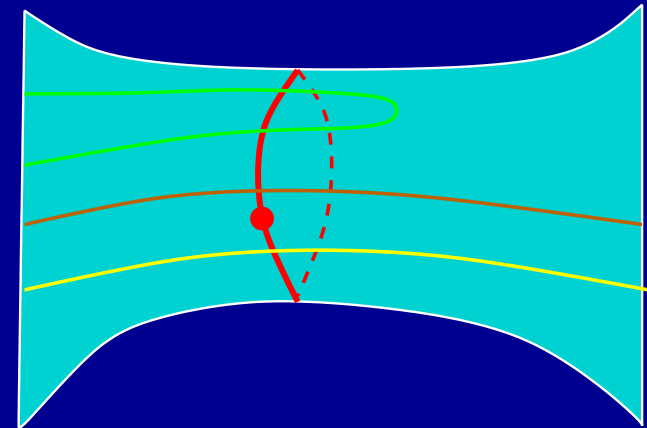
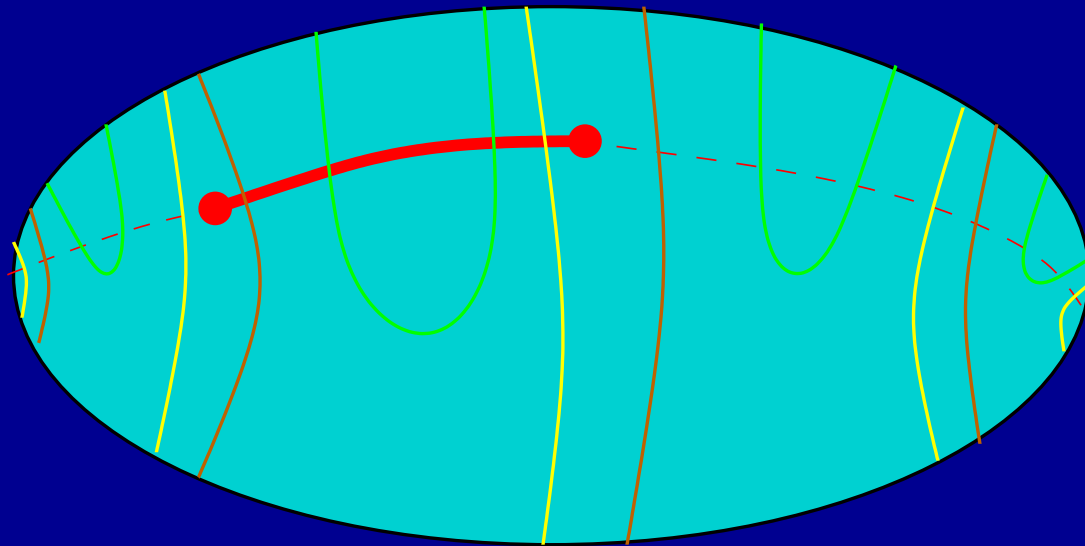


$$\text{red}_j : w . j^\alpha \bar{j}^\alpha . w' \rightarrow w . w', \quad j^\alpha . w . \bar{j}^{\tau(\alpha)} \rightarrow w.$$

Outils pour montrer (III)

Mots des croisements dans le **revêtement universel** de la surface : $[S/\tilde{t}_i]$ est l'ensemble des mots des croisements entre relevés de S et « périodes » d'un relevé \tilde{t}_i de t_i .

$$j^\alpha . w \simeq w . j^{\tau(\alpha)}$$



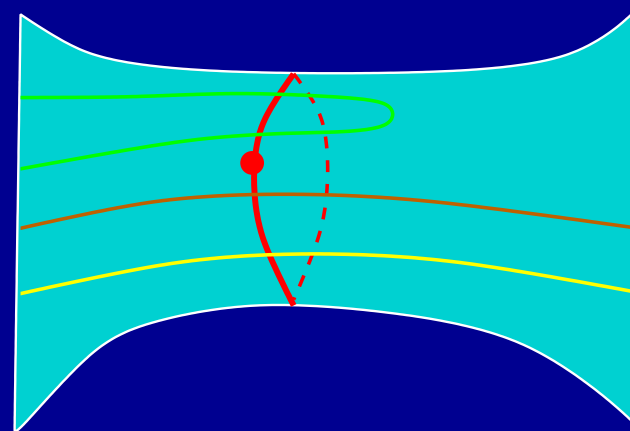
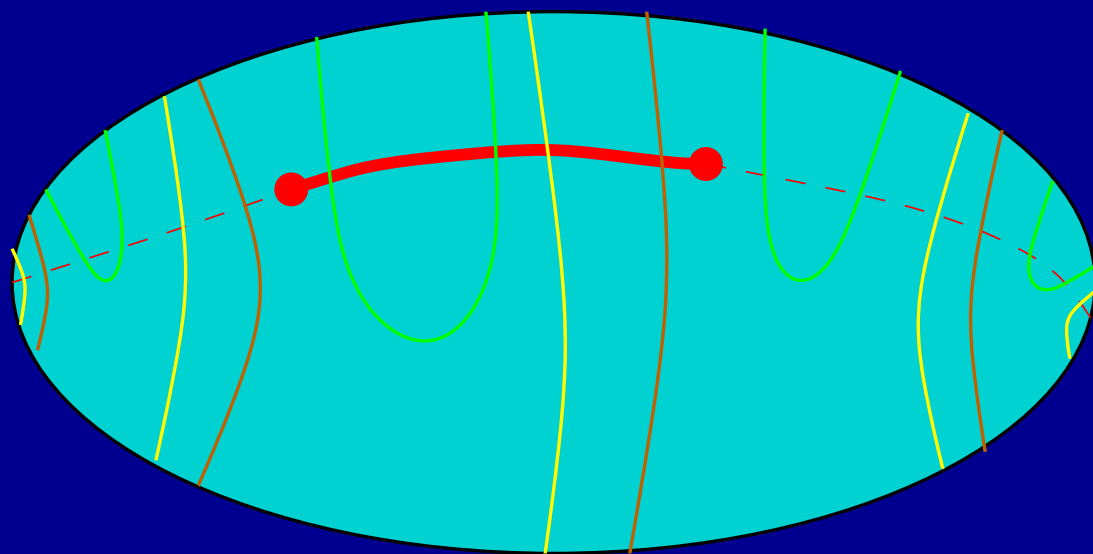
$$k^\beta \ell^\gamma \bar{\ell}^\gamma j^{\tau(\alpha)}$$

$$\text{red}_j : w . j^\alpha \bar{j}^\alpha . w' \rightarrow w . w', \quad j^\alpha . w . \bar{j}^{\tau(\alpha)} \rightarrow w.$$

Outils pour montrer (III)

Mots des croisements dans le **revêtement universel** de la surface : $[S/\tilde{t}_i]$ est l'ensemble des mots des croisements entre relevés de S et « périodes » d'un relevé \tilde{t}_i de t_i .

$$j^\alpha . w \simeq w . j^{\tau(\alpha)}$$



$$l^\gamma \bar{l}^\gamma j^{\tau(\alpha)} k^{\tau(\beta)}$$

$$\text{red}_j : w . j^\alpha \bar{j}^\alpha . w' \rightarrow w . w', \quad j^\alpha . w . \bar{j}^{\tau(\alpha)} \rightarrow w.$$

4. Complexité

Complexité

Algorithmiquement :

- optimisation : calculs de plus courts chemins dans un graphe planaire [Dijkstra], [Henzinger et al, 1997] ;
- complétion (*non présenté ici*).

Si tous les poids sont **unitaires** :

- algorithmes **polynomiaux** en :
 - ◆ la complexité de la surface ;
 - ◆ la complexité du plongement d'entrée ;
- en particulier :
 - ◆ problème NP-difficile : calcul du plus court schéma polygonal [Erickson, Har-Peled, 2002] ;
 - ◆ problème polynomial : calcul d'un plus court système fondamental de lacets d'une classe d'homotopie [nous].

Complexité, suite et fin

Analyse dans le cas général :

■ pour les graphes : $O(\alpha^2 d K^2 \mu^3 (n + g + b)^2)$;

■ pour les cycles : $O(\alpha^3 d K^3 \mu^4 (n + g + b)^2)$,
avec

■ complexité de la **surface** :

◆ d : degré maximal d'un sommet ;

◆ K : nombre de sommets, arêtes et faces ;

◆ g, b : nombre de « poignées » et de bords ;

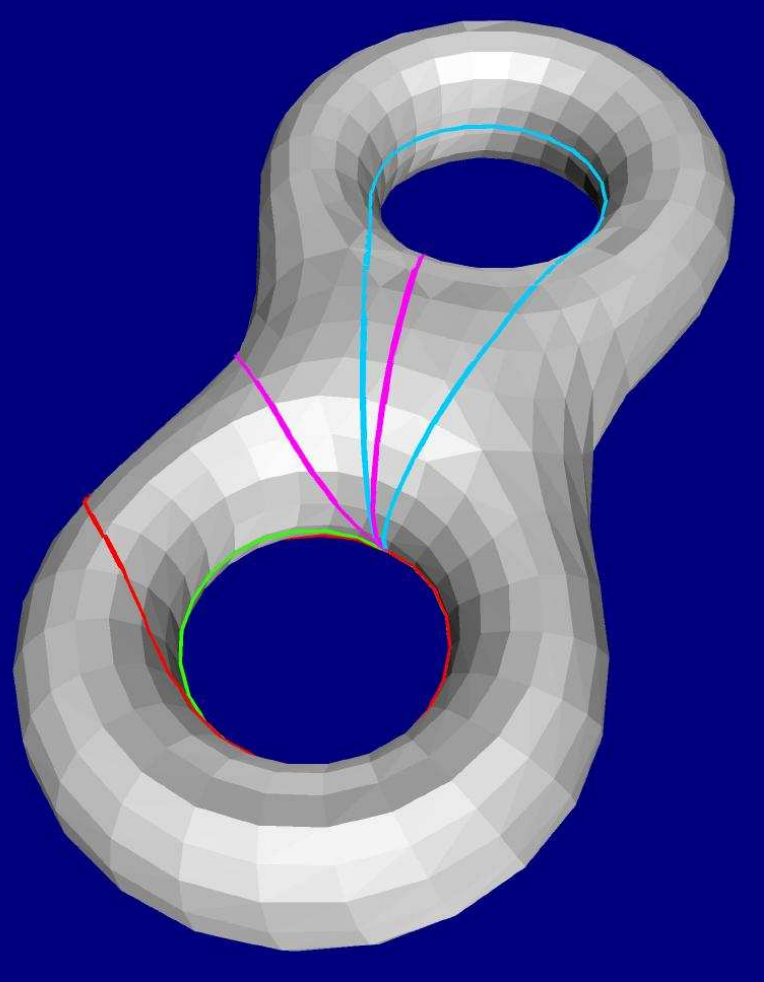
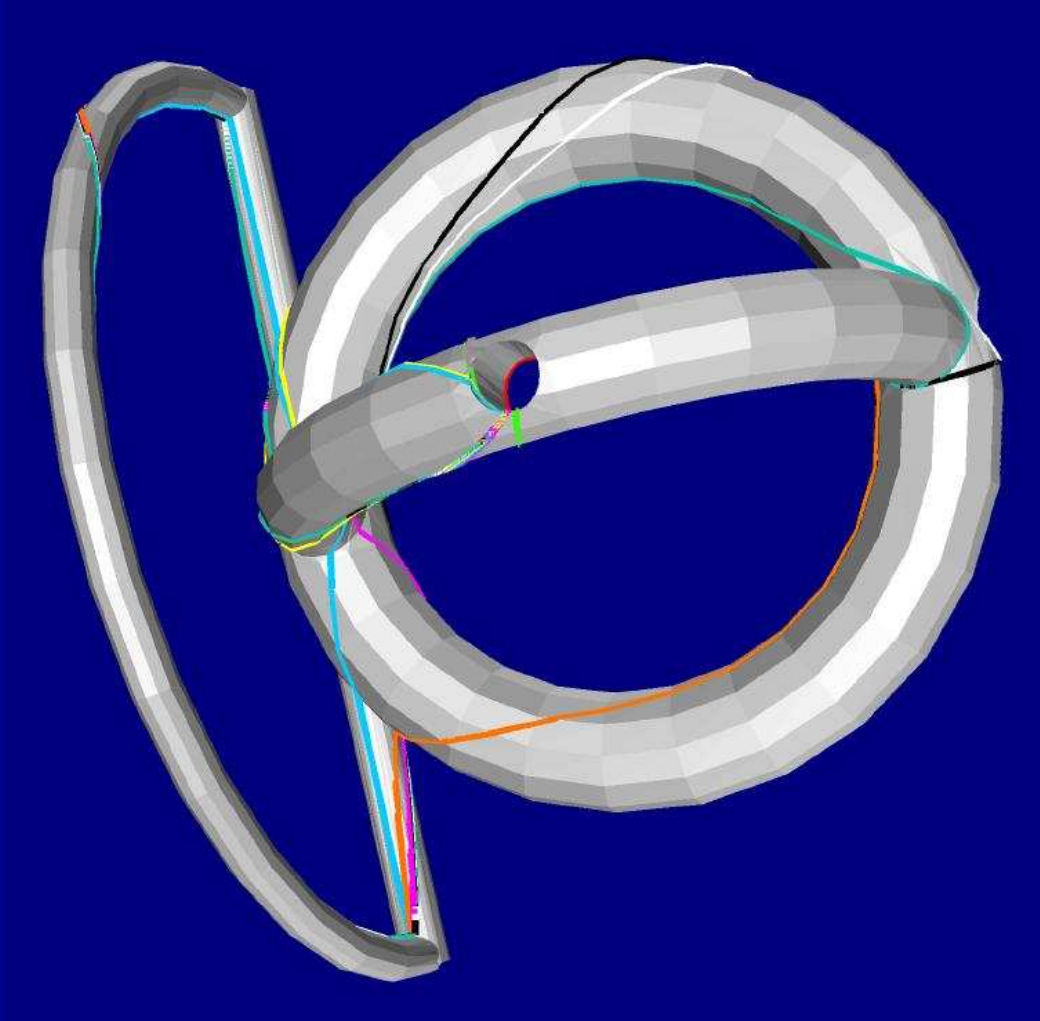
■ complexité des **courbes d'entrée** :

◆ μ : nombre maximal de passages d'une courbe d'entrée
par une arête du graphe ;

◆ n : nombre de courbes d'entrée ;

■ α : $\frac{\text{plus longue arête}}{\text{plus courte arête}}$ paramètre « **géométrique** ».

Conclusion



Prolongements possibles

- Implantation générique complète ; utilisation en pratique ;

Prolongements possibles

- Implantation générique complète ; utilisation en pratique ;
- analyse de complexité (α , borne inférieure) ;

Prolongements possibles

- Implantation générique complète ; utilisation en pratique ;
- analyse de complexité (α , borne inférieure) ;
- extension aux surfaces non orientables ;

Prolongements possibles

- Implantation générique complète ; utilisation en pratique ;
- analyse de complexité (α , borne inférieure) ;
- extension aux surfaces non orientables ;
- extension aux surfaces polyédrales, riemanniennes ;

Prolongements possibles

- Implantation générique complète ; utilisation en pratique ;
- analyse de complexité (α , borne inférieure) ;
- extension aux surfaces non orientables ;
- extension aux surfaces polyédrales, riemanniennes ;
- courbes avec intersections ;

Prolongements possibles

- Implantation générique complète ; utilisation en pratique ;
- analyse de complexité (α , borne inférieure) ;
- extension aux surfaces non orientables ;
- extension aux surfaces polyédrales, riemanniennes ;
- courbes avec intersections ;
- sommets du graphe non fixés ;

Prolongements possibles

- Implantation générique complète ; utilisation en pratique ;
- analyse de complexité (α , borne inférieure) ;
- extension aux surfaces non orientables ;
- extension aux surfaces polyédrales, riemanniennes ;
- courbes avec intersections ;
- sommets du graphe non fixés ;
- plus court plongement sans fixer la classe d'homotopie :
 - ◆ d'une décomposition en pantalons ;
 - ◆ d'un système fondamental de lacets ;
 - ◆ d'un graphe donné ;

Prolongements possibles

- Implantation générique complète ; utilisation en pratique ;
- analyse de complexité (α , borne inférieure) ;
- extension aux surfaces non orientables ;
- extension aux surfaces polyédrales, riemanniennes ;
- courbes avec intersections ;
- sommets du graphe non fixés ;
- plus court plongement sans fixer la classe d'homotopie :
 - ◆ d'une décomposition en pantalons ;
 - ◆ d'un système fondamental de lacets ;
 - ◆ d'un graphe donné ;
- surface plongée en 3D : choix plus naturels ?

Prolongements possibles

- Implantation générique complète ; utilisation en pratique ;
- analyse de complexité (α , borne inférieure) ;
- extension aux surfaces non orientables ;
- extension aux surfaces polyédrales, riemanniennes ;
- courbes avec intersections ;
- sommets du graphe non fixés ;
- plus court plongement sans fixer la classe d'homotopie :
 - ◆ d'une décomposition en pantalons ;
 - ◆ d'un système fondamental de lacets ;
 - ◆ d'un graphe donné ;
- surface plongée en 3D : choix plus naturels ?
- dimension supérieure ?