Comparaison d'arborescences non-ordonnées étiquetées

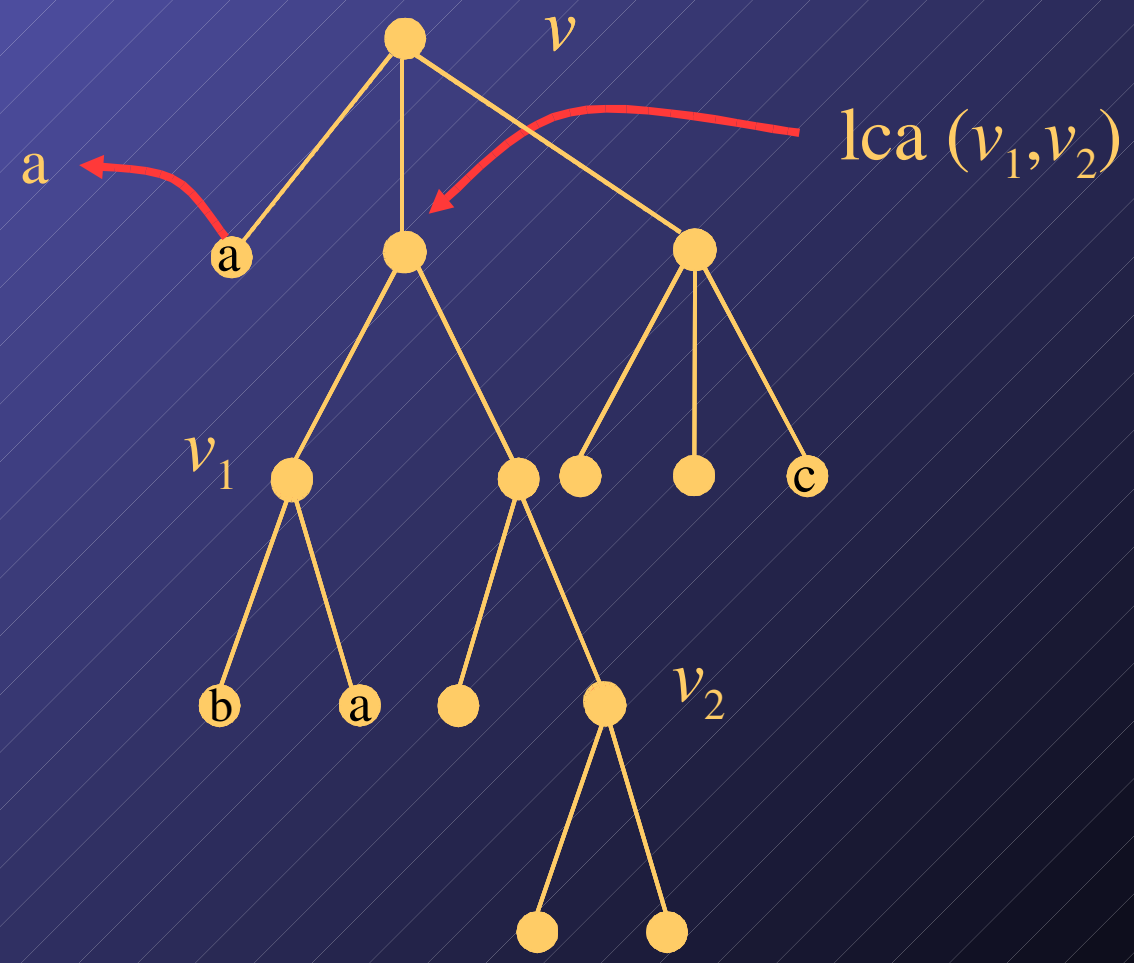
Définition d'une distance entre architecture de plantes

- Première approximation :
  $\Rightarrow$ Distance entre arborescences

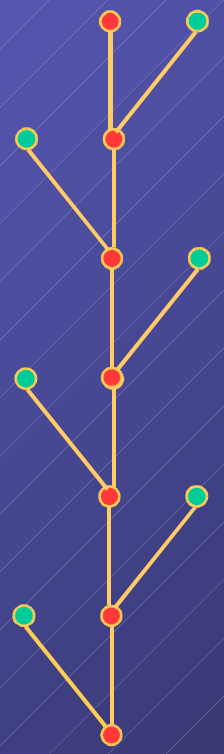
- Amélioration :
  $\Rightarrow$ Distance entre arborescences multi-échelles

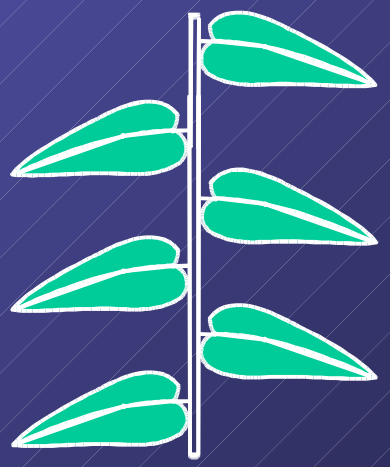
Définitions

Forêt $F[v]$ enracinée en v



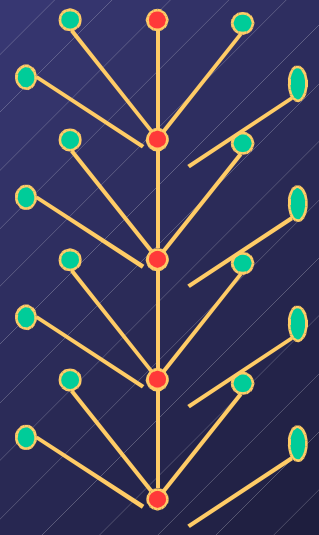
Description de la plante

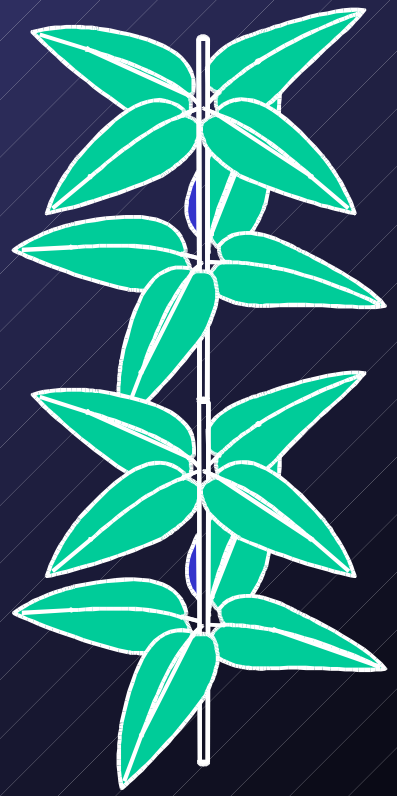




1 feuille par nœud

Arborescences





Plusieurs feuilles par nœud

non-ordonnées

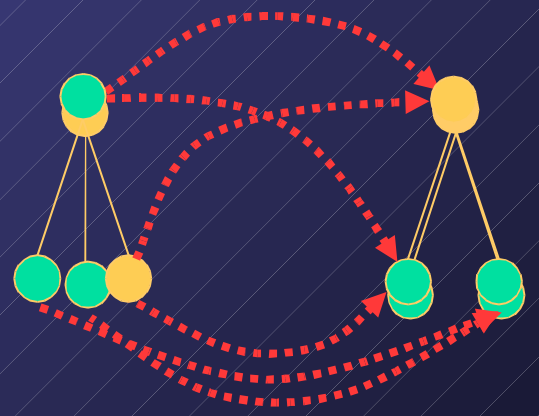
Alignement entre arborescences

Un alignement valide M est un ensemble de paires ordonnées de vertex des arborescences T_1 et T_2 vérifiant :

$$\forall (x_1, x_2), (y_1, y_2) \in M :$$

$$x_1 = y_1 \Leftrightarrow x_2 = y_2$$

$$x_1 \leq y_1 \Leftrightarrow x_2 \leq y_2$$



Alignement entre arborescences

Un alignement valide M est un ensemble de paires ordonnées de vertex des arborescences T_1 et T_2 vérifiant :

$$\forall (x_1, x_2), (y_1, y_2) \in M :$$

$$x_1 = y_1 \Leftrightarrow x_2 = y_2$$

$$x_1 \leq y_1 \Leftrightarrow x_2 \leq y_2$$

$$\Rightarrow d(M) = \underbrace{\sum_{(x,y) \in M} d(x,y)}_{\text{coût de substitution}} + \underbrace{\sum_{x \notin M} d(x, \quad)}_{\text{coût d'élision et d'insertion}}$$

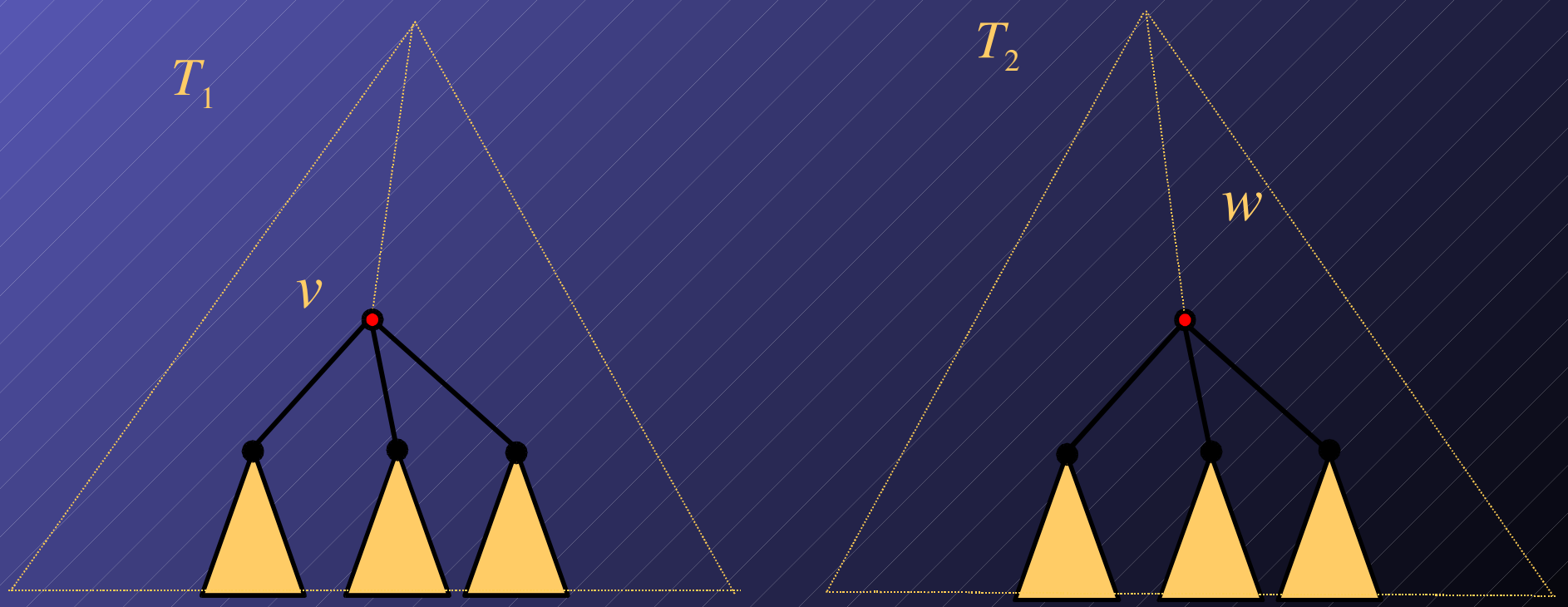
Distance d'édition entre arborescences

$$D(T_1,T_2) = \min_{M \in \mathcal{M}} \left\{ d(M) \right\}$$

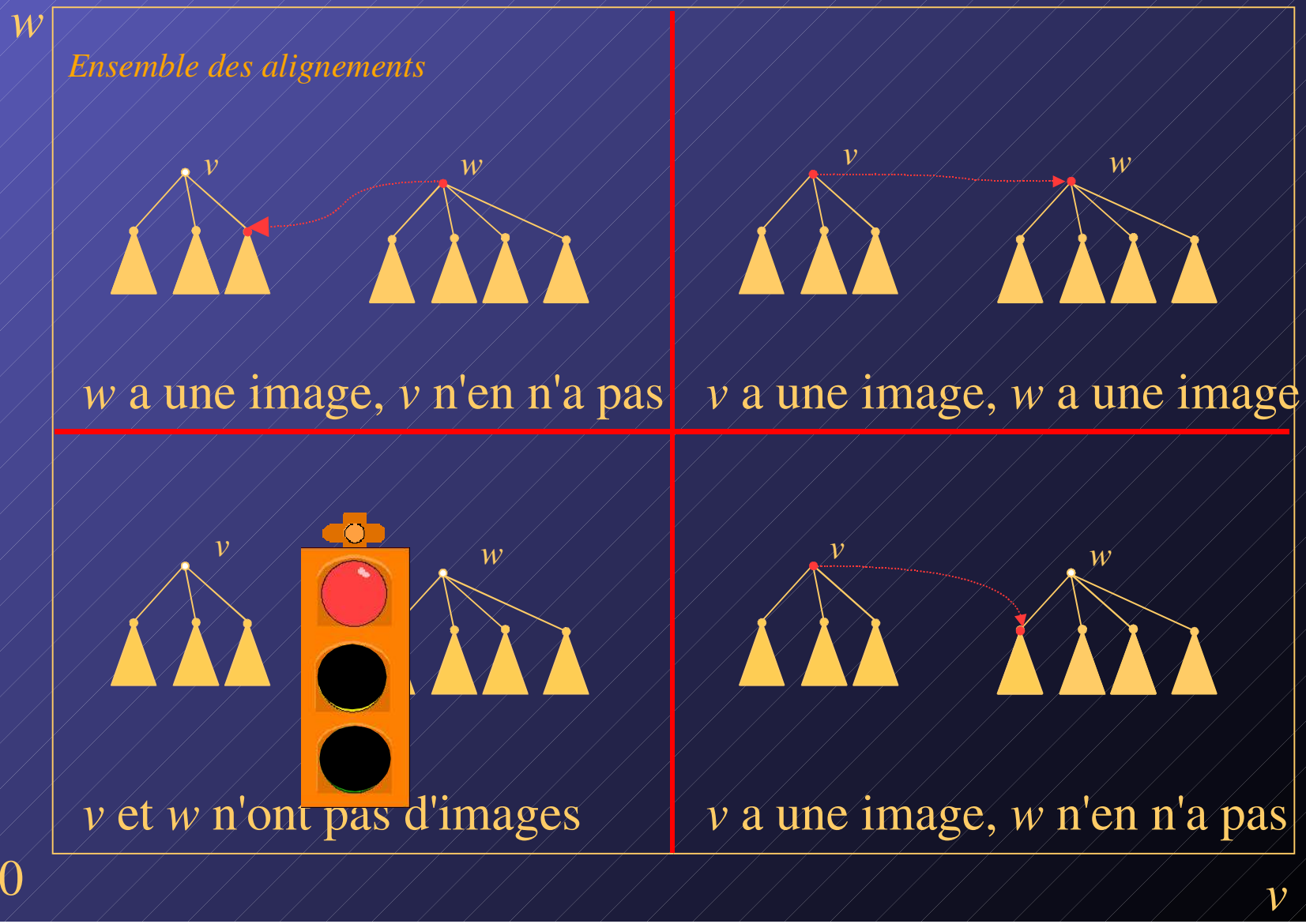
$\Rightarrow$ D est une distance

$\Rightarrow$ Recherche d'un algorithme s'appuyant sur le principe de programmation dynamique (optimisation récursive)

Principe des algorithmes de comparaison



Partition de l'ensemble des alignements



Calcul récursif de l'alignement optimal

$$D(T_1[v],T_2[w]) = \min \begin{cases} d(v,w) + D(F_1[v],F_2[w]) \\ D(\Theta,T_2[w]) + \min \left\{ D(T_1[v],T_2[w_k]) - D(\Theta,T_2[w_k]) \right\} \\ D(T_1[v],\Theta) + \min \left\{ D(T_1[v_k],T_2[w]) - D(T_1[v_k],\Theta) \right\} \end{cases}$$

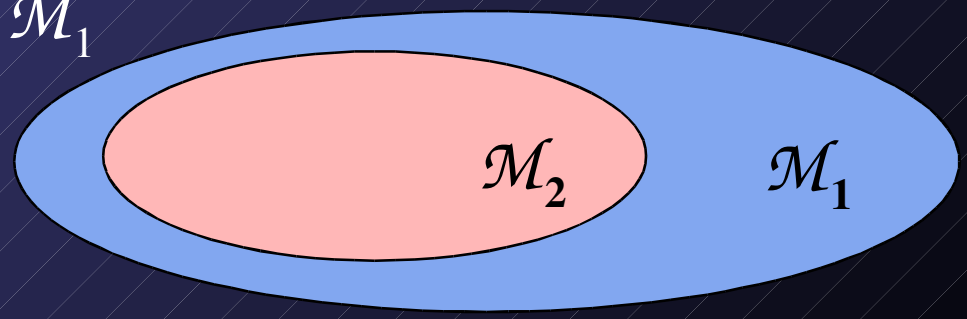
Calcul récursif de l'alignement optimal

$$D(T_1[v],T_2[w]) = \min \begin{cases} d(v,w) + \boldsymbol{D(F_1[v],F_2[w])} \\ D(\Theta,T_2[w]) + \min \left\{ D(T_1[v],T_2[w_k]) - D(\Theta,T_2[w_k]) \right\} \\ D(T_1[v],\Theta) + \min \left\{ D(T_1[v_k],T_2[w]) - D(T_1[v_k],\Theta) \right\} \end{cases}$$

Distance entre forêts

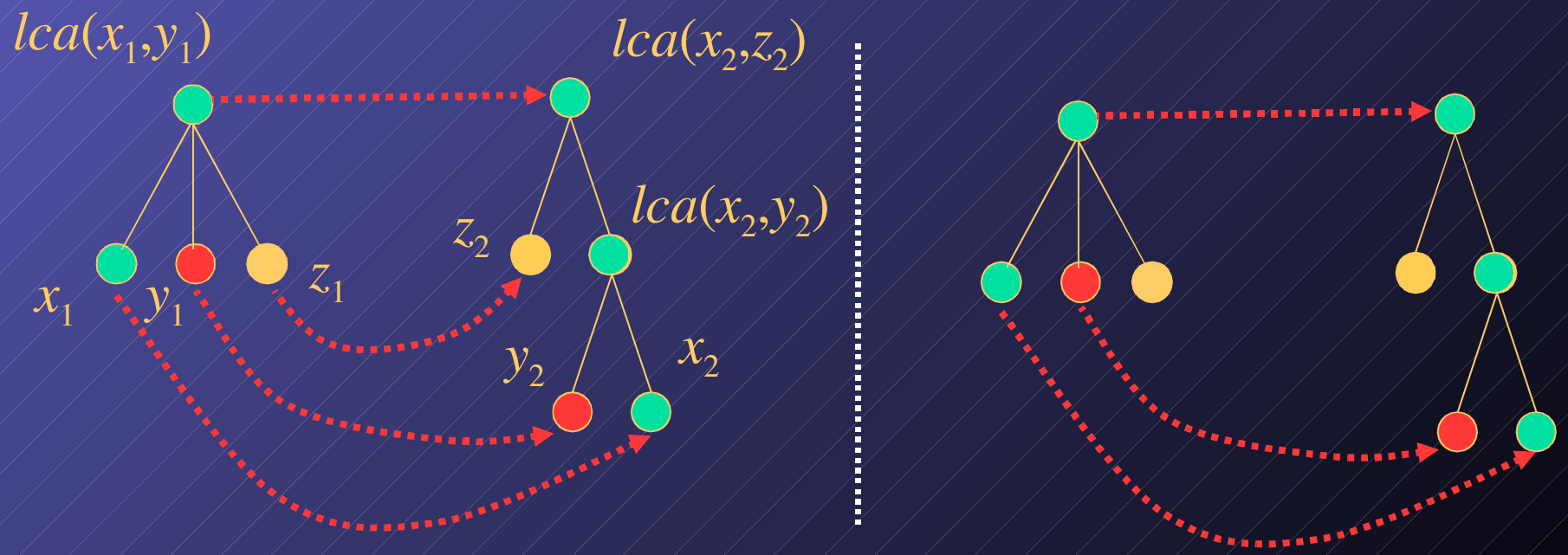
$$D(F_1,F_2)=\min_{M\in\mathcal{M}_1}\left\{d(M)\right\}$$

- Problème MAX SNP-dur dans $\mathcal{M}_1$ (Zhang 94)
- *Idée*: Trouver une solution du problème d'optimisation en ajoutant de nouvelles contraintes dans les alignements: $\mathcal{M}_2 \subset \mathcal{M}_1$



Contraintes de Zhang

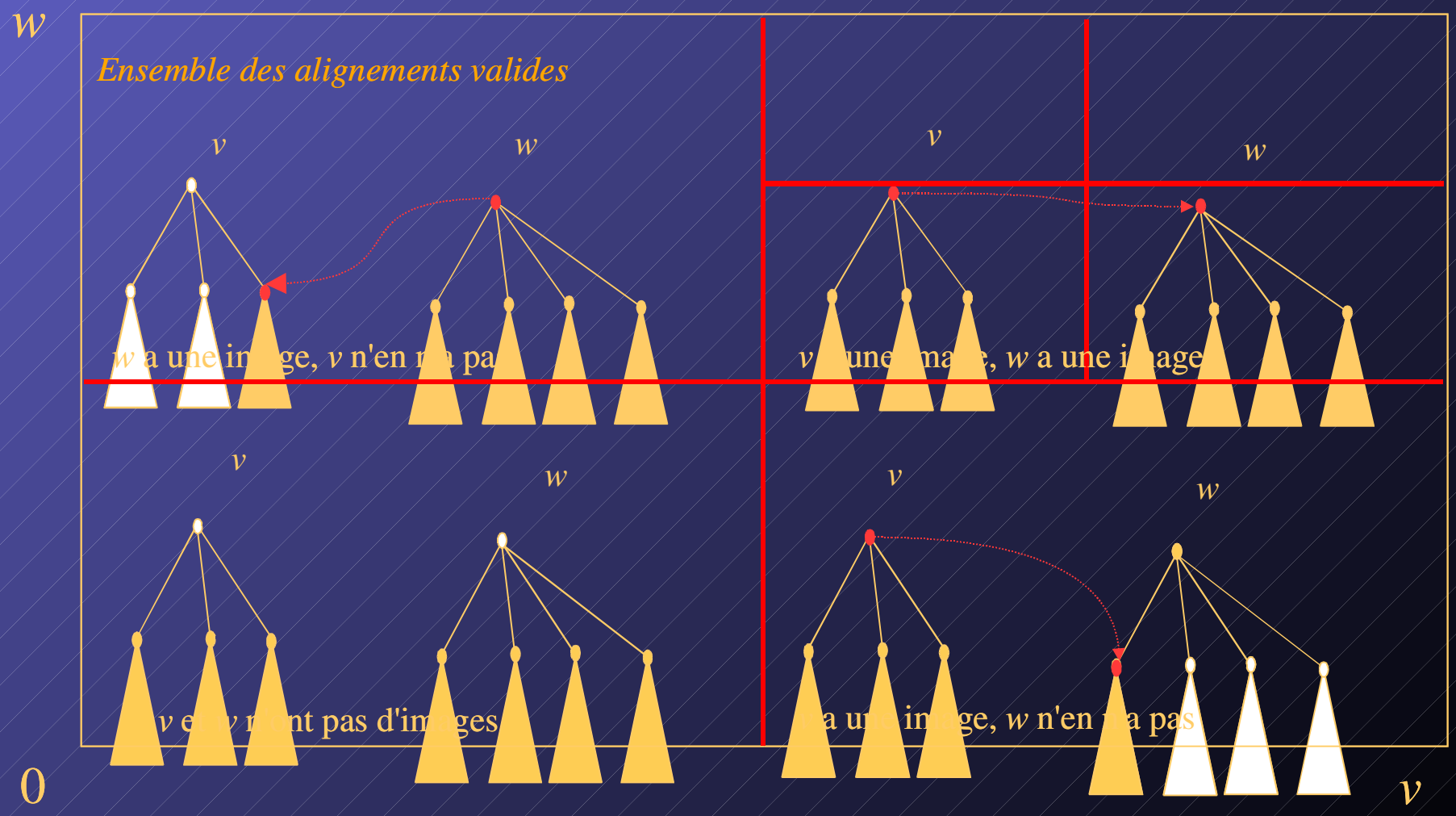
$$lca(x_1, y_1) < z_1 \Leftrightarrow lca(x_2, y_2) < z_2$$



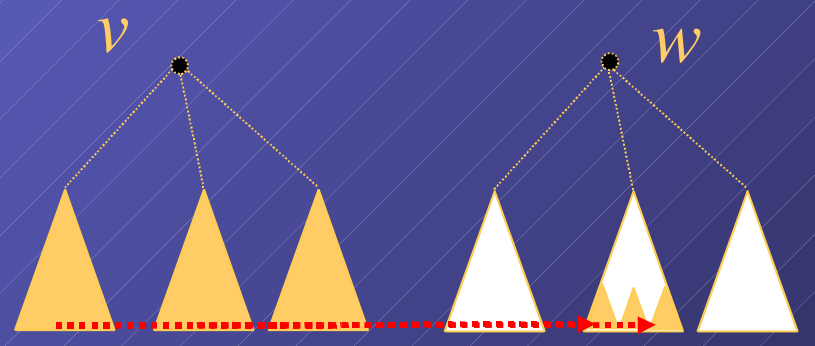
Alignement non-autorisé

Alignement autorisé

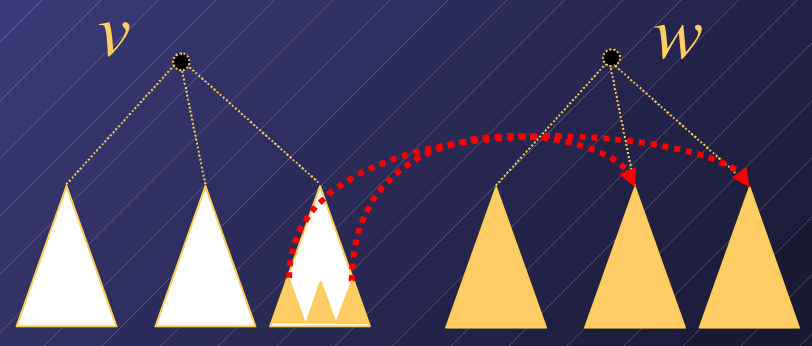
Raffinement de la partition



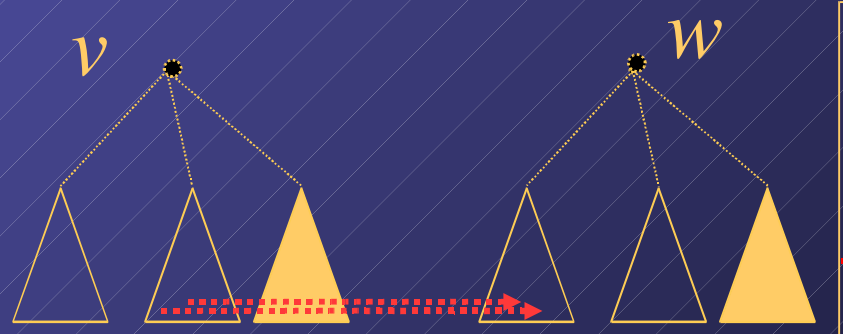
Raffinement de la partition



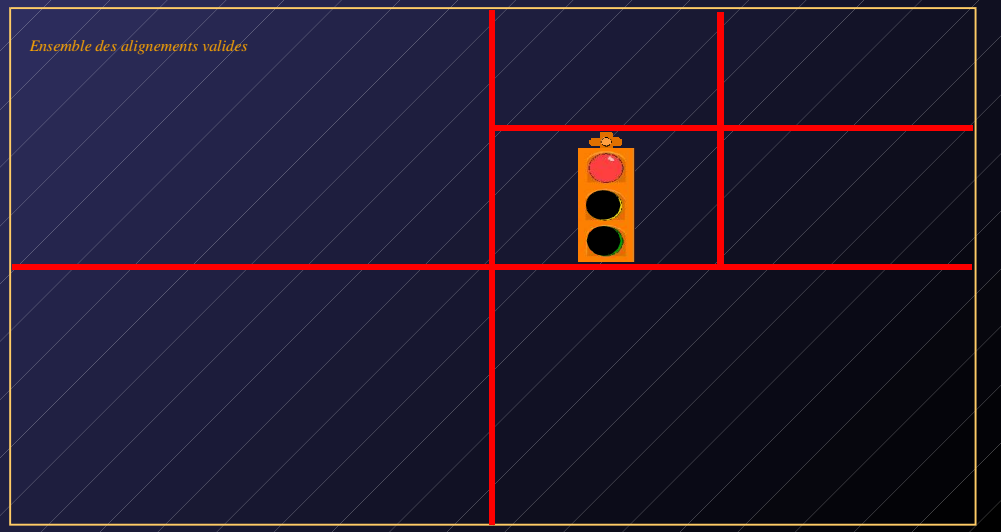
Cas 1



Cas 2



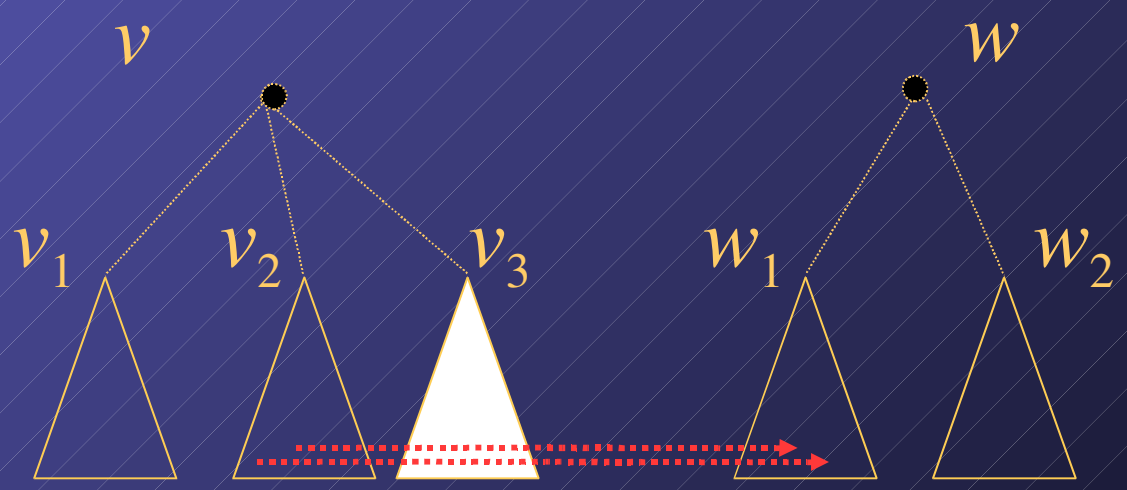
Cas 3



Calcul récursif de l'alignement optimal

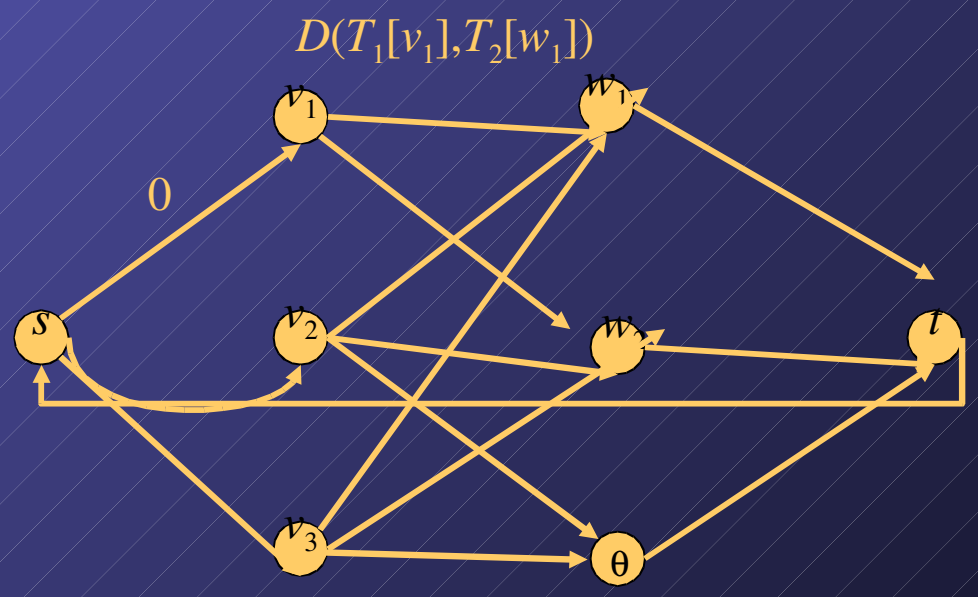
$$D(F_1[v],F_2[v])=\min\begin{cases} D(\Theta,F_2[w])+\min\left\{D(F_1[v],F_2[w_k])-D(\Theta,F_2[w_k])\right\} \\ D(F_1[v],\Theta)+\min\left\{D(F_1[v_k],F_2[w])-D(F_1[v_k],\Theta)\right\} \\ \sum_{(v_k,w_l)\in s[v]\mathrm{x}s[w]} D(T_1[v_k],T_2[w_l]) \end{cases}$$

Alignement restreint



Alignement restreint

Recherche d'un flot maximum de coût minimum (Edmons et Karp 1972)

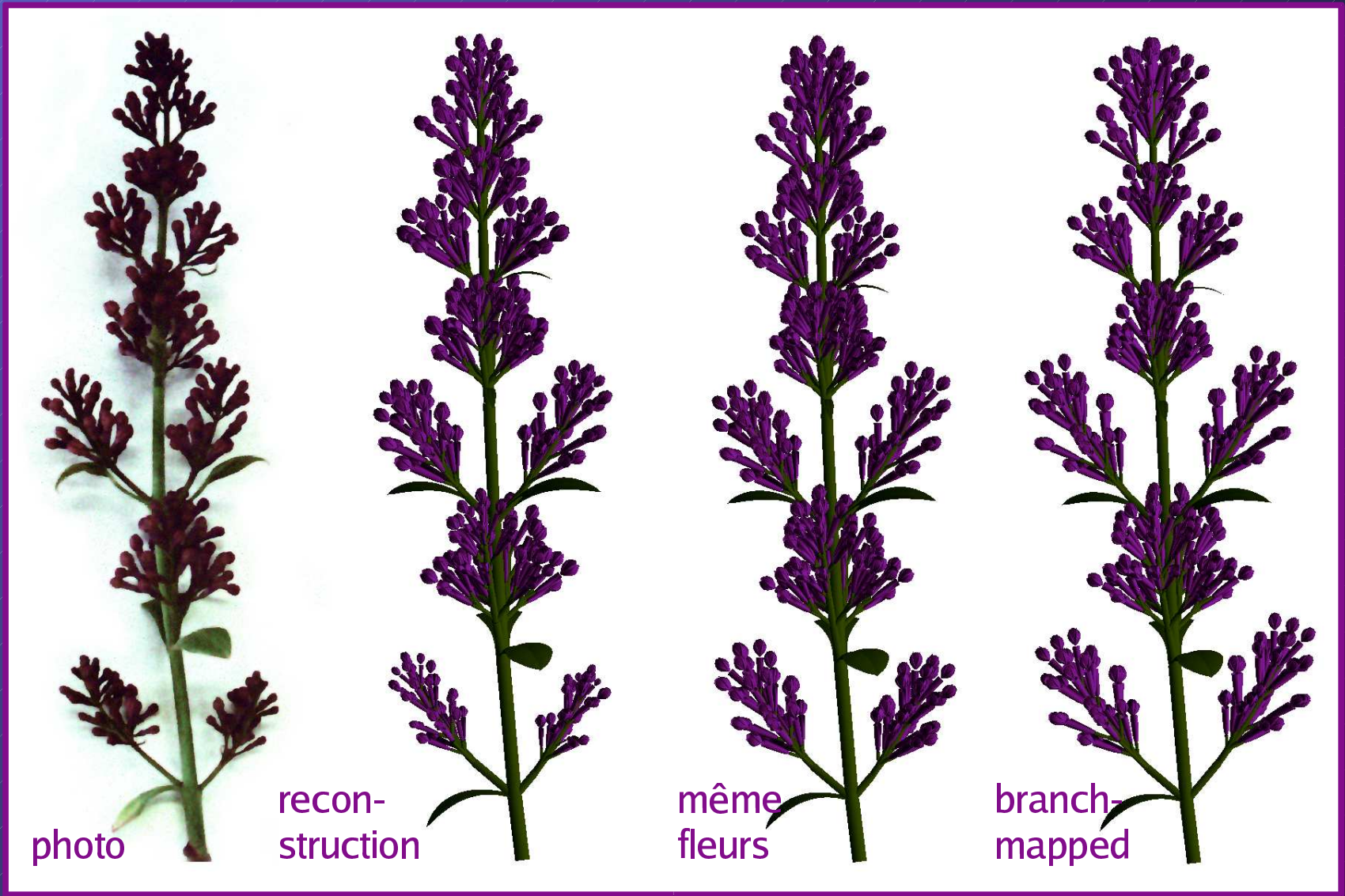


$$\min\{\sum_{(v_k,w_l)\in s[v]\times s[w]} D(T_1[x],T_2[y])\}$$

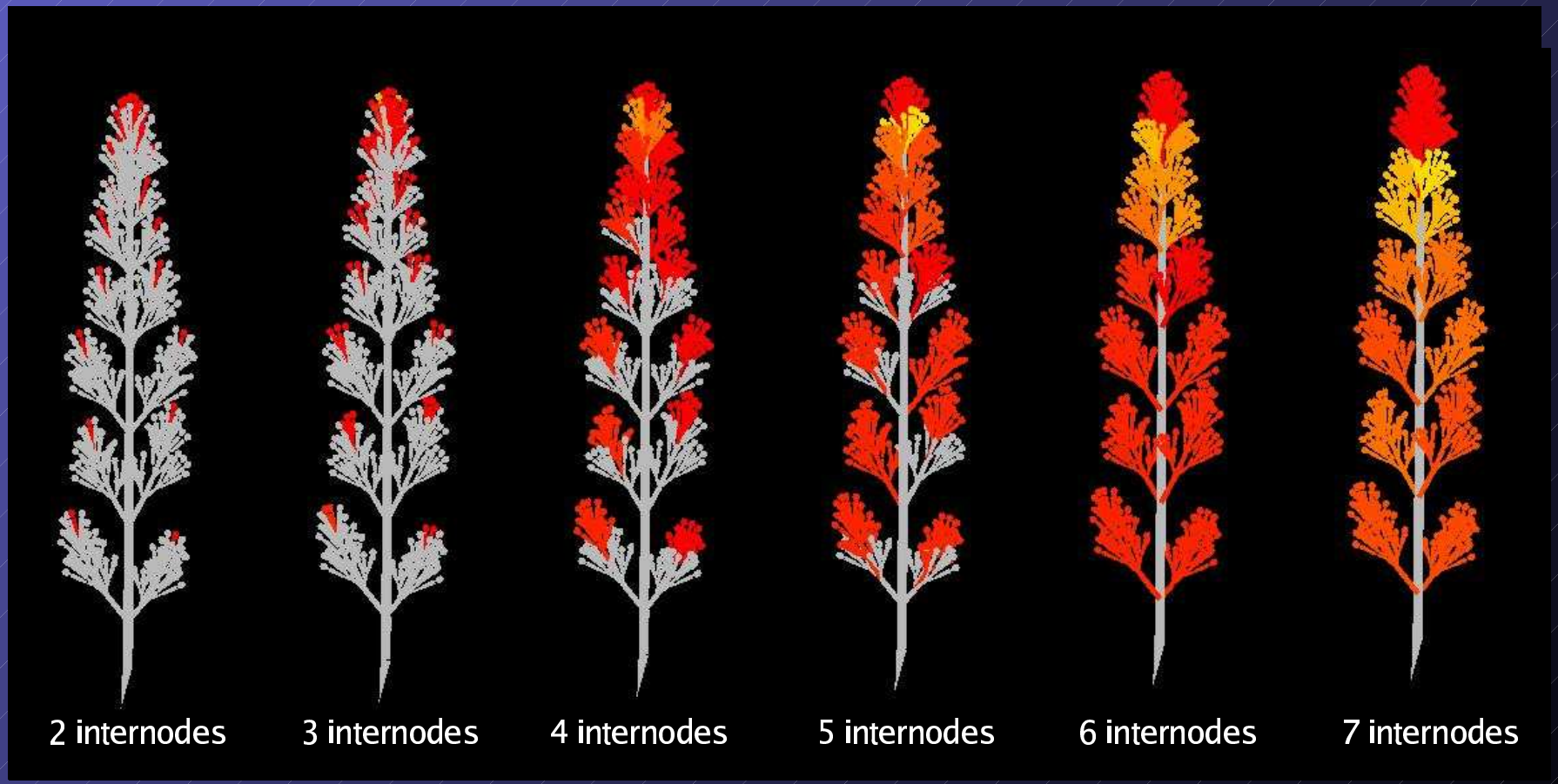
$\Rightarrow$ Complexité de l'algorithme final

$$O(|T_1||T_2|(\deg(T_1)+\deg(T_2))\log_2(\deg(T_1)+\deg(T_2))$$

Évaluation structurelle des propriétés d'autosimilarité d'inflorescences de lilas



Comparaison Topologique (relation paracladiale)



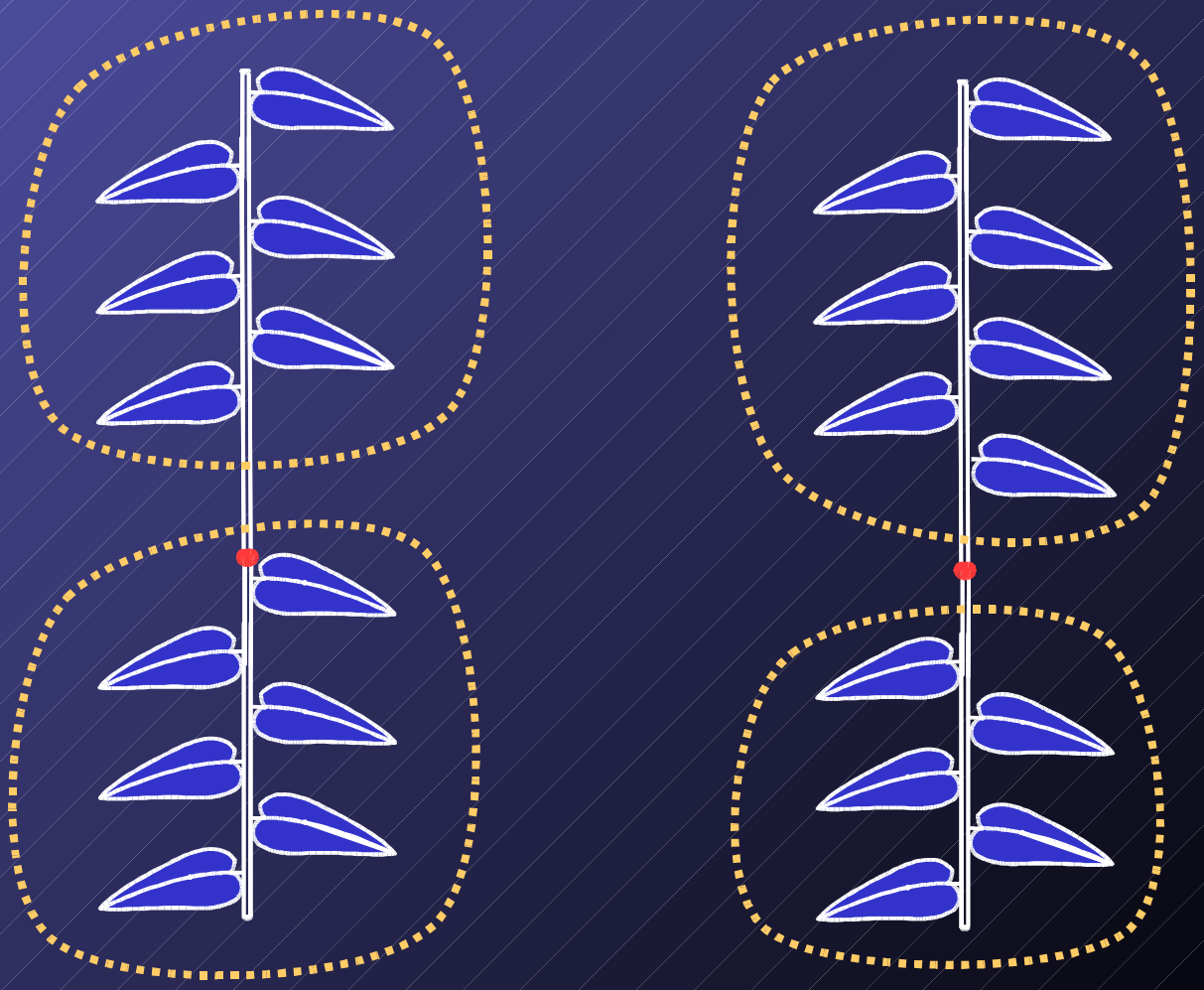
Distance

0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0

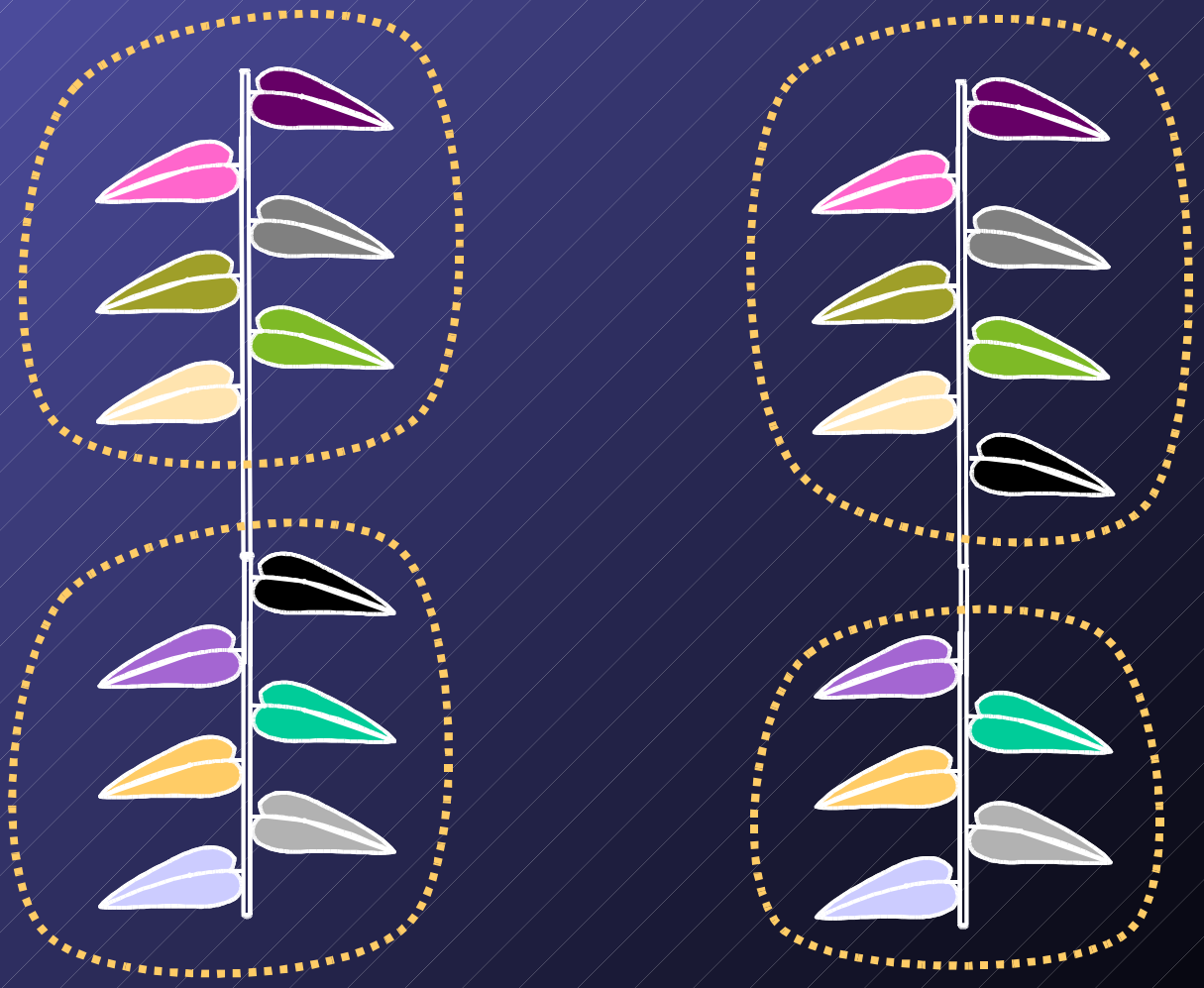
Entités non alignées

Comparaison d'arborescences multi-échelles

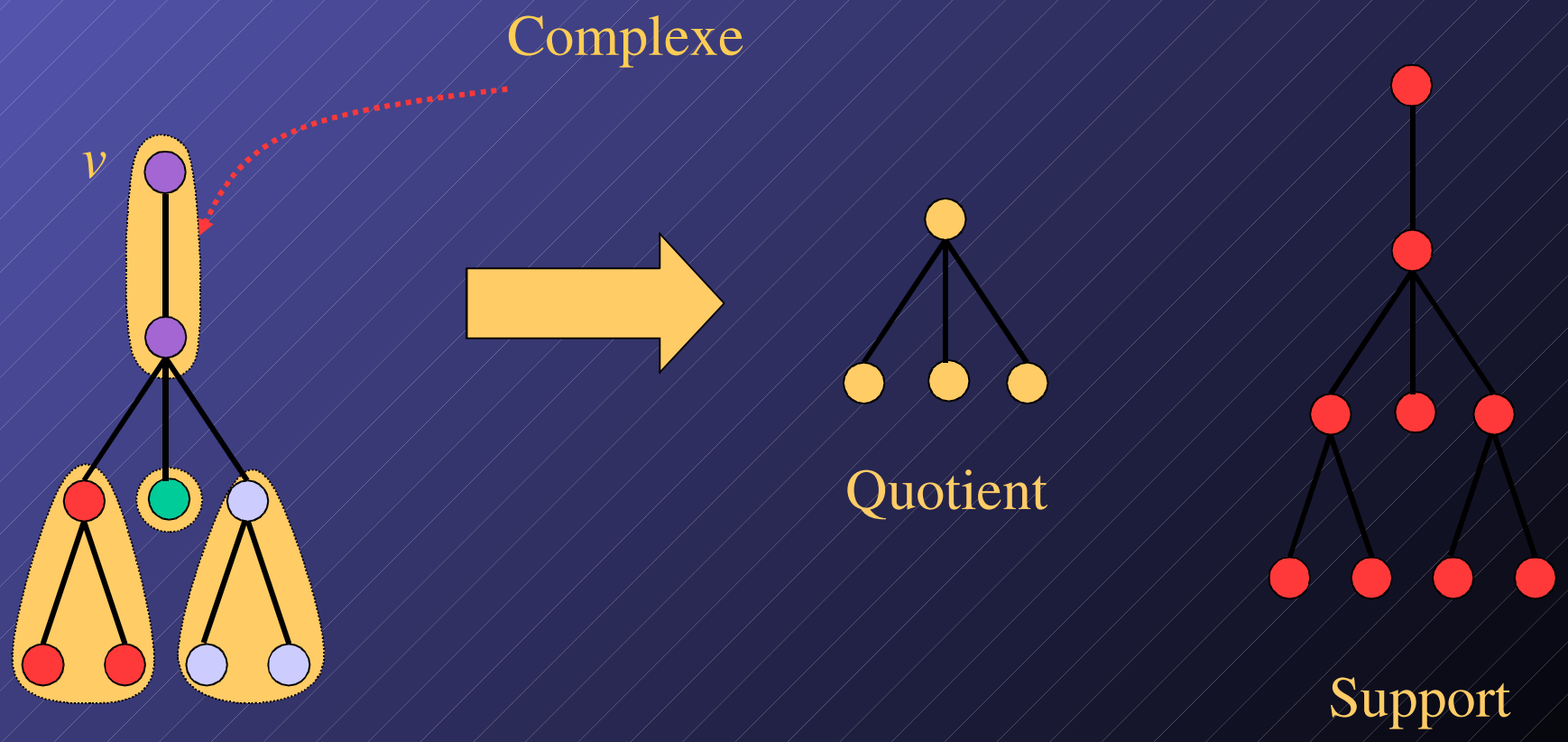
Nouveau problème



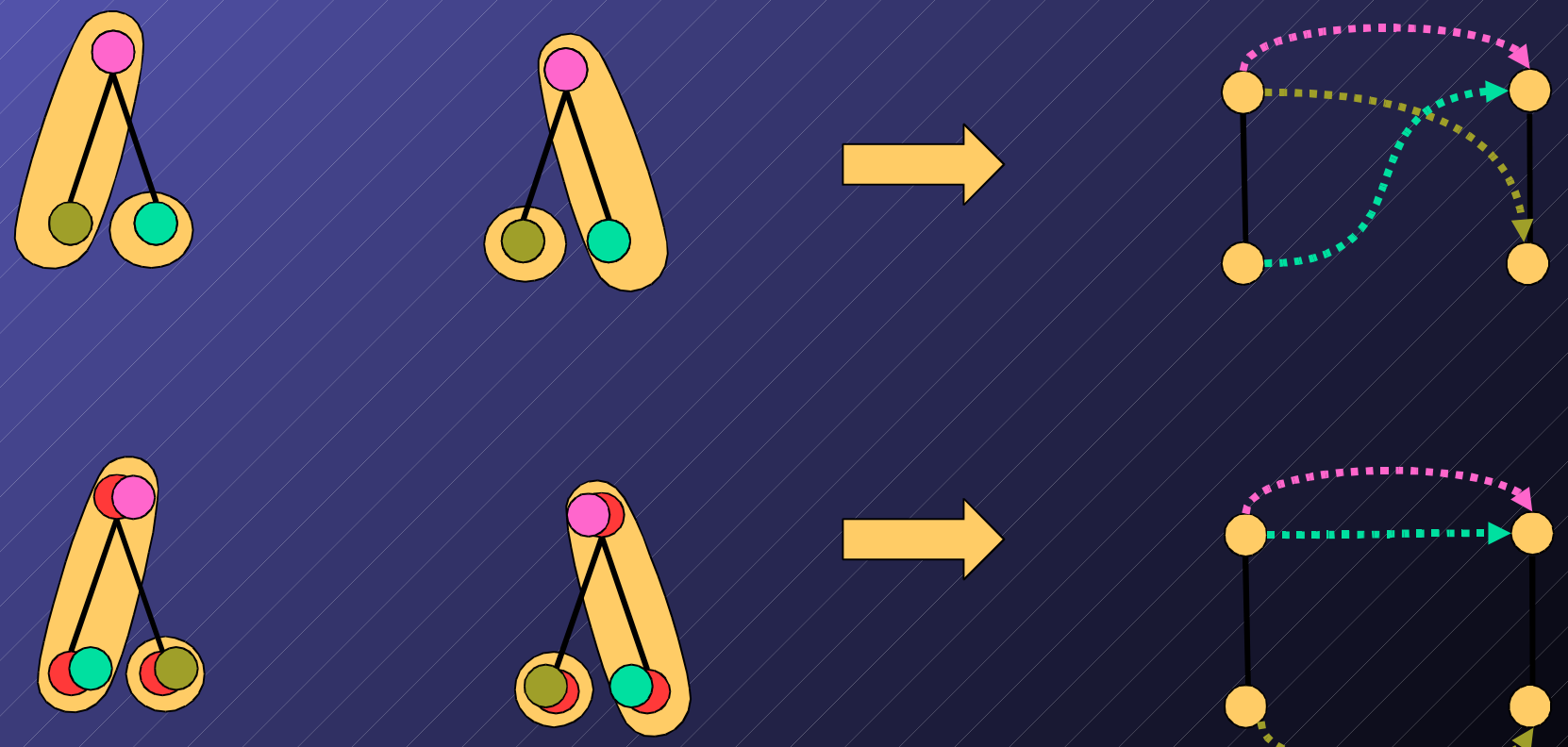
Nouveau problème



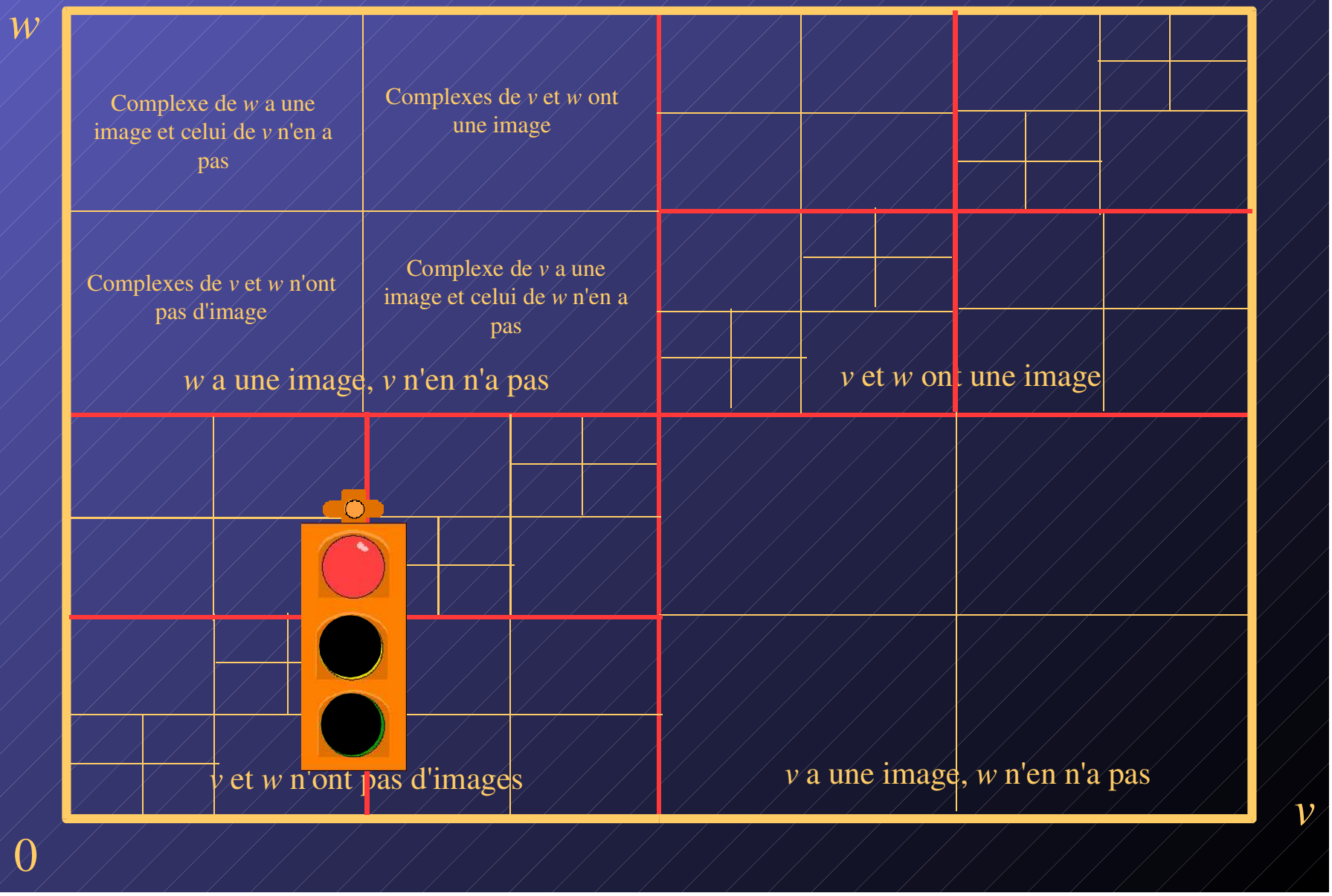
Arborescences quotientées



Alignement induit



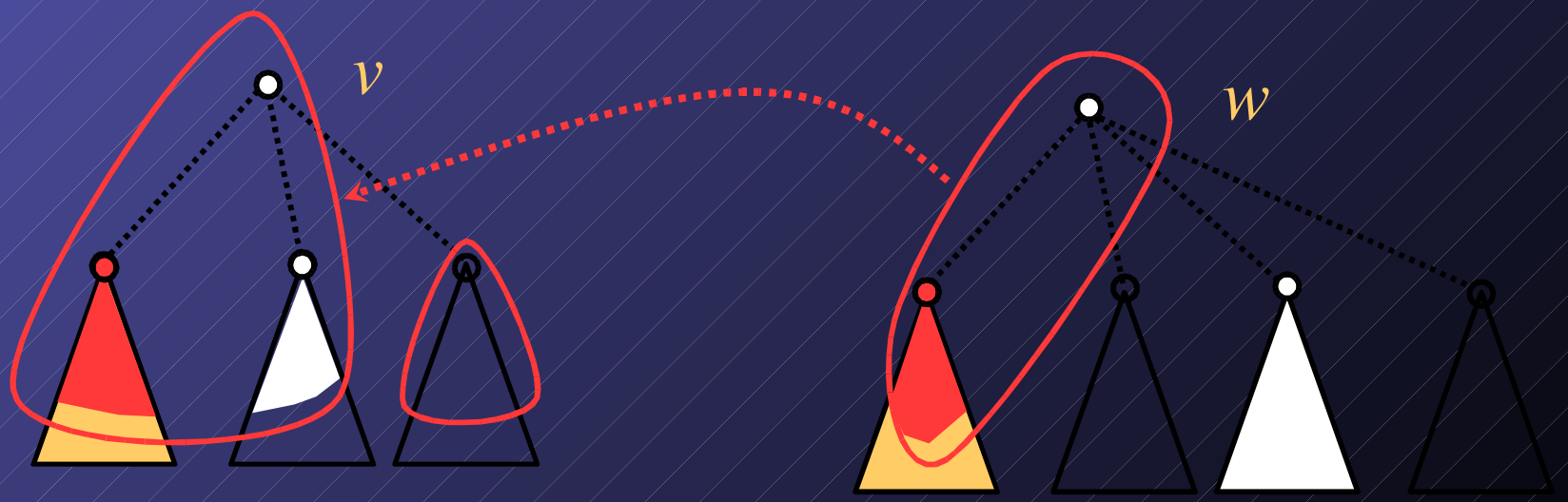
Partition de l'ensemble des alignements



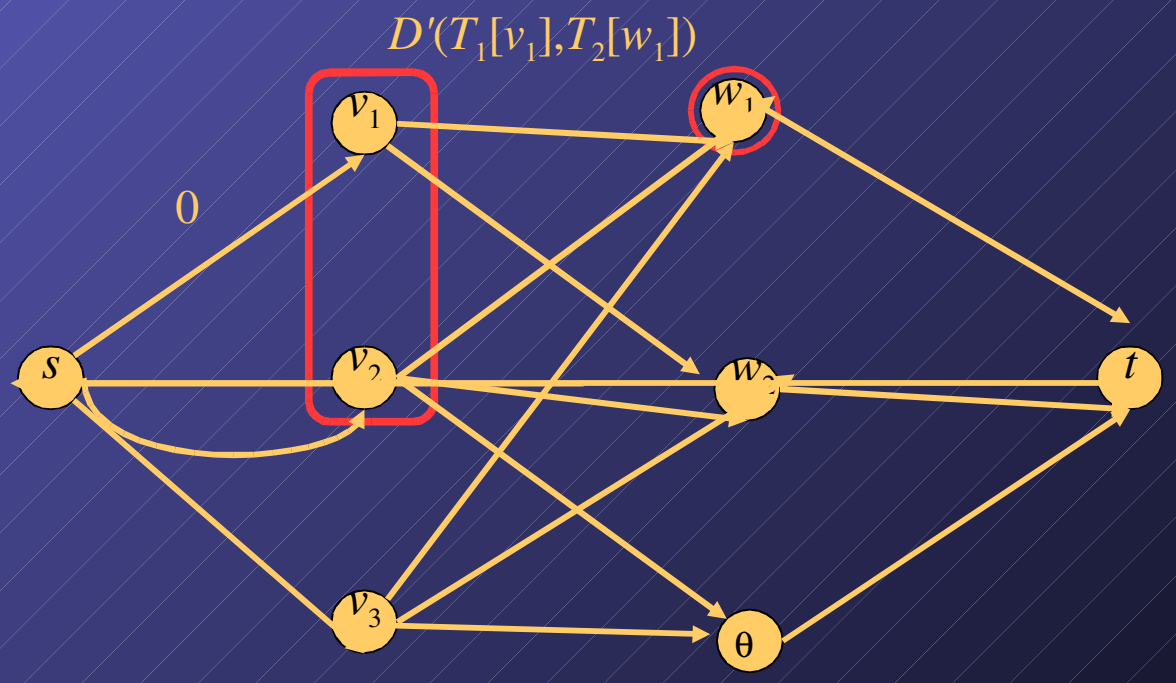
Principe de l'algorithme

2- Comparaison des forêts:

Les complexes de *v* et de *w* sont images l'un de l'autre



Alignement restreint



$$\min\{\sum_{(v_k,w_l)\in s[v]\times s[w]} D'(T_1[x],T_2[y])\}$$

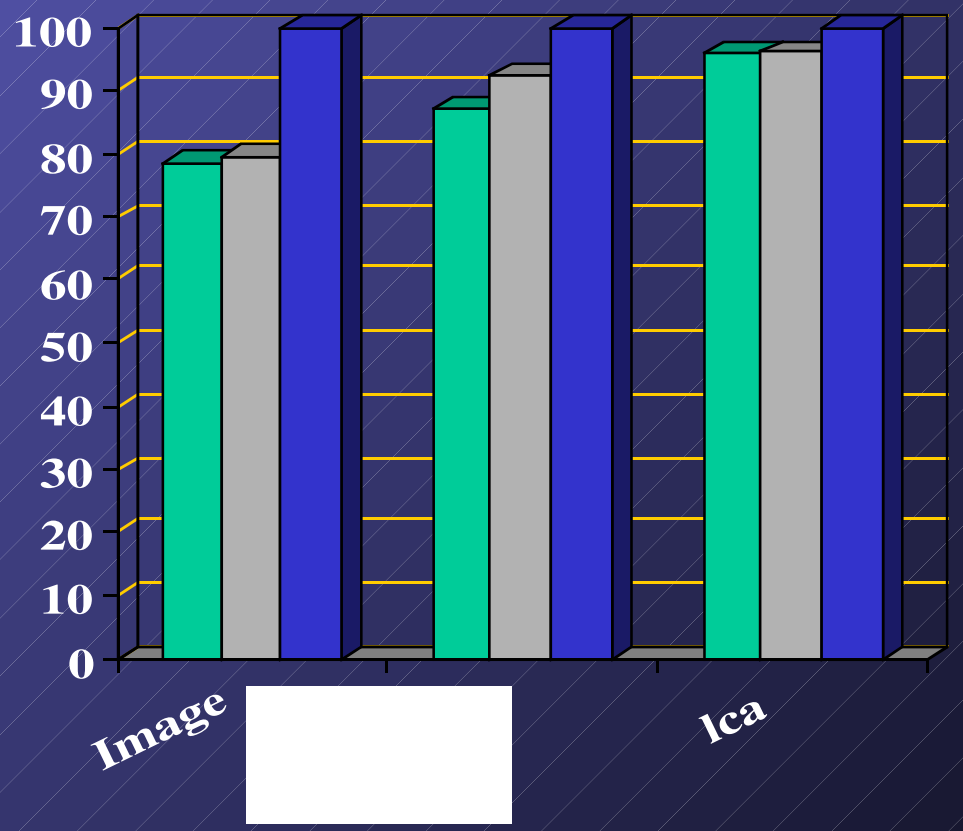
⇒ Complexité de l'algorithme final

$O(|T_1||T_2|(\deg(T_1)+\deg(T_2))\ \log_2(\deg(T_1)+\deg(T_2))\ \min\{\deg_\pi(T_1),\deg_\pi(T_2)\})$

Mise en œuvre pratique pour la comparaison des plantes

- Définition des distances locales
- Effets des paramètres de la comparaison
  - Structure topologique
  - Distance locale
  - Taille et structure des complexes
    - Variation du nombre d'entités par complexe
    - Équilibre des complexes
  - Alignement induit sur les graphes quotients

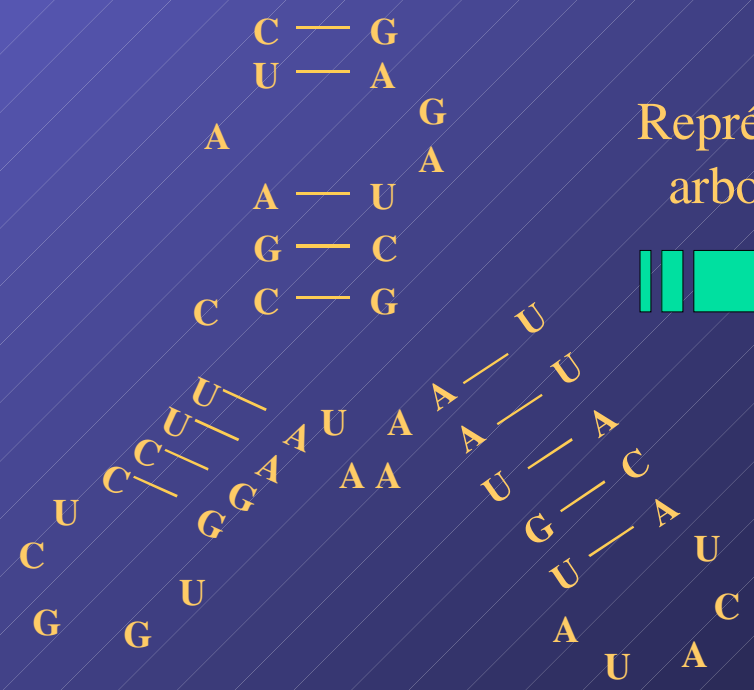
Étude du respect des contraintes sur le graphe quotient



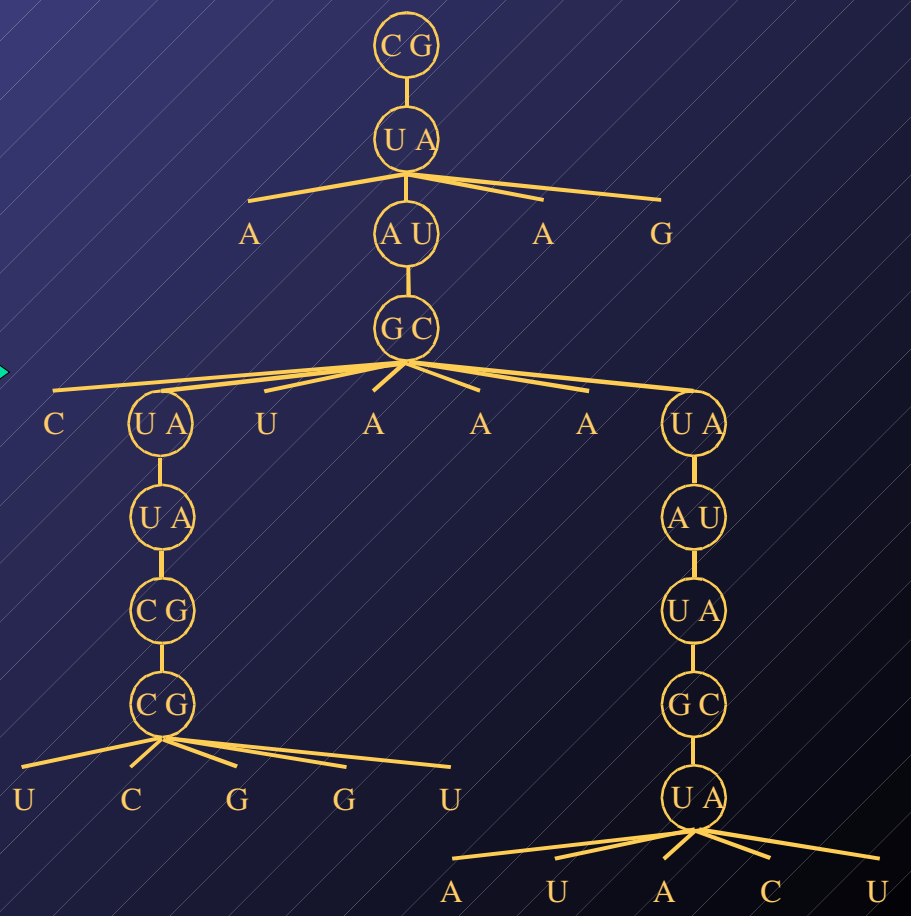
Comparaison de 10 pommiers par les trois méthodes

- Algorithme de Zhang
- Composantes connexes
- Arborescences quotientées

Étude des techniques utilisables pour la comparaison de séquences **génomiques**



Représentation arborescente

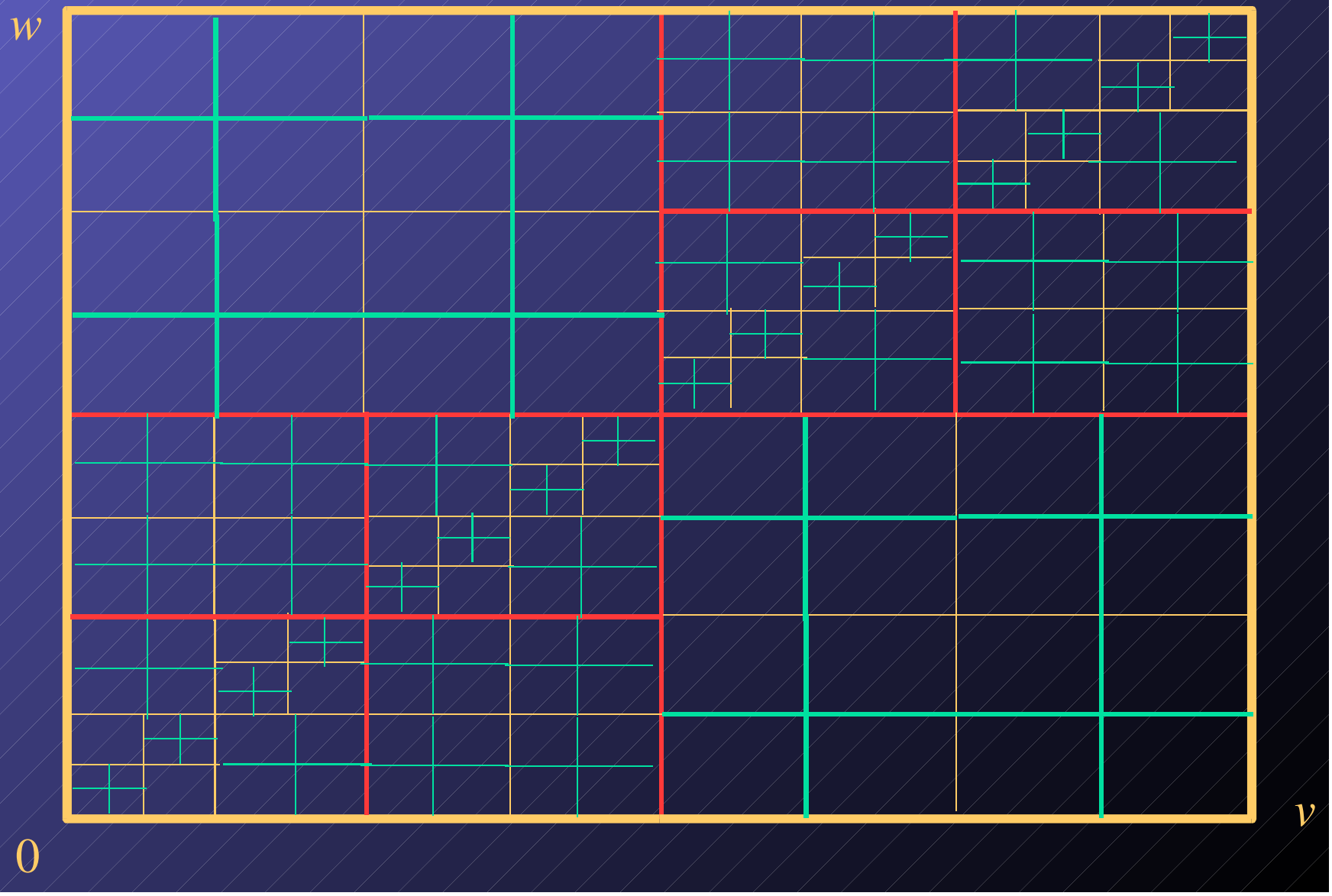


Structure secondaire d'ARN
(Zhang 2000, Labri)

Perspectives

- Applications à des bases de données architecturales
- Évaluation des modèles structure-fonction
- Généralisation aux arborescences multi-échelles
- Adaptation des algorithmes à d'autres domaines :
  - informatique théorique,
  - modélisation biologique,
  - génomique.

Partition de l'ensemble des alignements



Travaux en cours

- Développement d'un outil de visualisation de comparaison de structures secondaires d'ARN

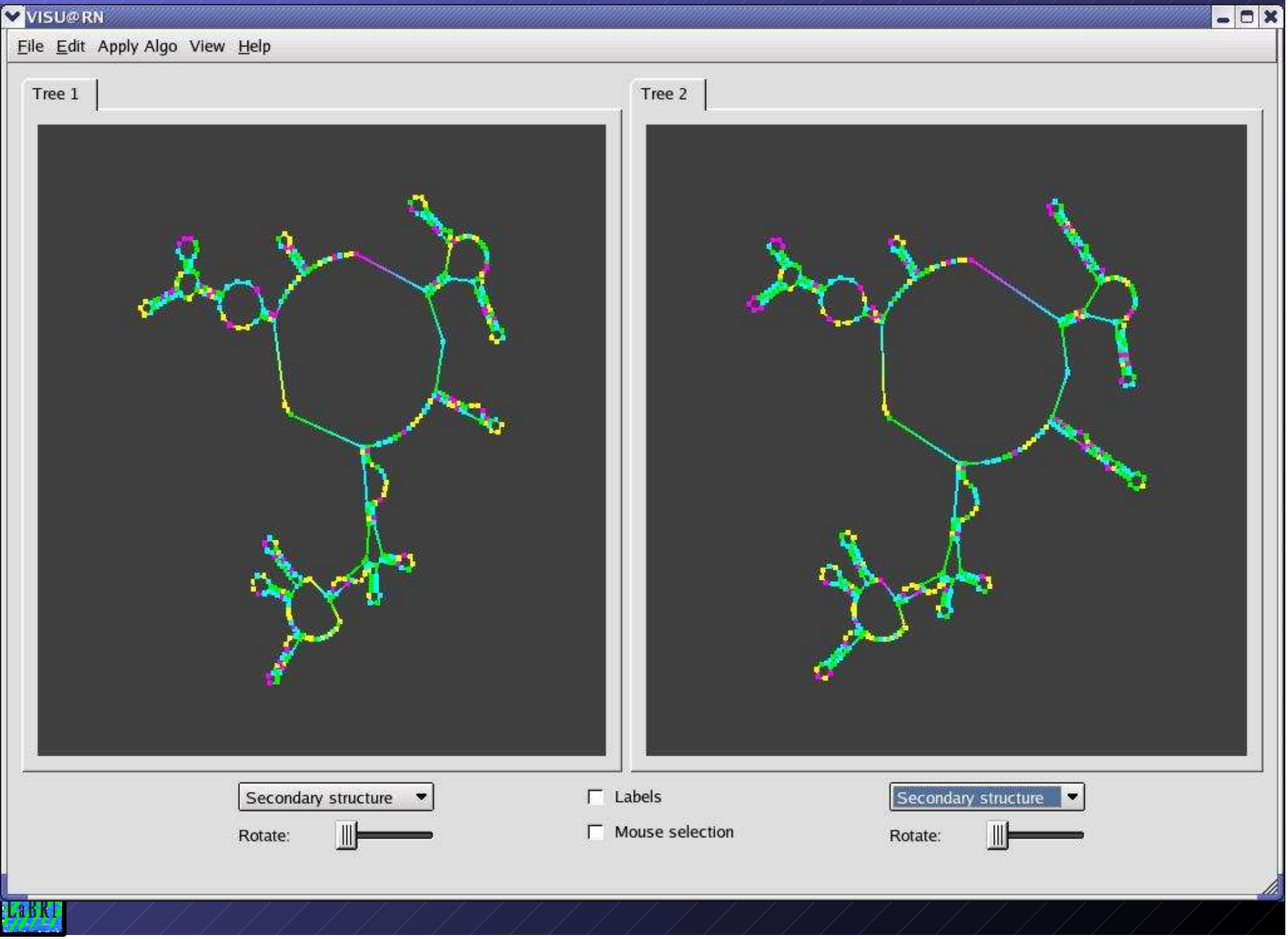
  $\Rightarrow$ tests : trouver les scores de substitution et les pénalités associées aux *gaps* les plus appropriés

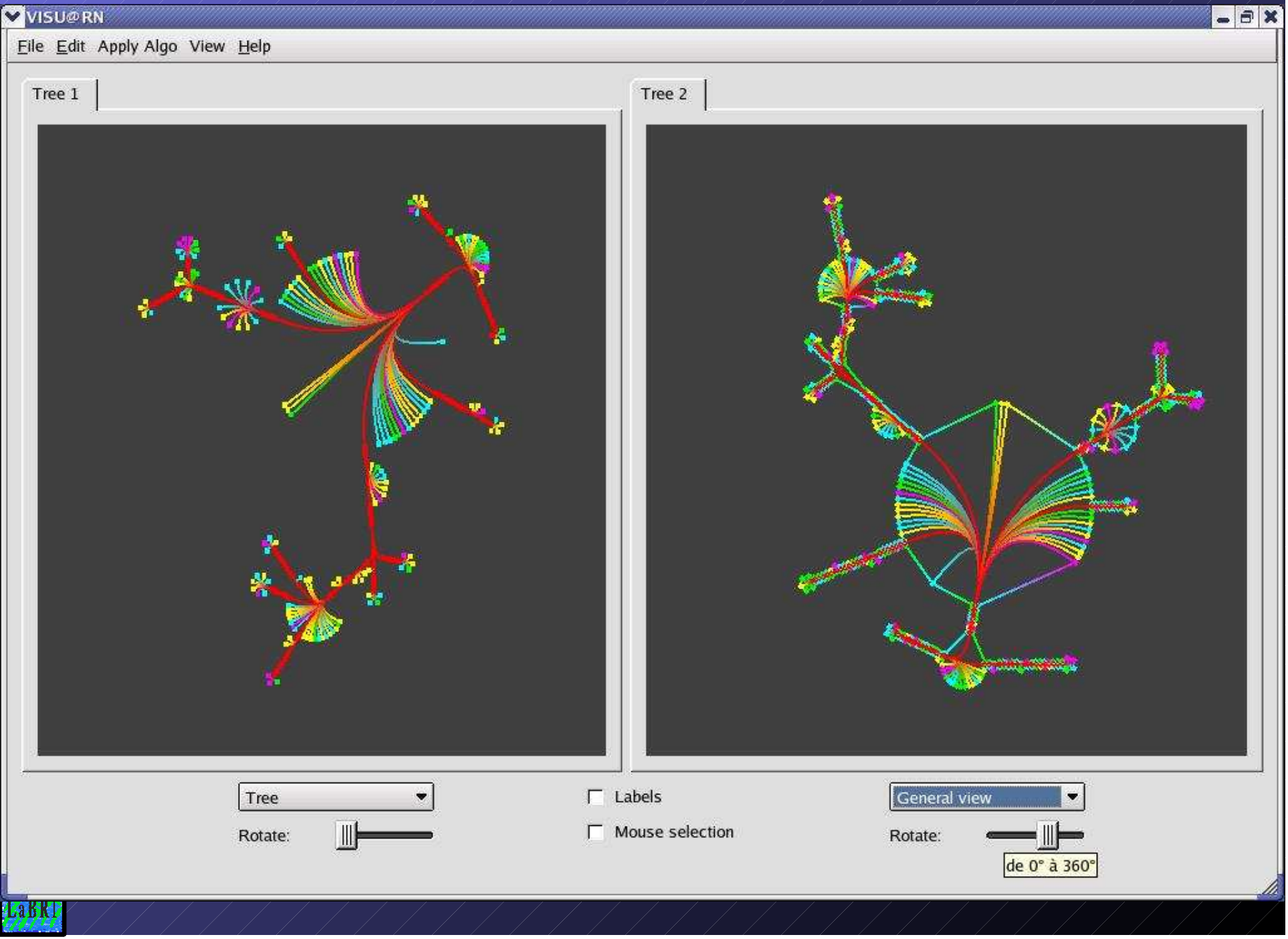
$\Rightarrow$ logiciel visu@rn

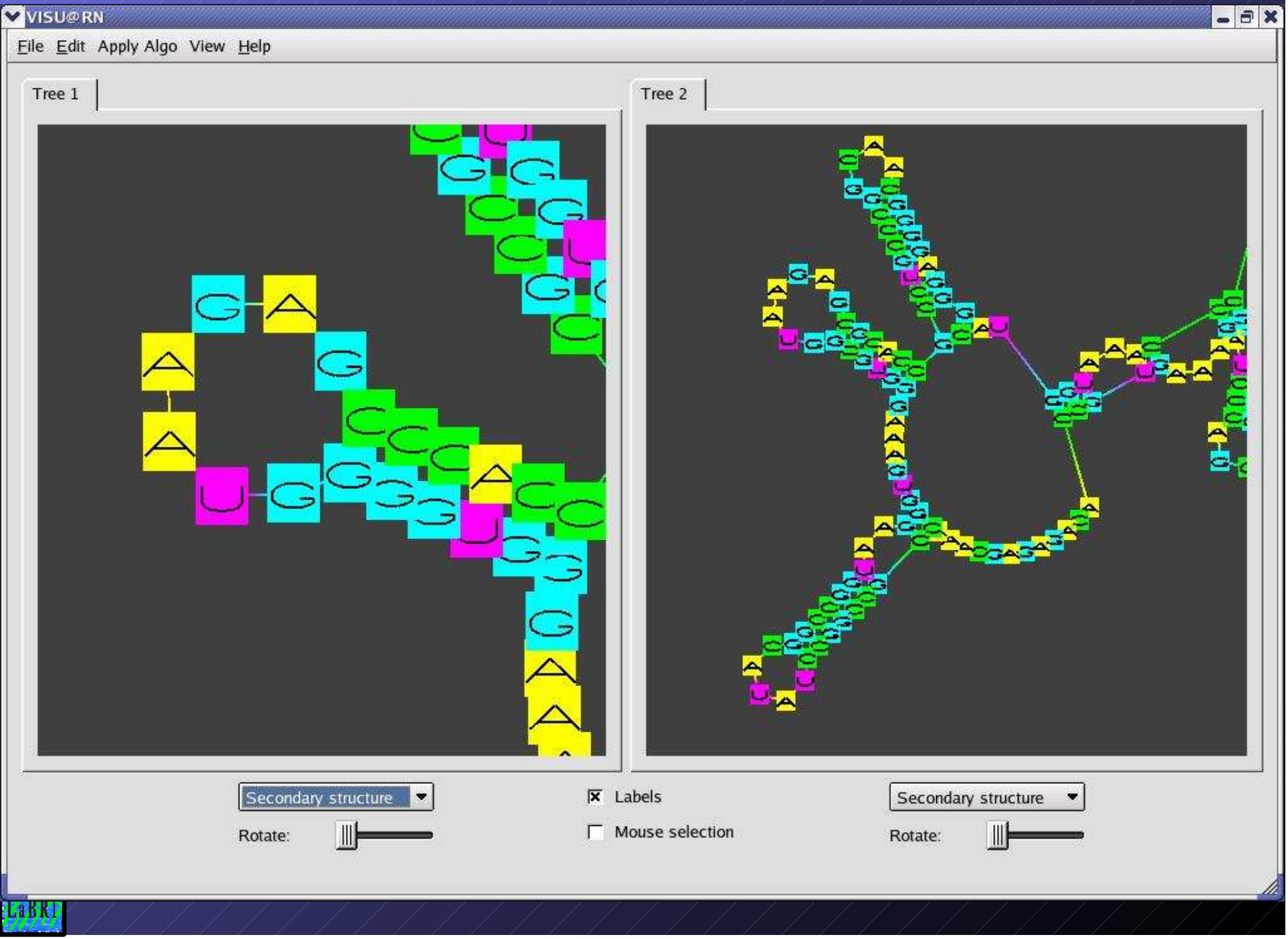
Visu@RN : objectifs

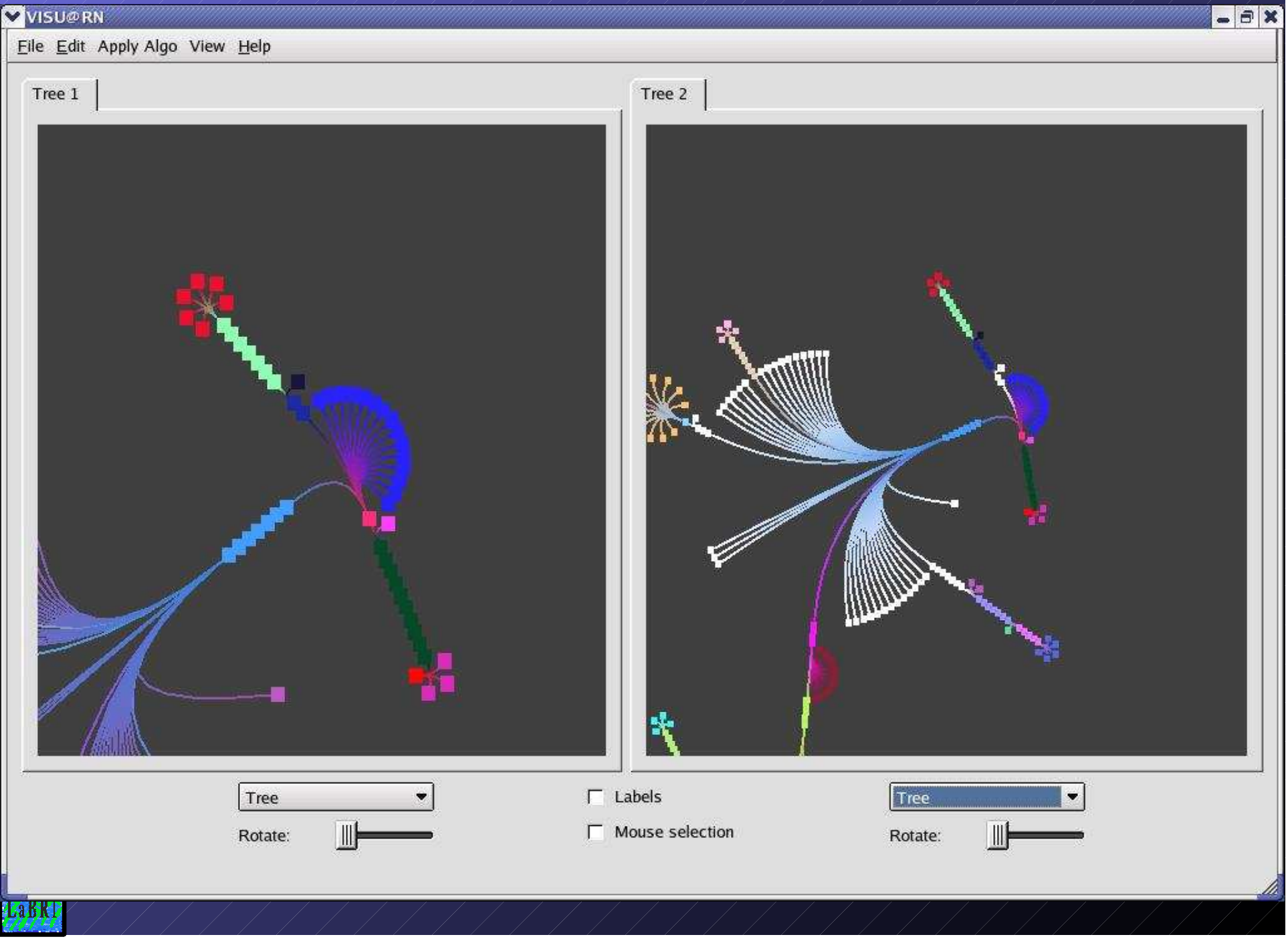
Obtenir un outil convivial permettant principalement :

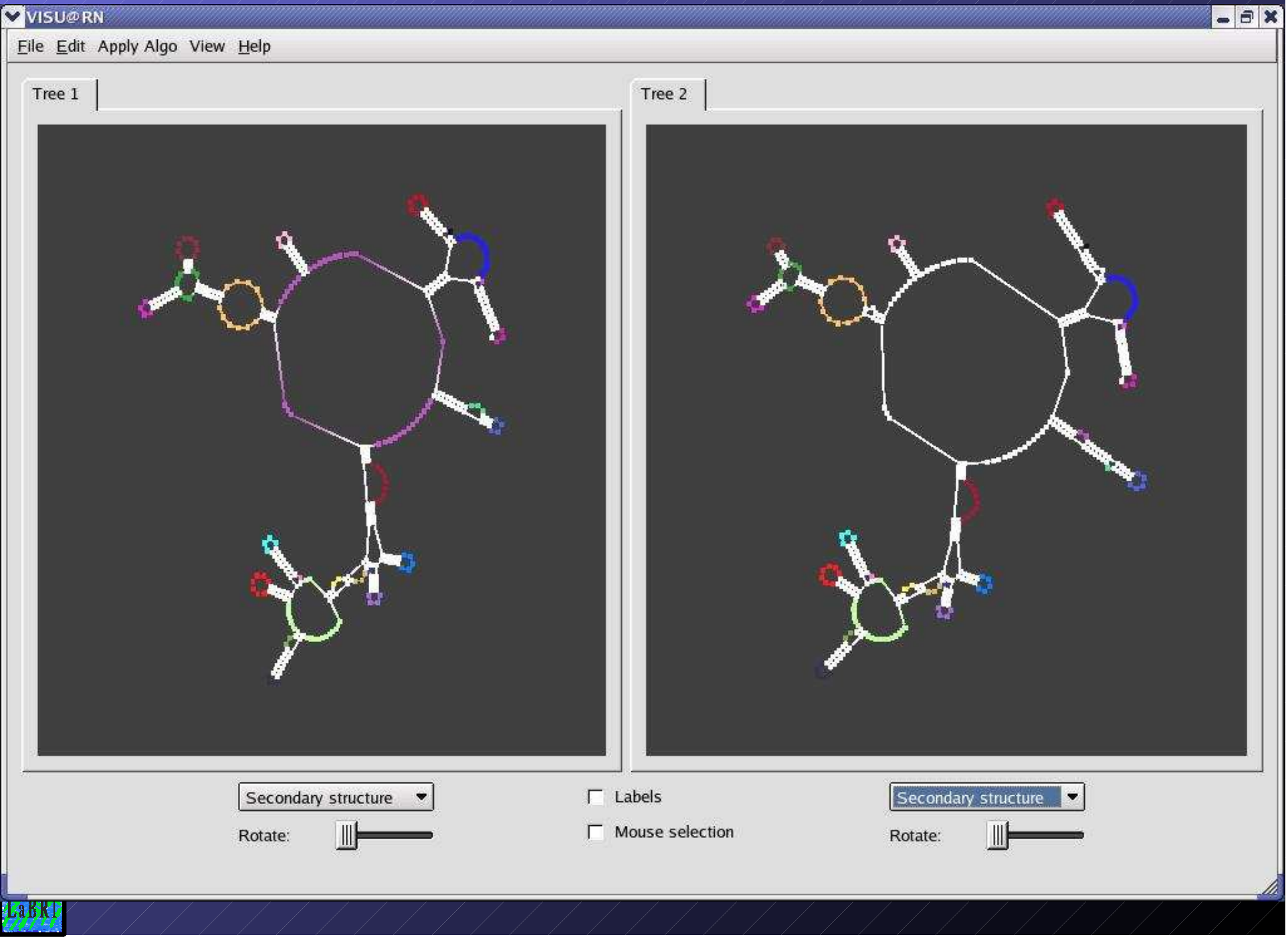
- d'obtenir un dessin totalement automatique, déterministe et reproductible des structures secondaires, sans demander d'expertise humaine,
- de lancer différents algorithmes de comparaison de structures secondaires, à différentes echelles données,
- de comparer un ensemble de structures secondaires entre elles

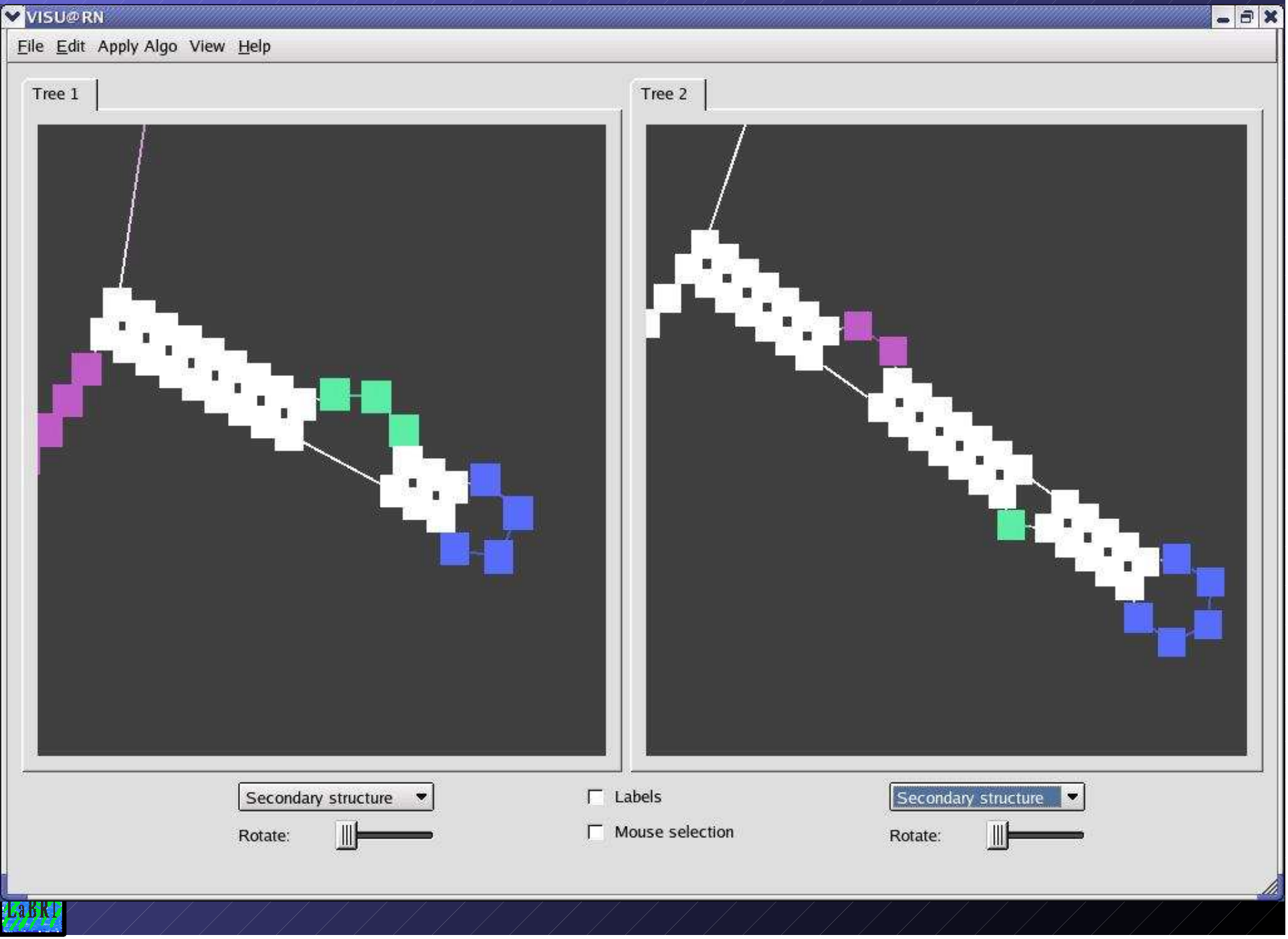

VISU@RN
File Edit Apply Algo View Help
Tree 1
Tree 2
Secondary structure
Rotate:
Labels
Mouse selection
Secondary structure
Rotate:


VISU@RN
File Edit Apply Algo View Help
Tree 1
Tree 2
Tree
Rotate:
Labels
Mouse selection
General view
Rotate:
de 0° à 360°


VISU@RN
File Edit Apply Algo View Help
Tree 1
Tree 2
Secondary structure
Rotate:
Labels
Mouse selection
Secondary structure
Rotate:


VISU@RN
File Edit Apply Algo View Help
Tree 1
Tree 2
Tree
Rotate:
Labels
Mouse selection
Tree
Rotate:

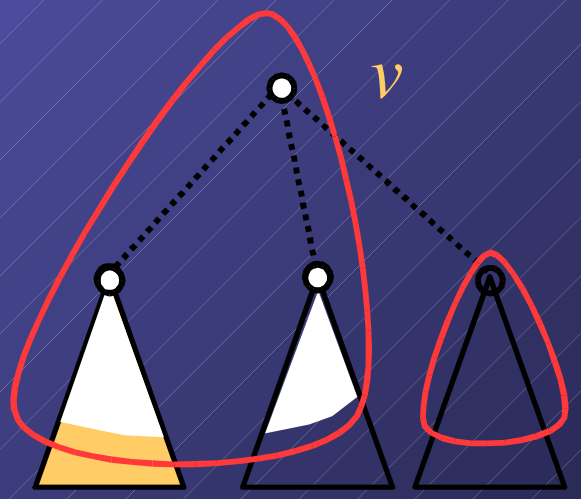

VISU@RN
File Edit Apply Algo View Help
Tree 1
Tree 2
Secondary structure
Rotate:
Labels
Mouse selection
Secondary structure
Rotate:

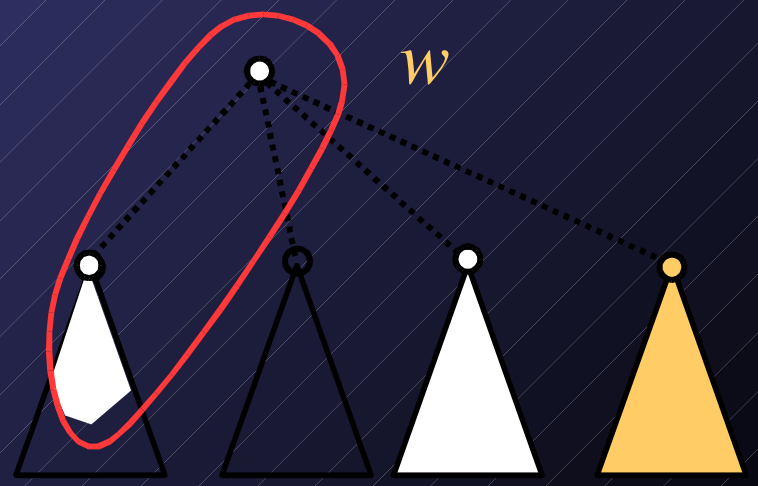

VISU@RN
File Edit Apply Algo View Help
Tree 1
Tree 2
Secondary structure
Rotate:
Labels
Mouse selection
Secondary structure
Rotate:

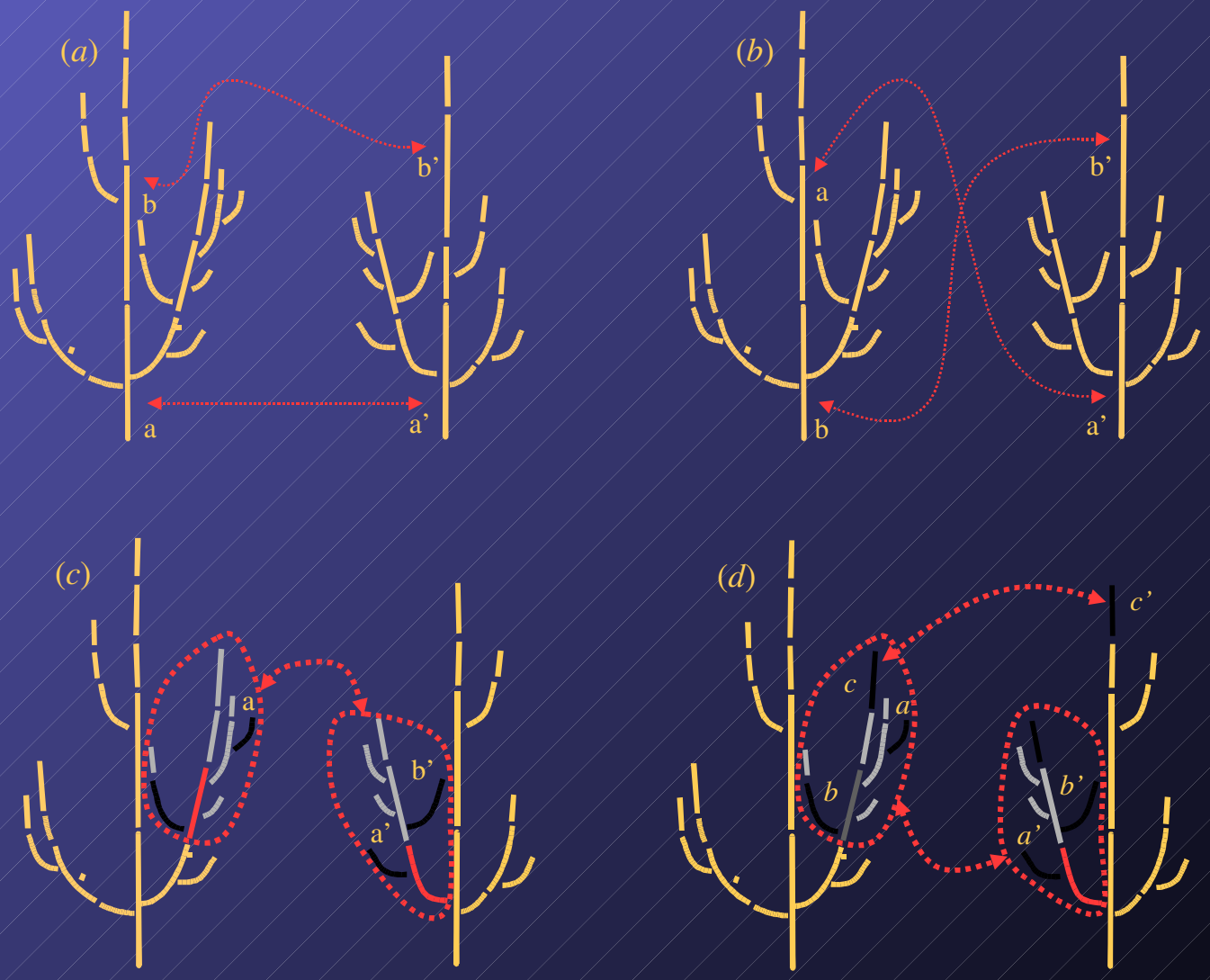
Principe de l'algorithme

2- <u>Comparaison des forêts</u>:

Les complexes de *v* et de *w* sont images l'un de l'autre






(a)
b
b'
a
a'
(b)
a
b'
b
a'
(c)
a
b'
a'
(d)
c'
c
a
b
b'
a'