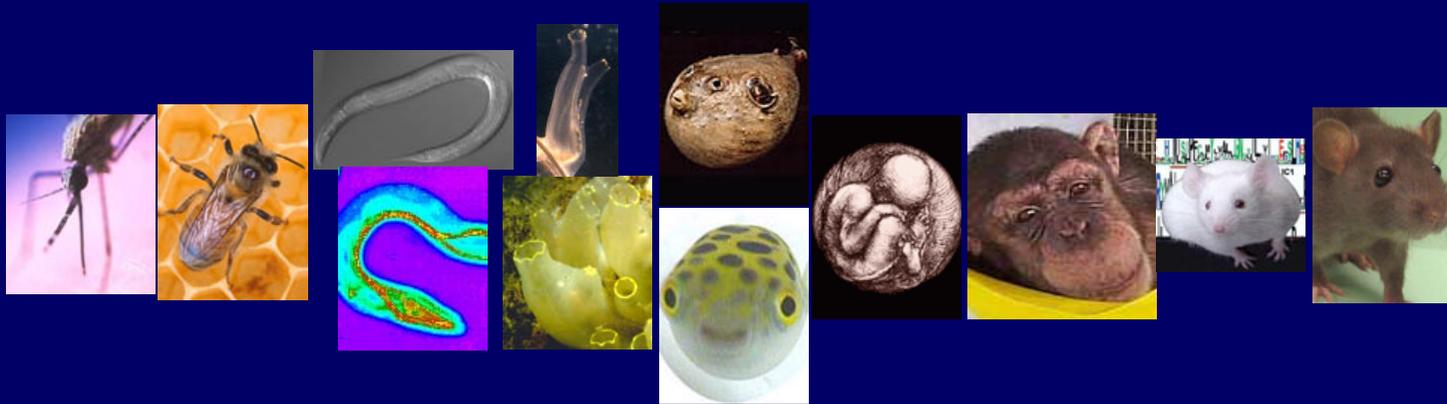
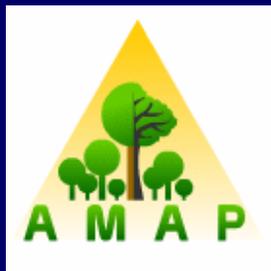


Comparaison de génomes Le problème Sorting By Reversals (SBR)



Sèverine Bérard



AMAP - Université Montpellier 2



- Introduction
- Anatomie d'une permutation signée
- Les effets d'une inversion
- Les algorithmes
- Conclusion et perspectives

- Mutations ponctuelles

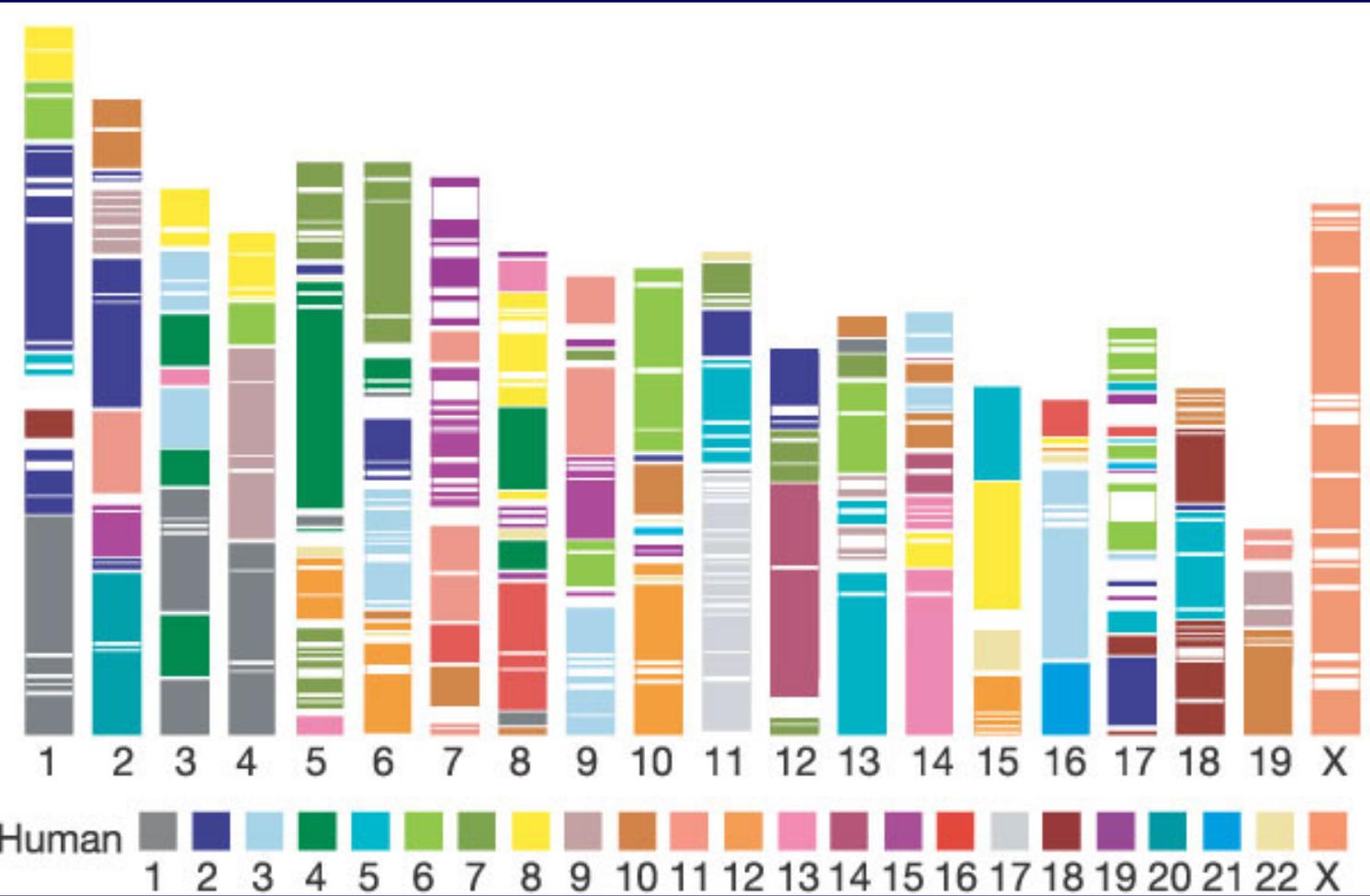
→ (substitutions, insertions, délétions de nucléotides)

- Réarrangements génomiques

→ Modifications dans l'organisation des génomes

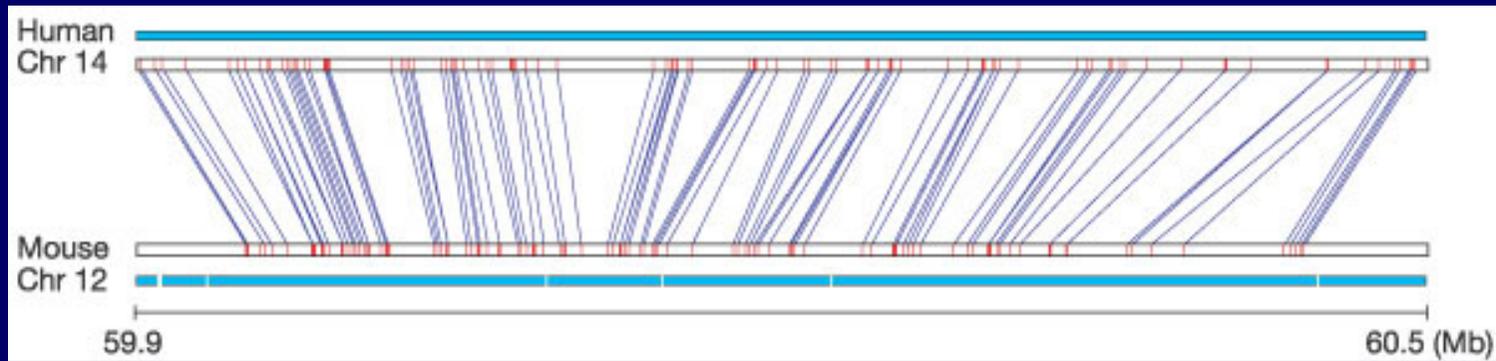
Mesure de l'évolution et comparaison entre espèces en analysant les mutations ponctuelles.

Génomes complets = nouvelles données disponibles



(Nature, 2002)

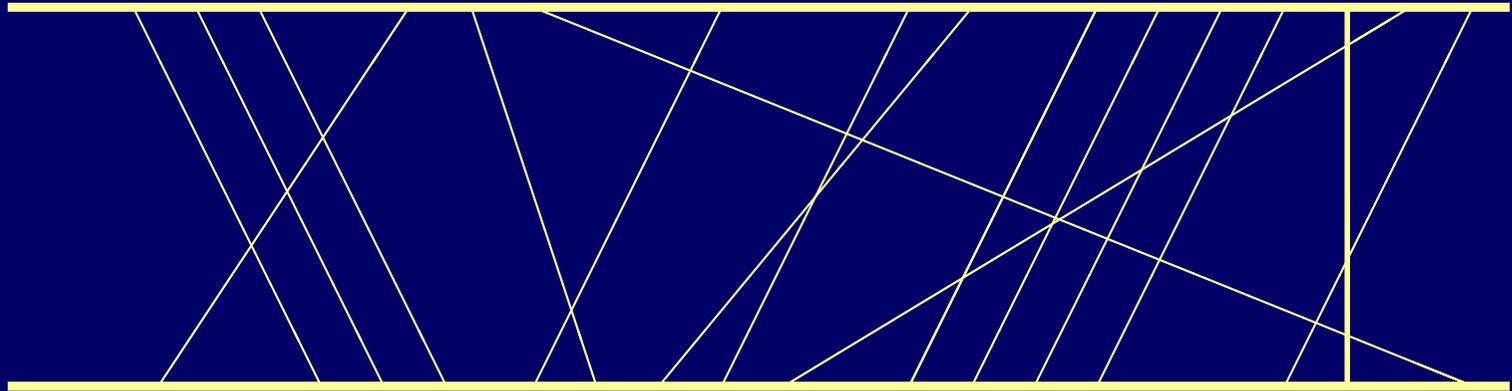
1. Recherche de synténies



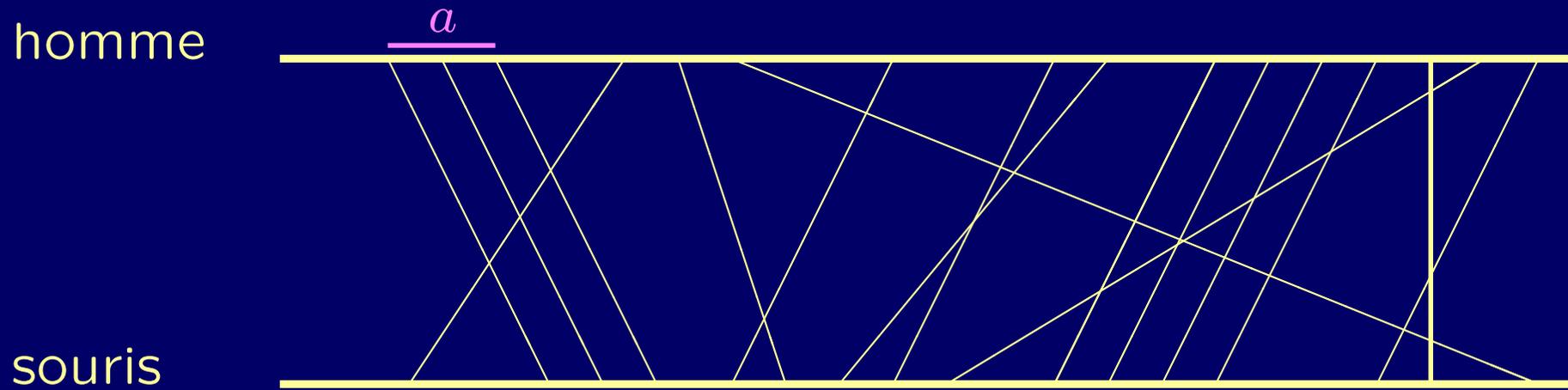
2. Réencodage

homme

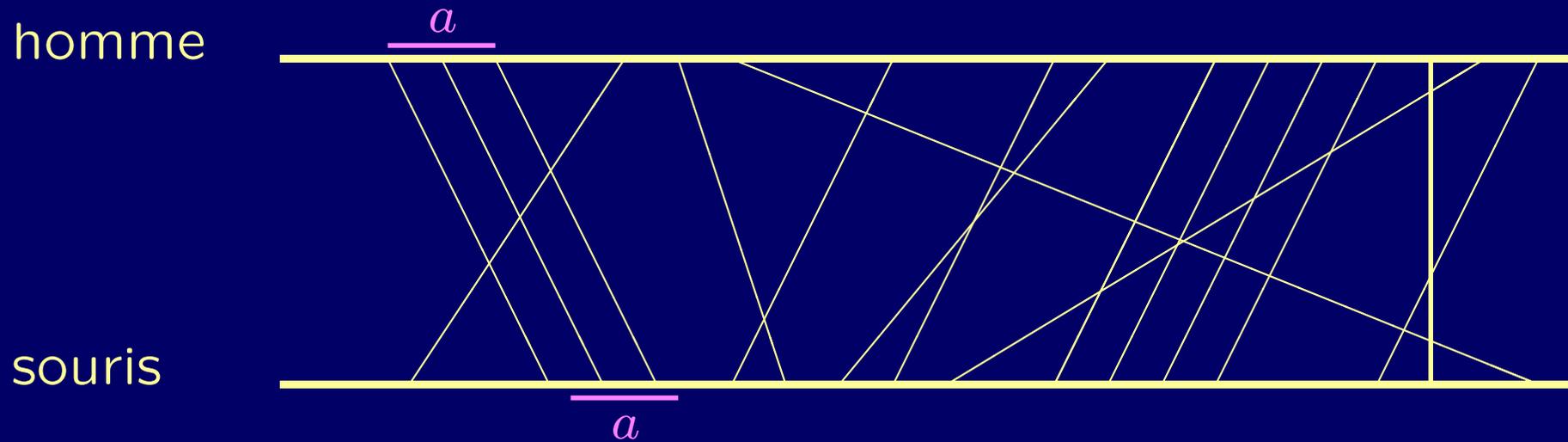
souris



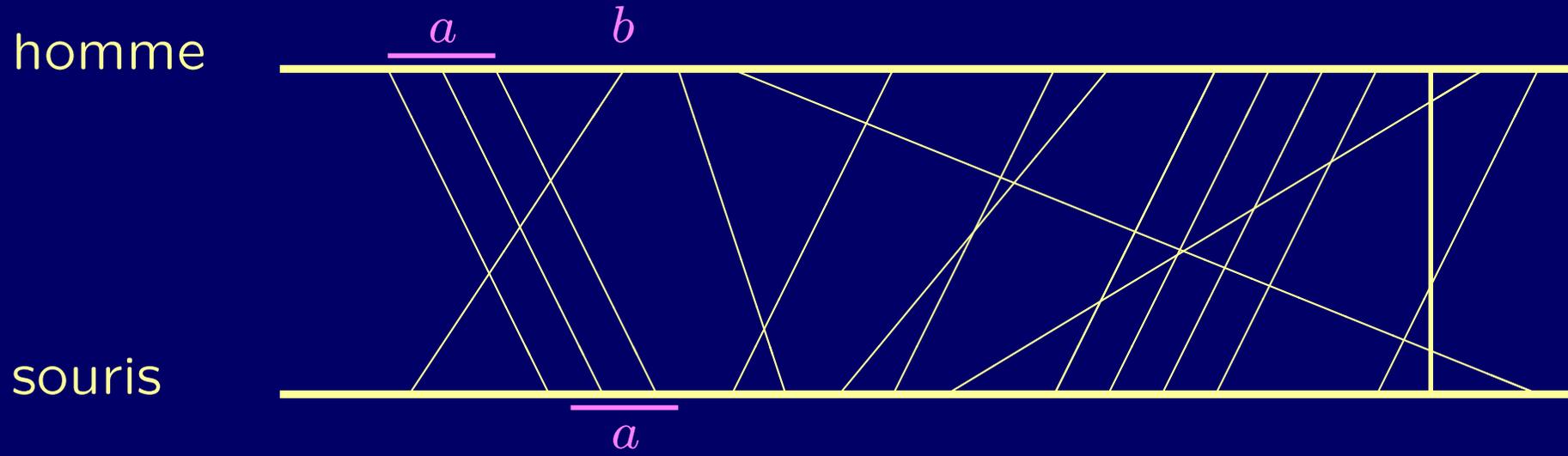
2. Réencodage



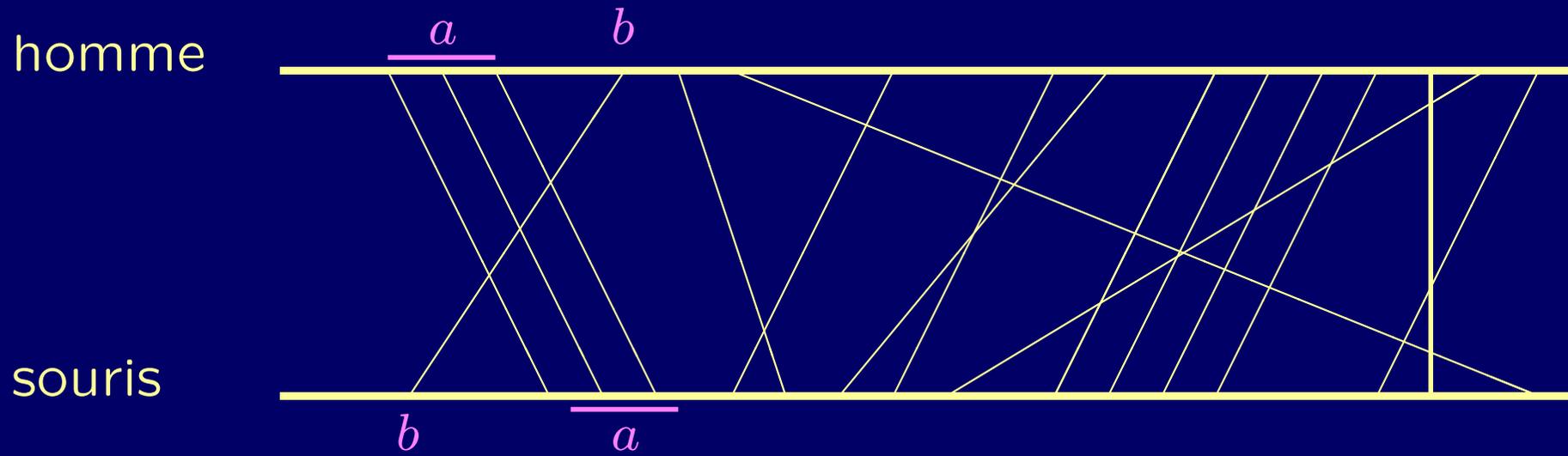
2. Réencodage



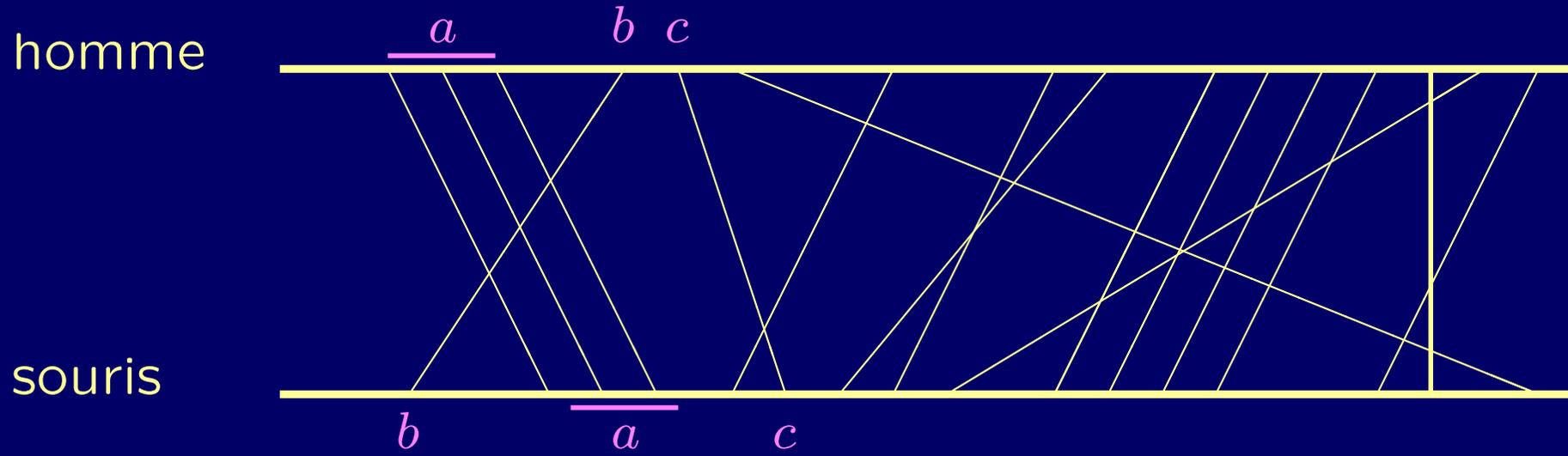
2. Réencodage



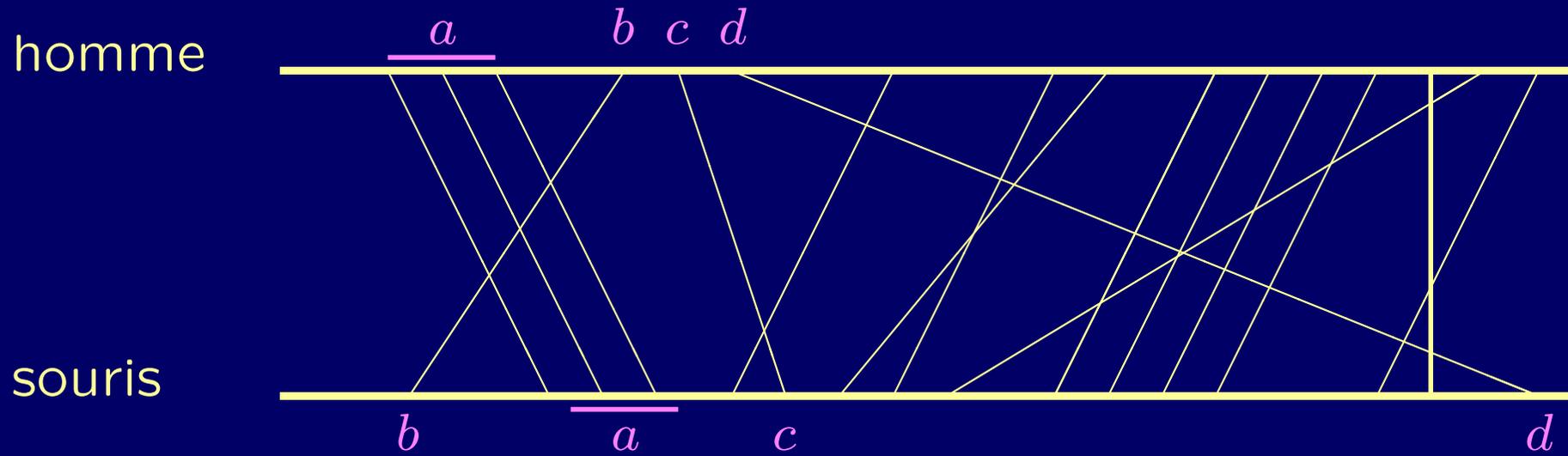
2. Réencodage



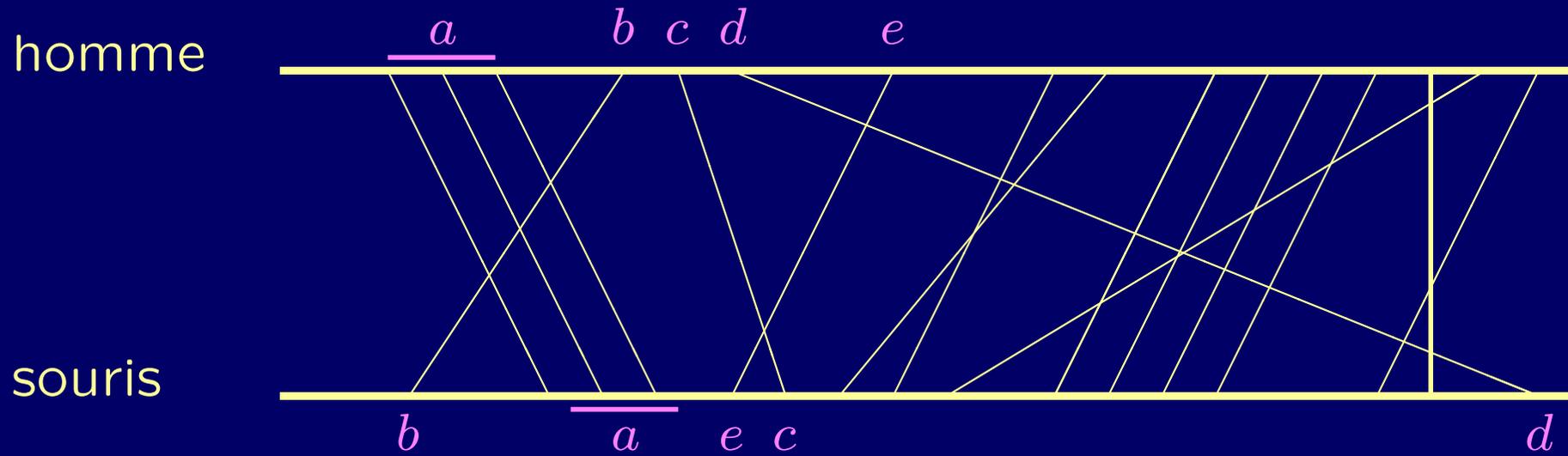
2. Réencodage



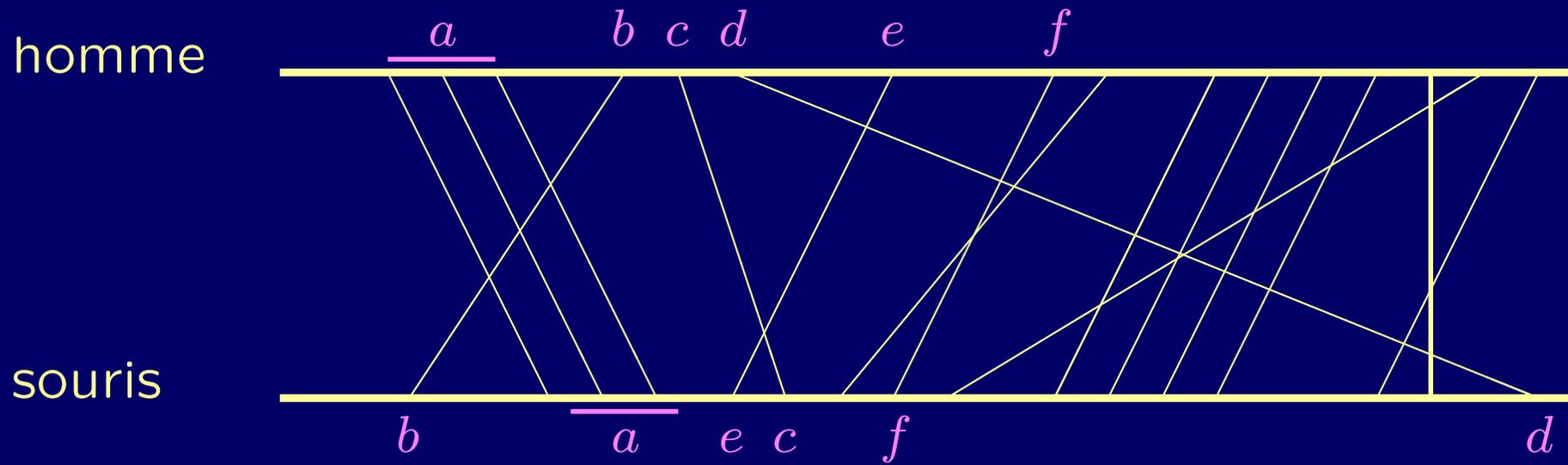
2. Réencodage



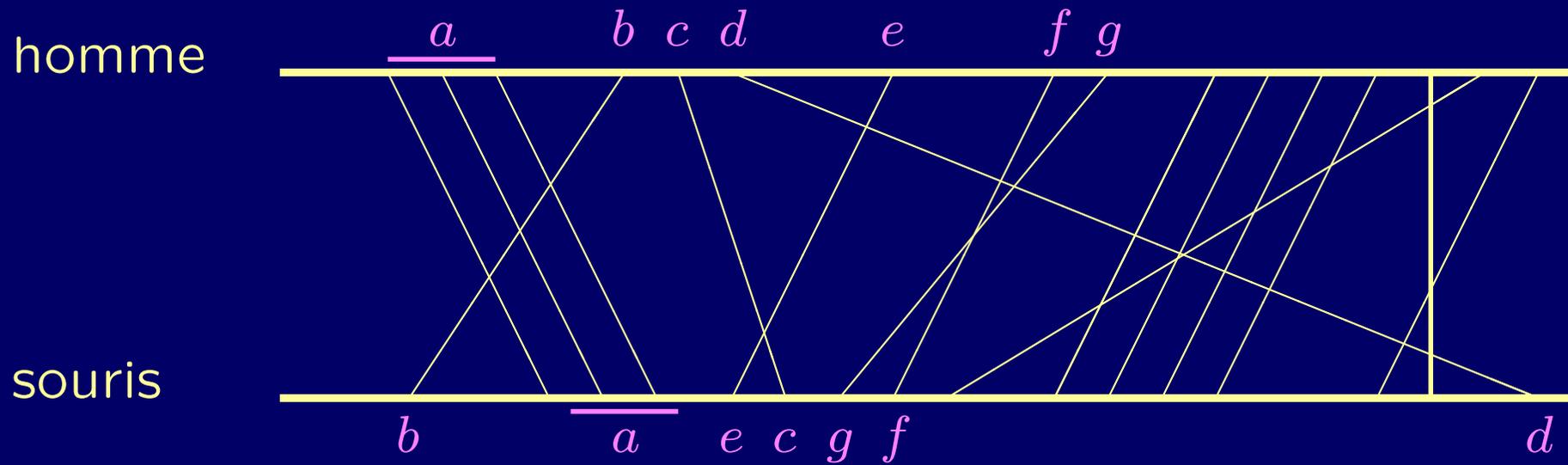
2. Réencodage



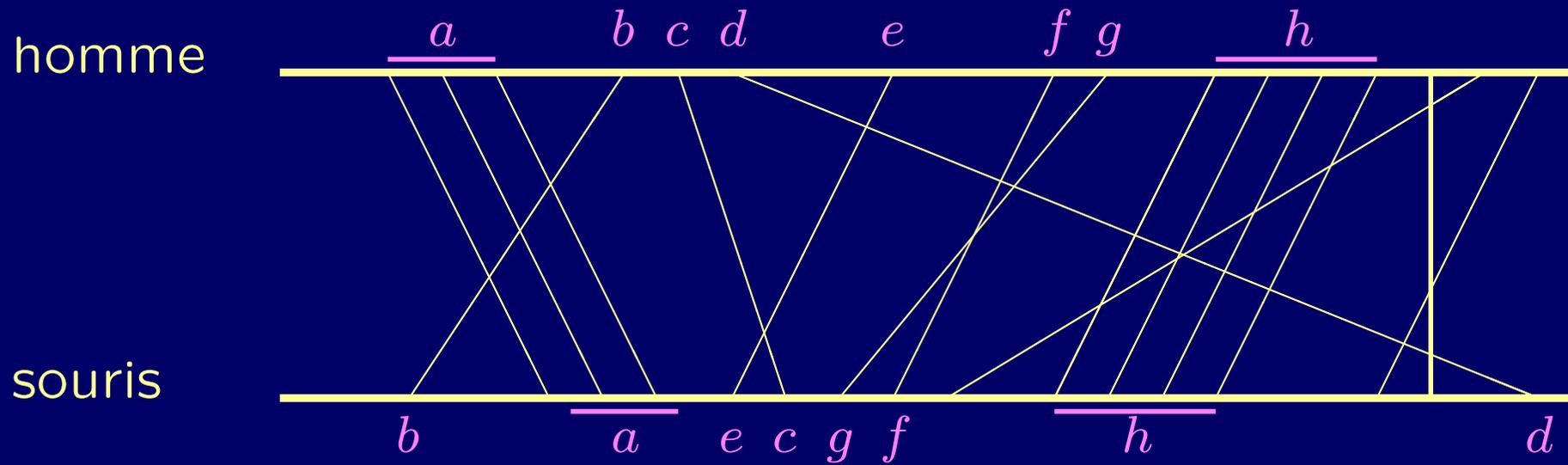
2. Réencodage



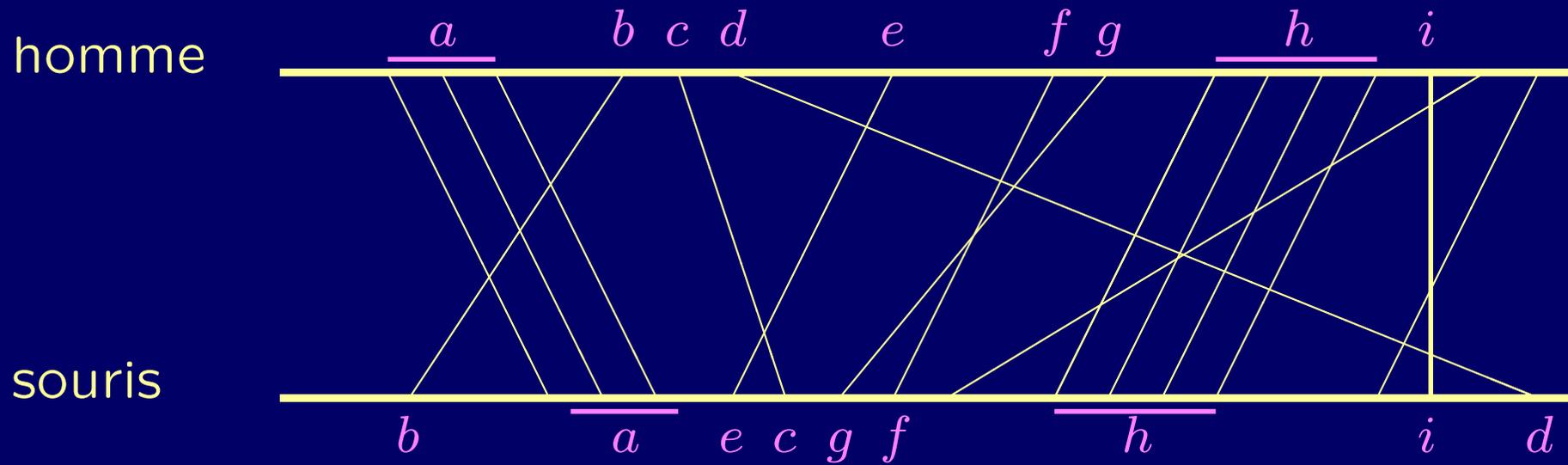
2. Réencodage



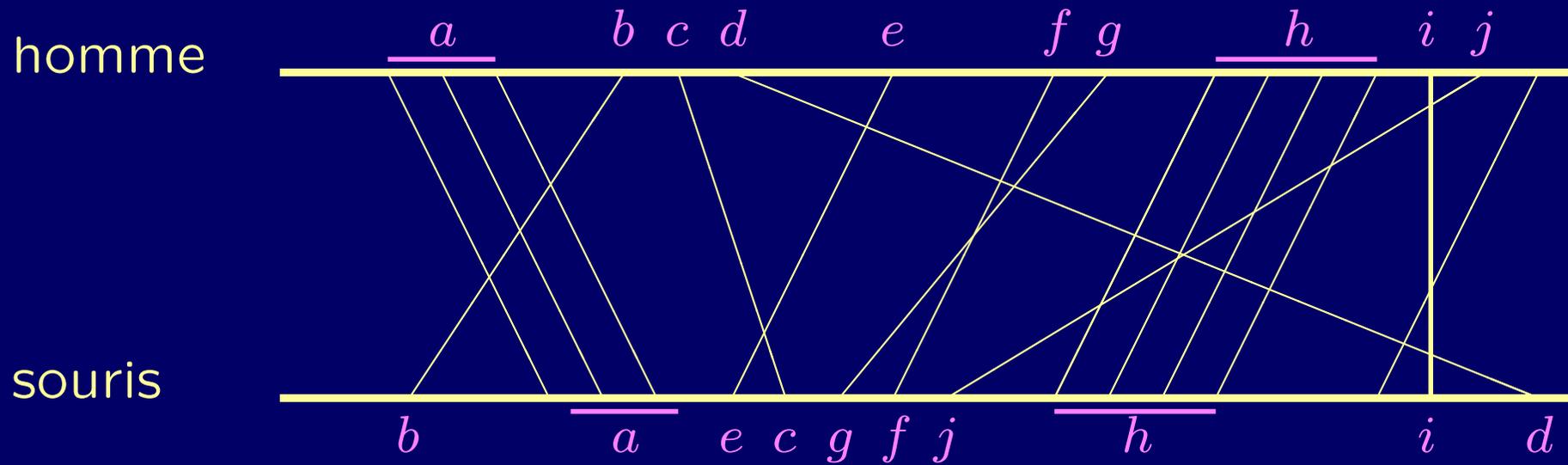
2. Réencodage



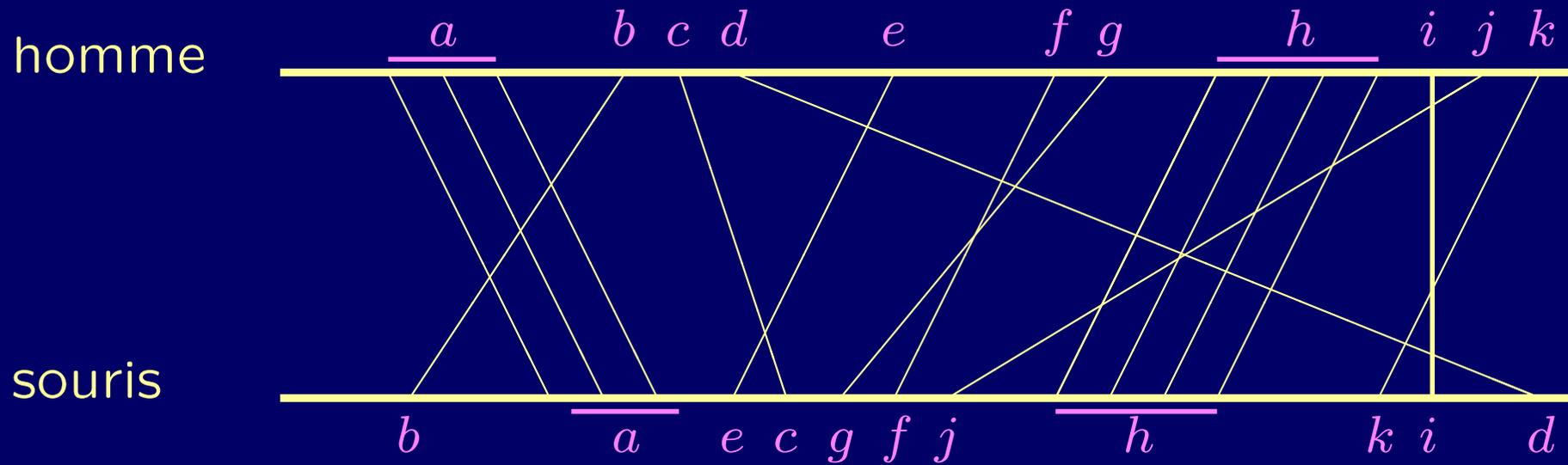
2. Réencodage



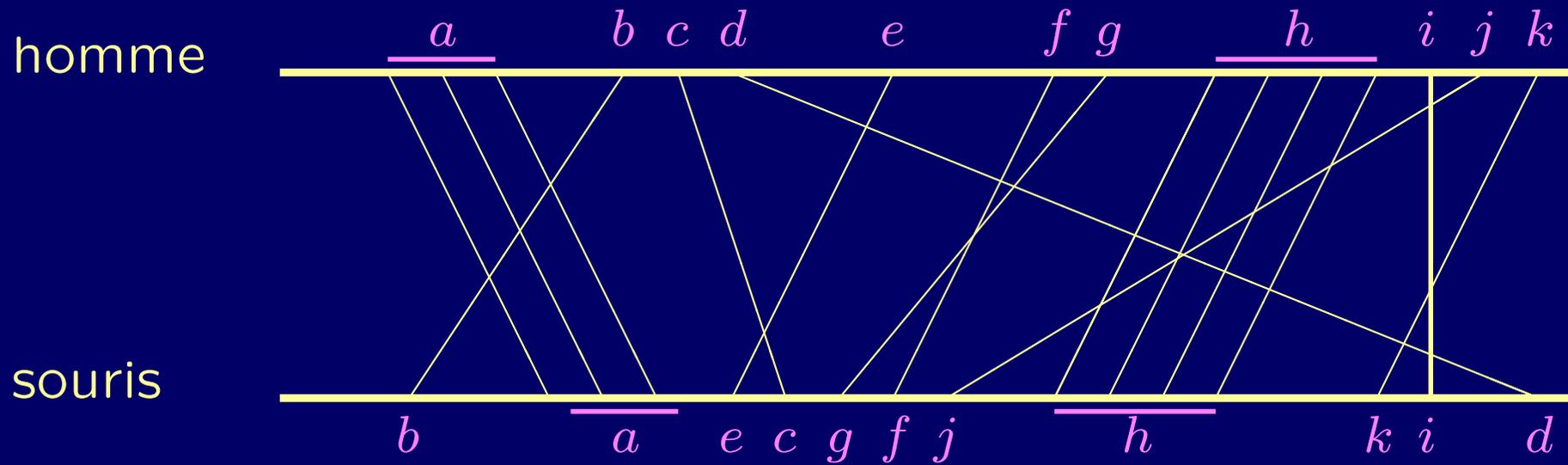
2. Réencodage



2. Réencodage

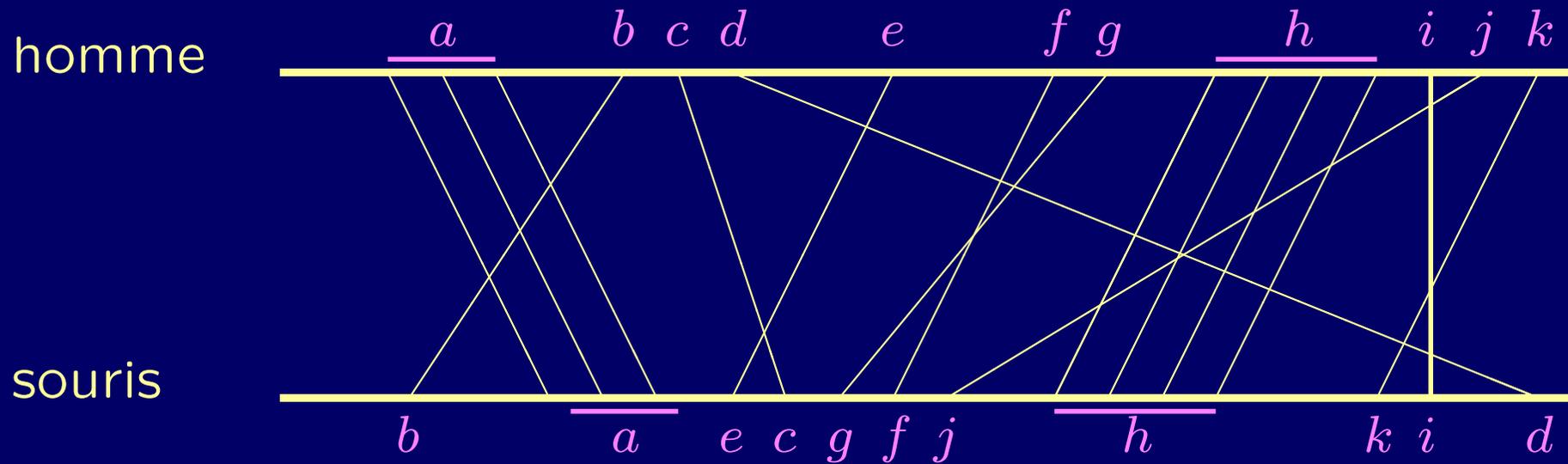


2. Réencodage



homme = (*a b c d e f g h i j k*)

2. Réencodage



homme = $(a\ b\ c\ d\ e\ f\ g\ h\ i\ j\ k)$

souris = $(b\ a\ e\ c\ g\ f\ j\ h\ k\ i\ d)$

- Opérations sur un seul chromosome

- Délétion

- Insertion

- Duplication

- Inversion (Reversal)

- Transposition

- Transposition inverse

- Opérations sur deux chromosomes

- Translocations

- Fusion

- Fission

- Opérations sur un seul chromosome

- Délétion $a b c \rightarrow a c$

- Insertion

- Duplication

- Inversion (Reversal)

- Transposition

- Transposition inverse

- Opérations sur deux chromosomes

- Translocations

- Fusion

- Fission

- Opérations sur un seul chromosome

- Délétion

$a b c \rightarrow a c$

- Insertion

$a b c \rightarrow a b d c$

- Duplication

- Inversion (Reversal)

- Transposition

- Transposition inverse

- Opérations sur deux chromosomes

- Translocations

- Fusion

- Fission

- Opérations sur un seul chromosome

- Délétion

$a b c \rightarrow a c$

- Insertion

$a b c \rightarrow a b d c$

- Duplication

en tandem : $a b c \rightarrow a b b b c$

- Inversion (Reversal)

- Transposition

- Transposition inverse

- Opérations sur deux chromosomes

- Translocations

- Fusion

- Fission

- Opérations sur un seul chromosome

- Délétion

$a b c \rightarrow a c$

- Insertion

$a b c \rightarrow a b d c$

- Duplication

ou non : $a b c d \rightarrow a b c b d b$

- Inversion (Reversal)

- Transposition

- Transposition inverse

- Opérations sur deux chromosomes

- Translocations

- Fusion

- Fission

- Opérations sur un seul chromosome

→ Délétion

$$a b c \rightarrow a c$$

→ Insertion

$$a b c \rightarrow a b d c$$

→ Duplication

$$\text{ou non : } a b c d \rightarrow a b c b d b$$

→ Inversion (Reversal)

$$a b_1 b_2 b_3 c \rightarrow a b_3 b_2 b_1 c$$

→ Transposition

→ Transposition inverse

- Opérations sur deux chromosomes

→ Translocations

→ Fusion

→ Fission

- Opérations sur un seul chromosome

→ Délétion

$$a b c \rightarrow a c$$

→ Insertion

$$a b c \rightarrow a b d c$$

→ Duplication

$$\text{ou non : } a b c d \rightarrow a b c b d b$$

→ Inversion (Reversal)

$$a b_1 b_2 b_3 c \rightarrow a b_3 b_2 b_1 c$$

→ Transposition

$$a b c d e \rightarrow a d b c e$$

→ Transposition inverse

$$a b c d e \rightarrow a d c b e$$

- Opérations sur deux chromosomes

→ Translocations

→ Fusion

→ Fission

- Opérations sur un seul chromosome

→ Délétion

$$a b c \rightarrow a c$$

→ Insertion

$$a b c \rightarrow a b d c$$

→ Duplication

$$\text{ou non : } a b c d \rightarrow a b c b d b$$

→ Inversion (Reversal)

$$a b_1 b_2 b_3 c \rightarrow a b_3 b_2 b_1 c$$

→ Transposition

$$a b c d e \rightarrow a d b c e$$

→ Transposition inverse

$$a b c d e \rightarrow a d c b e$$

- Opérations sur deux chromosomes

→ Translocations $a b c d ; \alpha \beta \chi \rightarrow a b \chi ; \alpha \beta c d$

→ Fusion

→ Fission

- Opérations sur un seul chromosome

→ Délétion

$$a b c \rightarrow a c$$

→ Insertion

$$a b c \rightarrow a b d c$$

→ Duplication

$$\text{ou non : } a b c d \rightarrow a b c b d b$$

→ Inversion (Reversal)

$$a b_1 b_2 b_3 c \rightarrow a b_3 b_2 b_1 c$$

→ Transposition

$$a b c d e \rightarrow a d b c e$$

→ Transposition inverse

$$a b c d e \rightarrow a d c b e$$

- Opérations sur deux chromosomes

→ Translocations $a b c d ; \alpha \beta \chi \rightarrow a b \chi ; \alpha \beta c d$

→ Fusion $a b c d ; \alpha \beta \chi \rightarrow a b c d \alpha \beta \chi$

→ Fission

- Opérations sur un seul chromosome

→ Délétion

$$a b c \rightarrow a c$$

→ Insertion

$$a b c \rightarrow a b d c$$

→ Duplication

$$\text{ou non : } a b c d \rightarrow a b c b d b$$

→ Inversion (Reversal)

$$a b_1 b_2 b_3 c \rightarrow a b_3 b_2 b_1 c$$

→ Transposition

$$a b c d e \rightarrow a d b c e$$

→ Transposition inverse

$$a b c d e \rightarrow a d c b e$$

- Opérations sur deux chromosomes

→ Translocations $a b c d ; \alpha \beta \chi \rightarrow a b \chi ; \alpha \beta c d$

→ Fusion $a b c d ; \alpha \beta \chi \rightarrow a b c d \alpha \beta \chi$

→ Fission $a b c d e \rightarrow a b c ; d e$

- Étant donnés 2 génomes

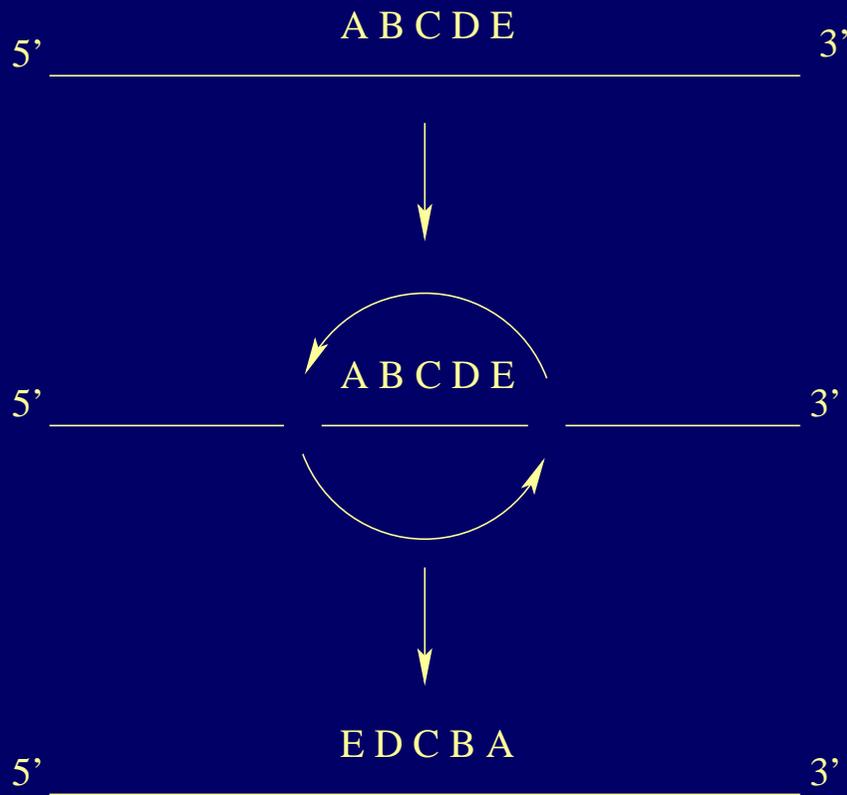
Trouver un scénario évolutif permettant de passer de l'un à l'autre

- Intérêt pour

→ la longueur de ce scénario (= nb d'événements évolutifs)
comme une mesure de distance évolutive

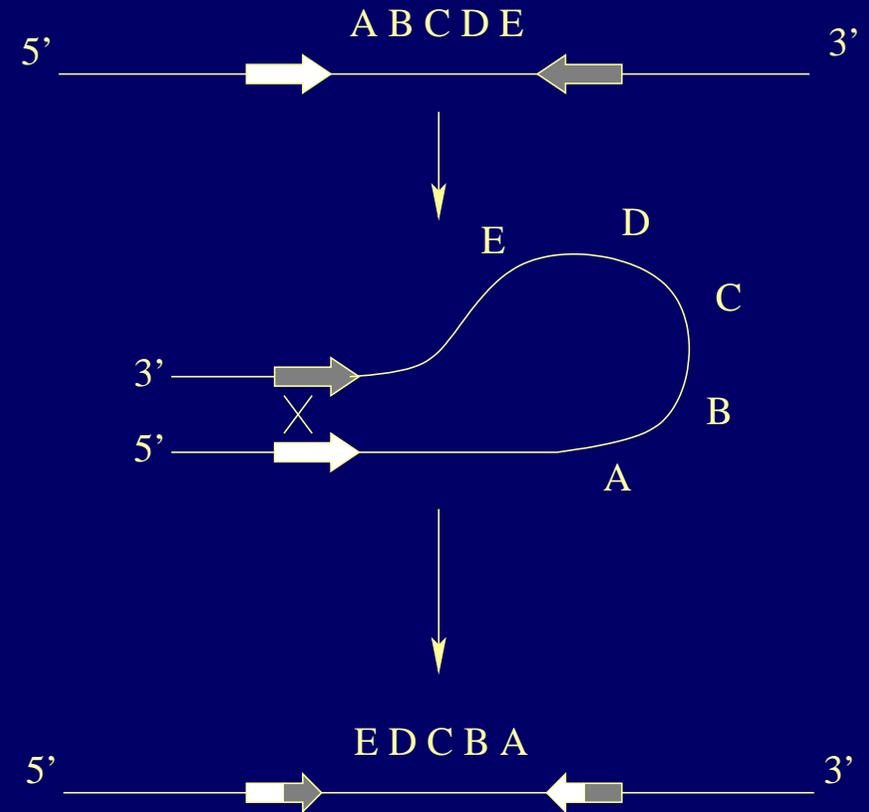
→ les réarrangements impliqués dans ce scénario

- On se restreint à l'inversion, un réarrangement très fréquent



Cassure

+ mauvais réassemblage



Crossing-over

intrachromosomique

(Graur et Li, 2000)

- L'ensemble des gènes est représenté par l'ensemble $\{1, 2, \dots, n\}$
- Les gènes sont **signés** : $+i$ ou $-i$, suivant le brin à partir duquel il est transcrit
- Les génomes sont représentés par des **permutations signée** sur l'ensemble $\{1, 2, \dots, n\}$
- L'**inversion** d'un intervalle de la permutation inverse l'**ordre** et l'**orientation** des éléments de cet intervalle

Le problème **SBR** : Étant données 2 permutations signées, trouver un scénario d'inversion de longueur minimale pour passer de l'une à l'autre.

Trouver la longueur de ce scénario est le problème de la **distance d'inversion**.

- Sans perte de généralité, on considère qu'une des 2 permutations est l'identité (renommage)
⇒ d'où le nom de **Sorting By Reversals**
- Permutation signée sur $\{1, 2, \dots, 8\}$:

$$\pi_1 = \quad -2 \quad -1 \quad 4 \quad 3 \quad 5 \quad -8 \quad 6 \quad 7$$

- Sans perte de généralité, on considère qu'une des 2 permutations est l'identité (renommage)
⇒ d'où le nom de Sorting By Reversals
- Permutation signée sur $\{1, 2, \dots, 8\}$:

$$\pi_1 = (0 \ -2 \ -1 \ 4 \ 3 \ 5 \ -8 \ 6 \ 7 \ 9).$$

- Sans perte de généralité, on considère qu'une des 2 permutations est l'identité (renommage)
⇒ d'où le nom de **Sorting By Reversals**
- Permutation signée sur $\{1, 2, \dots, 8\}$:

$$\pi_1 = (0 \ -2 \ -1 \ 4 \ 3 \ 5 \ -8 \ 6 \ 7 \ 9).$$

L'**inversion** de l'intervalle $[2..6]$ de π_1 , notée $\rho(2, 6)$, donne la permutation $\pi_2 = \pi_1 \cdot \rho(2, 6)$:

(0 -2 -1 4 3 5 -8 6 7 9)

Trier $\pi_1 = (0 -2 -1 4 3 5 -8 6 7 9)$ par inversions.

(0 -2 -1 4 3 5 -8 6 7 9)



Trier $\pi_1 = (0 - 2 - 1 4 3 5 - 8 6 7 9)$ par inversions.

(0 -2 -1 4 3 5 -8 6 7 9)



(0 -4 1 2 3 5 -8 6 7 9)

Trier $\pi_1 = (0 - 2 - 1 4 3 5 - 8 6 7 9)$ par inversions.

(0 -2 -1 4 3 5 -8 6 7 9)



(0 -4 1 2 3 5 -8 6 7 9)



Trier $\pi_1 = (0 -2 -1 4 3 5 -8 6 7 9)$ par inversions.

(0 -2 -1 4 3 5 -8 6 7 9)



(0 -4 1 2 3 5 -8 6 7 9)



(0 -3 -2 -1 4 5 -8 6 7 9)

Trier $\pi_1 = (0 -2 -1 4 3 5 -8 6 7 9)$ par inversions.

(0 -2 -1 4 3 5 -8 6 7 9)



(0 -4 1 2 3 5 -8 6 7 9)



(0 -3 -2 -1 4 5 -8 6 7 9)



Trier $\pi_1 = (0 -2 -1 4 3 5 -8 6 7 9)$ par inversions.

(0 -2 -1 4 3 5 -8 6 7 9)



(0 -4 1 2 3 5 -8 6 7 9)



(0 -3 -2 -1 4 5 -8 6 7 9)



(0 1 2 3 4 5 -8 6 7 9)

Trier $\pi_1 = (0 -2 -1 4 3 5 -8 6 7 9)$ par inversions.

(0 -2 -1 4 3 5 -8 6 7 9)



(0 -4 1 2 3 5 -8 6 7 9)



(0 -3 -2 -1 4 5 -8 6 7 9)



(0 1 2 3 4 5 -8 6 7 9)



Trier $\pi_1 = (0 -2 -1 4 3 5 -8 6 7 9)$ par inversions.

(0 -2 -1 4 3 5 -8 6 7 9)



(0 -4 1 2 3 5 -8 6 7 9)



(0 -3 -2 -1 4 5 -8 6 7 9)



(0 1 2 3 4 5 -8 6 7 9)



(0 1 2 3 4 5 -7 -6 8 9)

Trier $\pi_1 = (0 -2 -1 4 3 5 -8 6 7 9)$ par inversions.

(0 -2 -1 4 3 5 -8 6 7 9)



(0 -4 1 2 3 5 -8 6 7 9)



(0 -3 -2 -1 4 5 -8 6 7 9)



(0 1 2 3 4 5 -8 6 7 9)



(0 1 2 3 4 5 -7 -6 8 9)



Trier $\pi_1 = (0 -2 -1 4 3 5 -8 6 7 9)$ par inversions.

(0 -2 -1 4 3 5 -8 6 7 9)



(0 -4 1 2 3 5 -8 6 7 9)



(0 -3 -2 -1 4 5 -8 6 7 9)



(0 1 2 3 4 5 -8 6 7 9)



(0 1 2 3 4 5 -7 -6 8 9)



(0 1 2 3 4 5 6 7 8 9)

Trier $\pi_1 = (0 -2 -1 4 3 5 -8 6 7 9)$ par inversions.

(0 -2 -1 4 3 5 -8 6 7 9)



(0 -4 1 2 3 5 -8 6 7 9)



(0 -3 -2 -1 4 5 -8 6 7 9)



(0 1 2 3 4 5 -8 6 7 9)



(0 1 2 3 4 5 -7 -6 8 9)



(0 1 2 3 4 5 6 7 8 9)

Trier $\pi_1 = (0 -2 -1 4 3 5 -8 6 7 9)$ par inversions.

$$d(\pi_1) = 5$$

- 1992 Sankoff 1re formulation du problème
- 1995 Hannenhalli et Pevzner "*Transforming Cabbage into Turnip*"
1er algorithme polynomial & LA théorie d'H.-P.
- 1996 Berman et Hannenhalli Amélioration de la complexité
- 1999 Kaplan, Shamir et Tarjan
"*A Faster and Simpler Algorithm for SBR*"
- 2001 Bergeron
"*A Very Elementary Presentation of the H-P Theory*"
- 2001 Bader, Moret et Yan
Algorithme linéaire pour le problème de la distance
- 2004 Tannier et Sagot
"*Sorting by reversals in subquadratic time*"

- Anne Bergeron, Julia Mixtacki et Jens Stoye
"The inversion distance problem"
Chapitre 10 du livre *Mathematics of Evolution and Phylogeny* 2005
(ed. O. Gascuel)
- Michal Ozery-Flato et Ron Shamir
"Two notes on genome rearrangement"
Dans *Journal of Bioinformatics and Computational Biology* 2003

-
- Introduction
 - Anatomie d'une permutation signée
 - Les points
 - Le graphe de breakpoint et la théorie d'H.-P.
 - Les intervalles élémentaires
 - Les cycles
 - Le graphe de chevauchement
 - Les effets d'une inversion
 - Les algorithmes
 - Conclusion et perspectives

$$\pi_1 = (0 \ -2 \ -1 \ 4 \ 3 \ 5 \ -8 \ 6 \ 7 \ 9).$$

Un point $p \cdot q$ est défini par une paire d'éléments consécutifs dans la permutation.

$$\pi_1 = (0 \ -2 \ -1 \ 4 \ 3 \ 5 \ -8 \ 6 \ 7 \ 9).$$

Un point $p \cdot q$ est défini par une paire d'éléments consécutifs dans la permutation.

Exemple : $0 \cdot -2$ et $-2 \cdot -1$ sont les deux premiers points de π_1 .

$$\pi_1 = (0 \ -2 \ -1 \ 4 \ 3 \ 5 \ -8 \ 6 \ 7 \ 9).$$

Un point $p \cdot q$ est défini par une paire d'éléments consécutifs dans la permutation.

Exemple : $0 \cdot -2$ et $-2 \cdot -1$ sont les deux premiers points de π_1 .

Un point de la forme $i \cdot i + 1$, ou $-(i + 1) \cdot -i$ est une **adjacence**, autrement c'est un **breakpoint**.

$$\pi_1 = (0 \ -2 \ -1 \ 4 \ 3 \ 5 \ -8 \ 6 \ 7 \ 9).$$

Un point $p \cdot q$ est défini par une paire d'éléments consécutifs dans la permutation.

Exemple : $0 \cdot -2$ et $-2 \cdot -1$ sont les deux premiers points de π_1 .

Un point de la forme $i \cdot i + 1$, ou $-(i + 1) \cdot -i$ est une **adjacence**, autrement c'est un **breakpoint**.

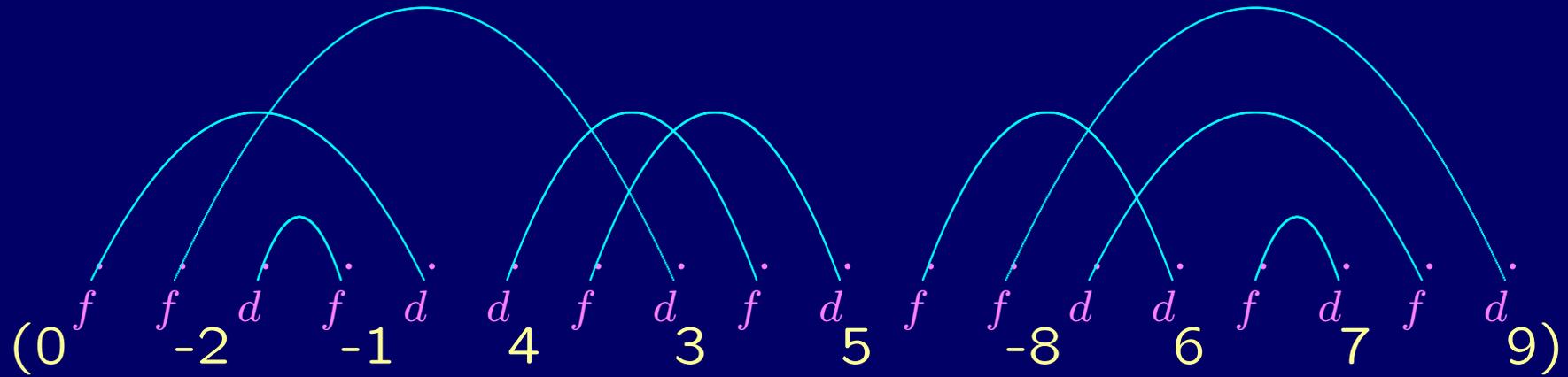
Exemple : $-2 \cdot -1$ et $6 \cdot 7$ sont des adjacences.

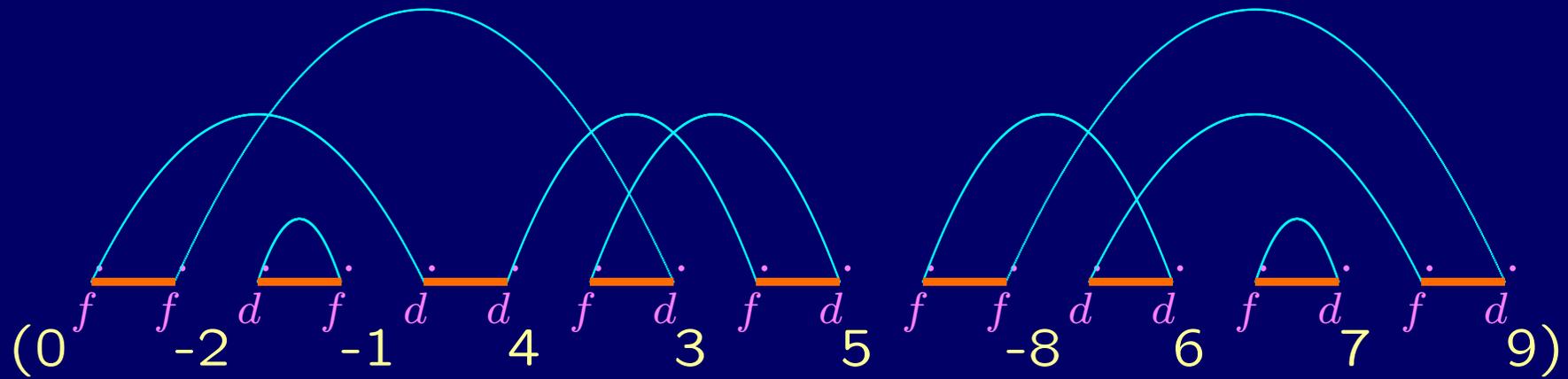
Tous les autres points de π_1 sont des breakpoints.

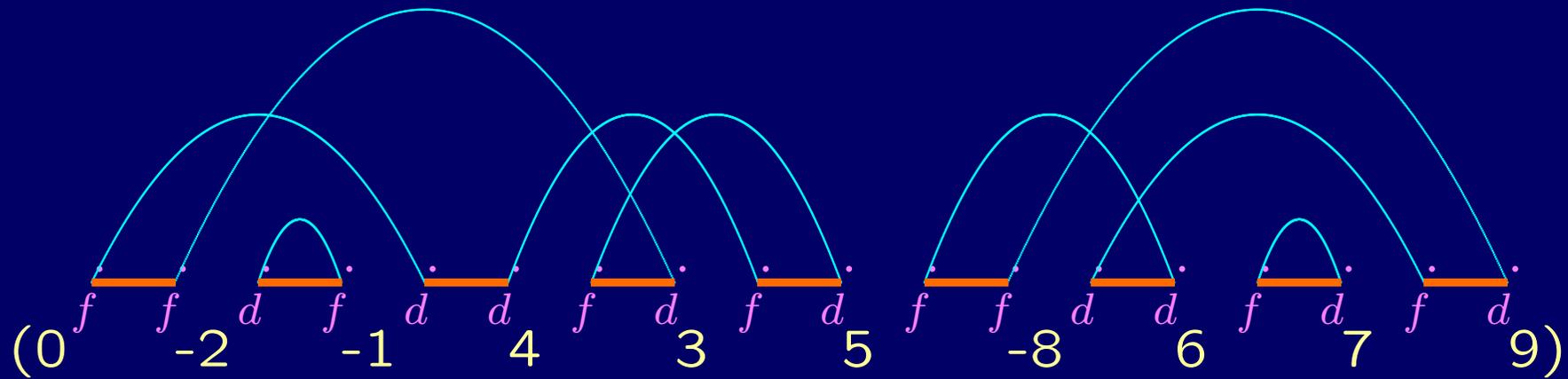
(0 -2 -1 4 3 5 -8 6 7 9)



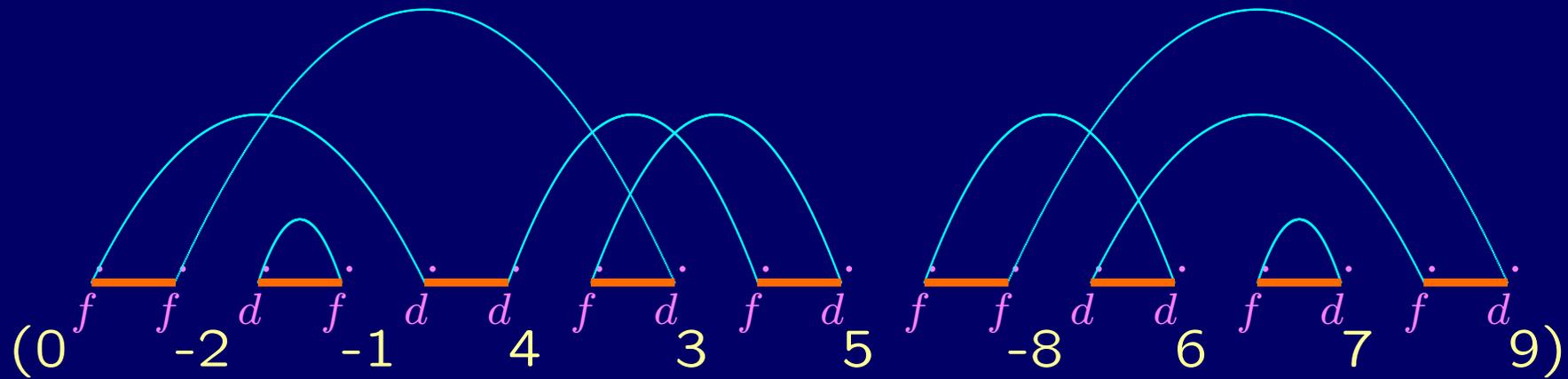
(0^f -2^f -1^d 4^f 3^d 5^f -8^d 6^f 7^d 9^f)







- Arêtes du désir : traits fins et bleu



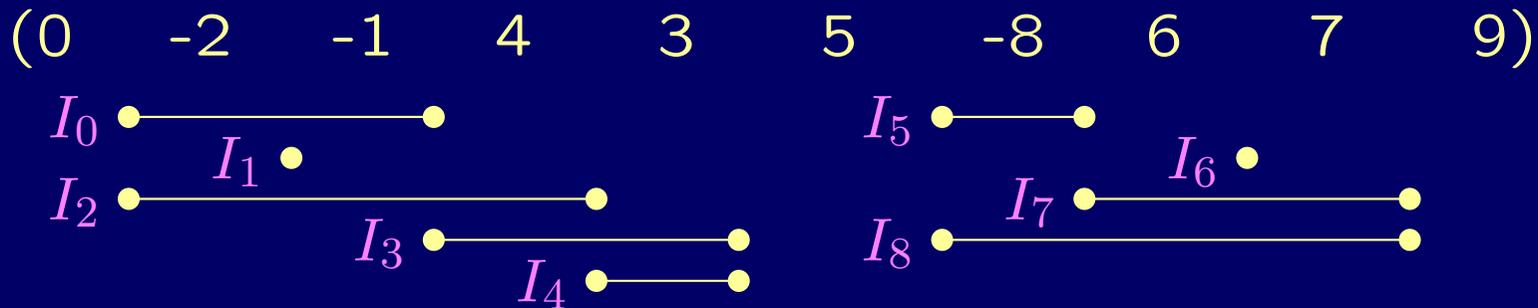
- Arêtes du désir : traits fins et bleu
- Arêtes de la réalité : traits épais et orange

$$d(\pi) = n - c(\pi) + h(\pi) + f(\pi)$$

- $c(\pi)$: nombre de **cycles** de la permutation π
- $h(\pi)$: nombre de **hurdles** de la permutation π
- $f(\pi) = 1$ si π est une **forteresse**, $f(\pi) = 0$ sinon

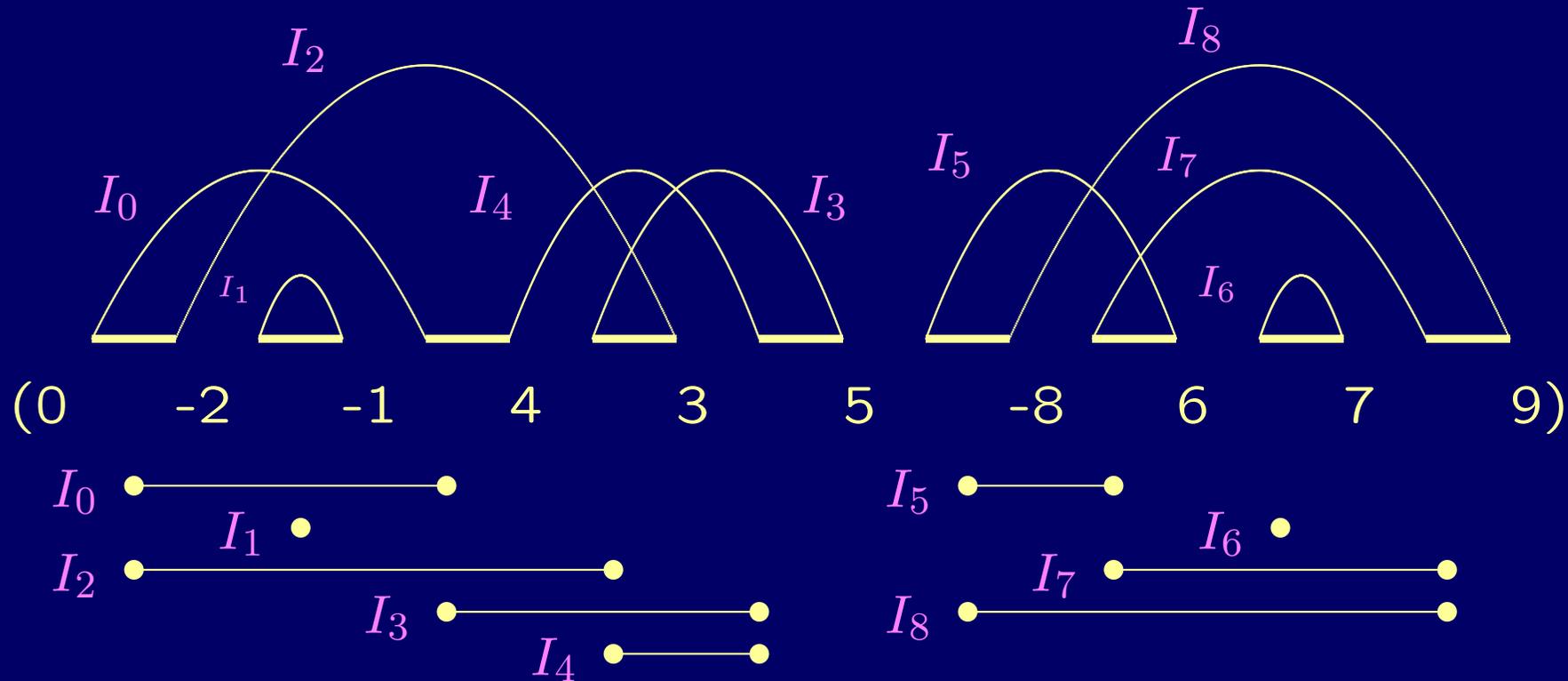
À chaque paire d'éléments non signés $(k, k + 1)$, $0 \leq k < n$, on associe l'intervalle élémentaire I_k dont les extrémités sont :

1. Le point droit de k , si k est positif, son point gauche sinon.
2. Le point gauche de $k + 1$, si $k + 1$ est positif, son point droit sinon.

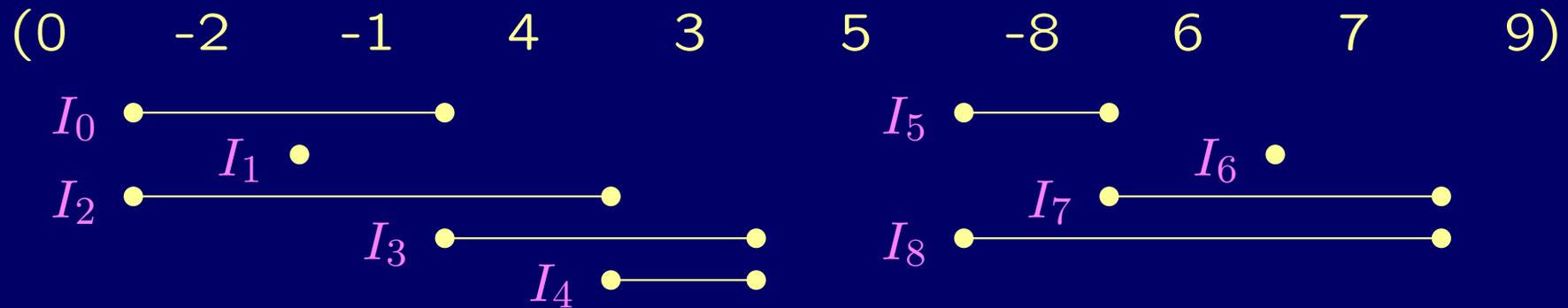


Les éléments k et $k + 1$ sont appelés les **extrémités** de l'intervalle élémentaire.

Intervalles élémentaires et le graphe de breakpoint 19



Intervalles élémentaires = arêtes du désir

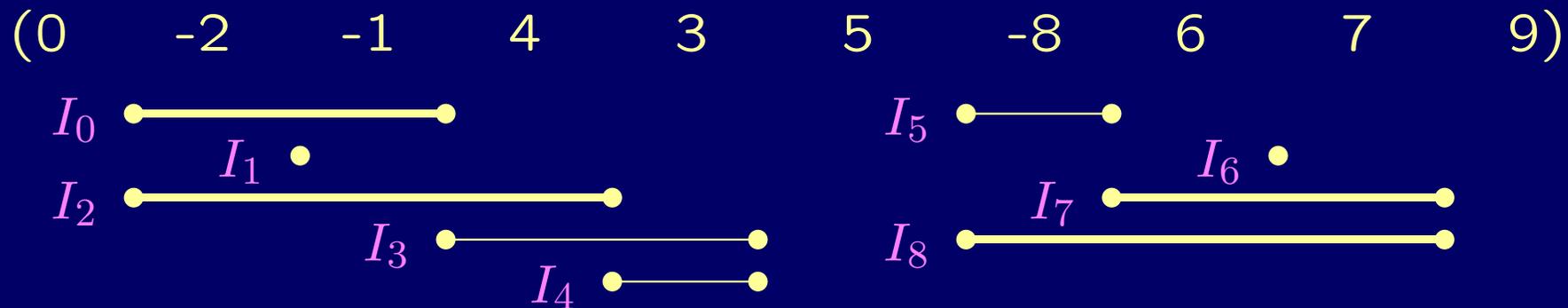


Un intervalle élémentaire peut contenir 0, 1 ou 2 de ses extrémités.

Exemple :

- I_0 contient une de ses extrémités,
- I_3 contient les deux,
- I_5 n'en contient aucune.

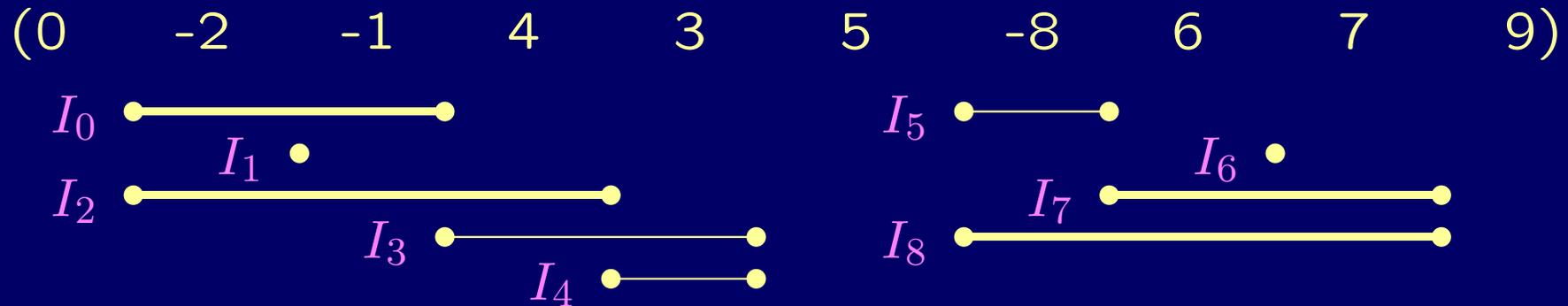
Les intervalles élémentaires vides, tels que I_1 et I_6 , correspondent à des adjacences de la permutation.



Un intervalle dont les extrémités ont des signes différents est dit **orienté**, sinon il est **non orienté**.

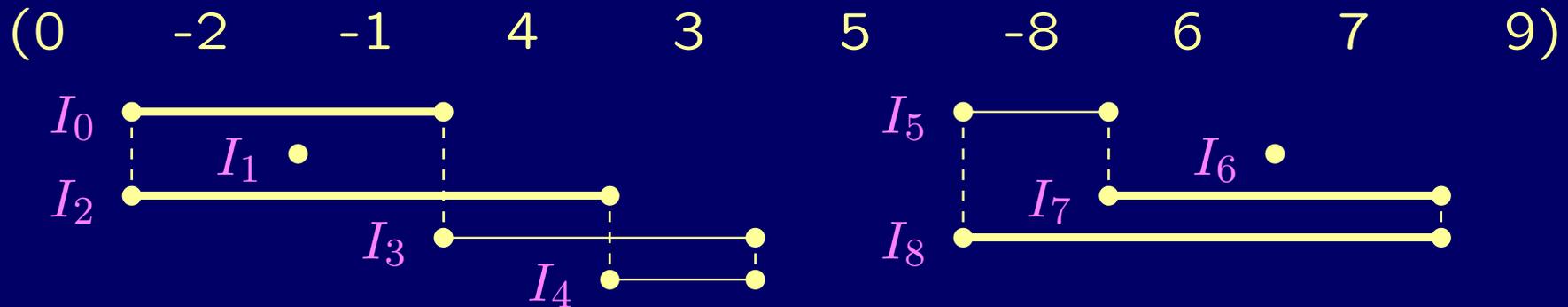
Les intervalles orientés sont exactement ceux qui contiennent **une seule** de leurs extrémités.

Les intervalles orientés jouent un rôle crucial puisqu'ils peuvent être utilisés pour créer des adjacences.



Inverser un intervalle orienté I_k crée, dans la permutation résultante, soit l'adjacence $k \cdot k + 1$ soit l'adjacence $-(k + 1) \cdot -k$.

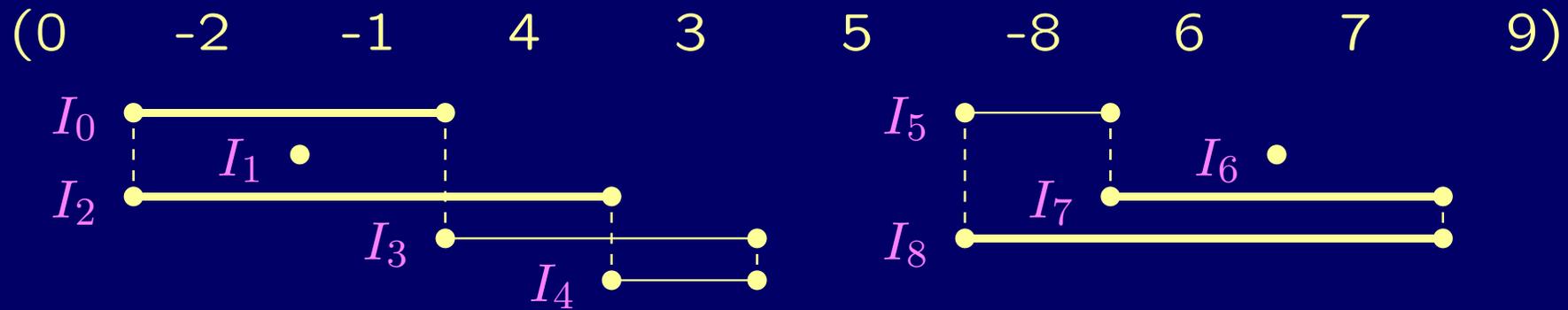
Exactement 2 intervalles élémentaires se rencontrent à chaque breakpoint de la permutation.



Un **cycle** est une séquence b_1, b_2, \dots, b_k de points tels que 2 points successifs soient les extrémités d'un intervalle élémentaire, y compris b_k et b_1 .

Les adjacences sont des cycles triviaux consistant en un seul point.

Un cycle contient toujours un nombre pair intervalles orientés.



Deux intervalles élémentaires I et J se chevauchent si chacun contient exactement une des extrémités de l'autre.

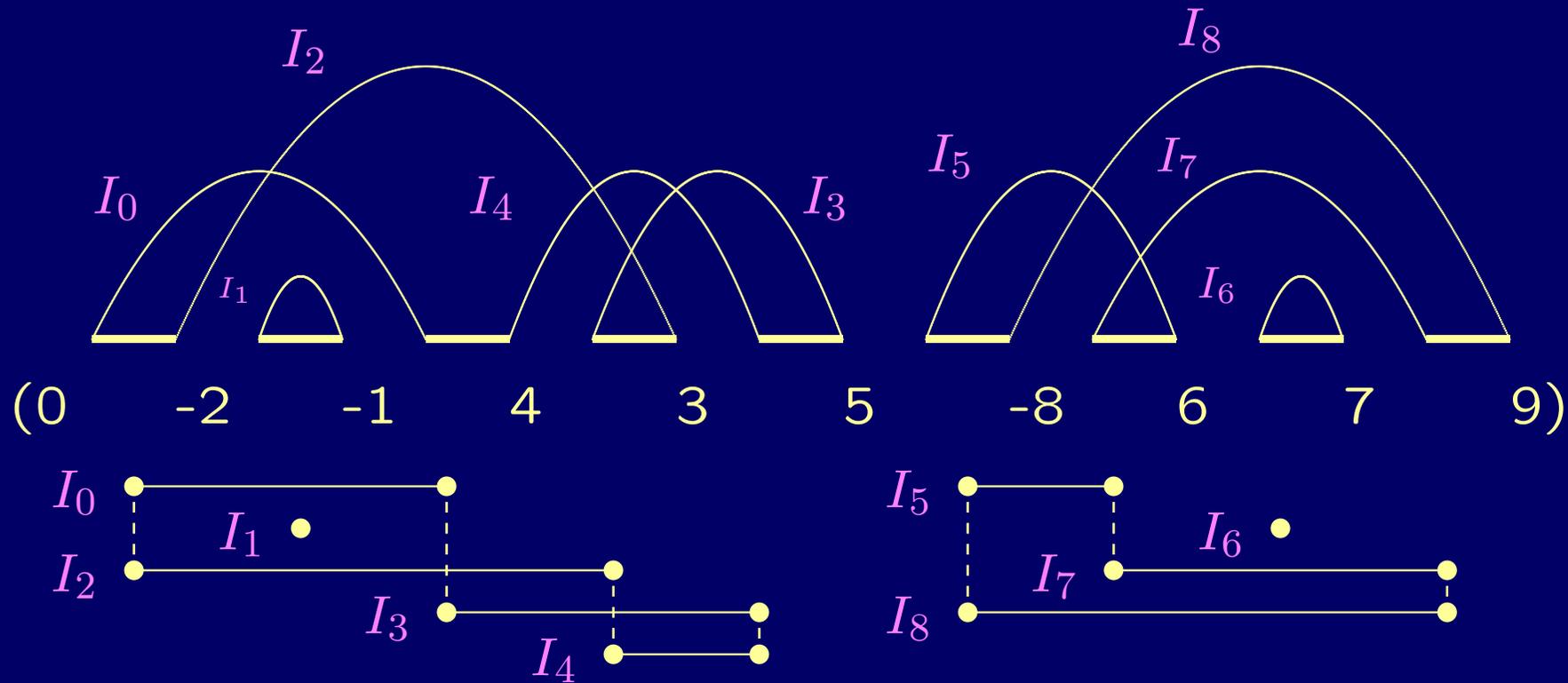
Les intervalles qui se rencontrent à un breakpoint peuvent se chevaucher ou non.

Exemples :

→ I_0 et I_2 se chevauchent (I_0 contient -2, et I_2 contient 1)

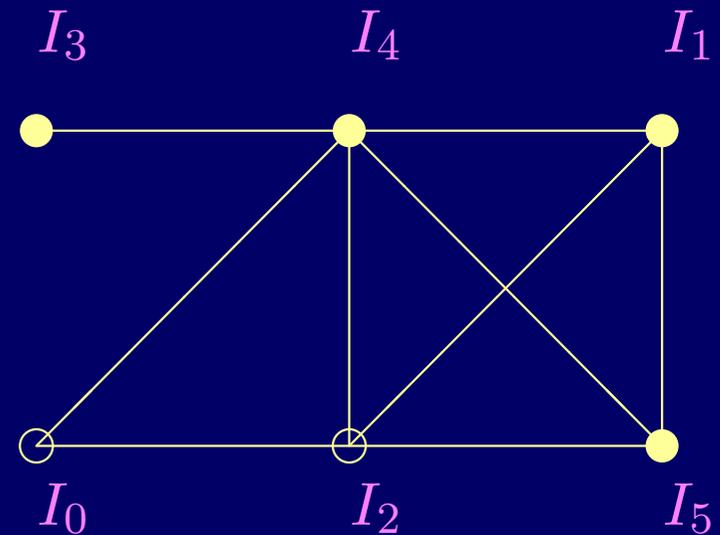
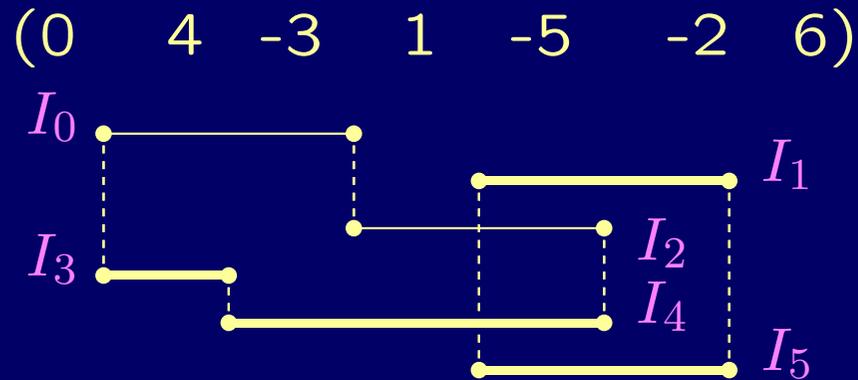
→ I_0 et I_3 ne se chevauchent pas

→ I_3 et I_4 se chevauchent (I_3 contient 4, et I_4 contient 3)



Exemples :

- I_0 et I_2 se chevauchent (I_0 contient -3, et I_2 contient 1)
- I_0 et I_3 ne se chevauchent pas
- I_3 et I_4 se chevauchent (I_3 contient 4, et I_4 contient 3)



Une permutation et son graphe de chevauchement \mathcal{O}

Deux sommets sont connectés dans \mathcal{O} si et seulement si les intervalles correspondants se chevauchent.

- Sommets pleins pour les intervalles orientés \Leftrightarrow degrés impair
- Sommets vides pour les intervalles non orientés.

-
- Introduction
 - Anatomie d'une permutation signée
 - Les effets d'une inversion
 - Inversions et nombre de cycles
 - Inversion d'un intervalle orienté
 - Inversion et graphe de chevauchement
 - Les composants
 - Inversions et composants
 - Les algorithmes
 - Conclusion et perspectives

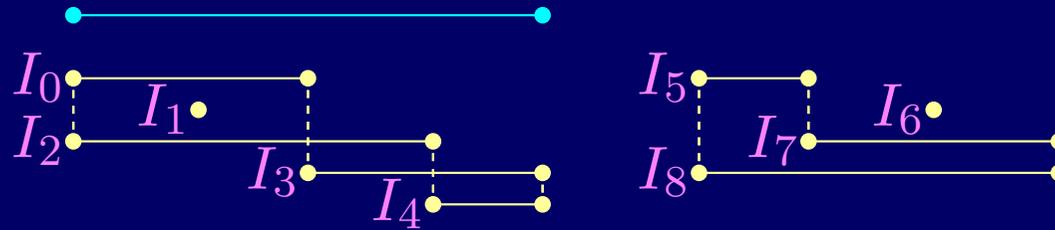
Une inversion peut seulement modifier le nombre de cycles par $+1$, 0 , ou -1 .

Preuve :

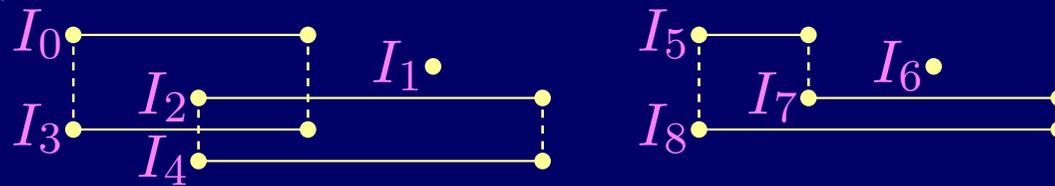
- Une inversion échange les éléments de deux points d'une permutation.
- Si ces deux points appartiennent au même cycle, alors
 - soit le cycle est **coupé** en deux, $(+1)$
 - soit il est **conservé** mais avec des breakpoints différents. (0)
- Si les deux points appartiennent à des cycles différents, alors
 - ces deux cycles sont **fusionnés**. (-1)

(Kececioglu et Sankoff, 1994)

$$\pi_a = (0 \quad -2 \quad -1 \quad 4 \quad 3 \quad 5 \quad -8 \quad 6 \quad 7 \quad 9)$$



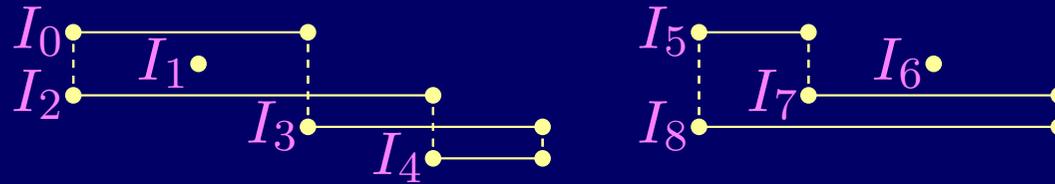
$$\pi_b = (0 \quad -3 \quad -4 \quad 1 \quad 2 \quad 5 \quad -8 \quad 6 \quad 7 \quad 9)$$



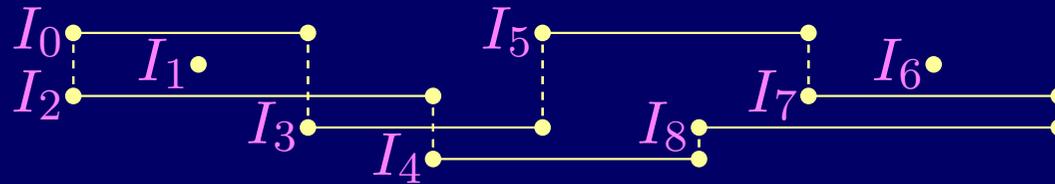
$$c(\pi_b) = c(\pi_a) + 1$$

Un cycle coupé en deux

$$\pi_a = (0 \quad -2 \quad -1 \quad 4 \quad 3 \quad 5 \quad -8 \quad 6 \quad 7 \quad 9)$$



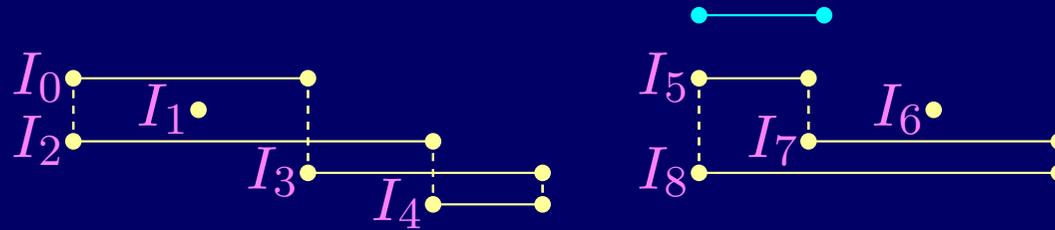
$$\pi_c = (0 \quad -2 \quad -1 \quad 4 \quad 3 \quad -5 \quad -8 \quad 6 \quad 7 \quad 9)$$



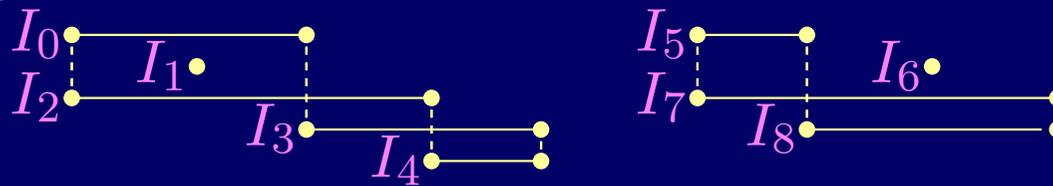
$$c(\pi_c) = c(\pi_a) - 1$$

Deux cycles fusionnés en un seul

$$\pi_a = (0 \ -2 \ -1 \ 4 \ 3 \ 5 \ -8 \ 6 \ 7 \ 9)$$



$$\pi_d = (0 \ -2 \ -1 \ 4 \ 3 \ 5 \ 8 \ 6 \ 7 \ 9)$$



$$c(\pi_d) = c(\pi_a)$$

Aucun changement dans le nombre de cycles

- Inverser un intervalle orienté crée une adjacence
- Une adjacence est un cycle trivial

- Inverser un intervalle orienté crée une adjacence
- Une adjacence est un cycle trivial

Inverser un intervalle orienté augmente de $+1$ le nombre de cycles .

- Inverser un intervalle orienté crée une adjacence
- Une adjacence est un cycle trivial

Inverser un intervalle orienté augmente de $+1$ le nombre de cycles .

- La permutation identité sur l'ensemble $\{0, 1, 2, \dots, n\}$ est la seule avec n cycles, tous des adjacences.
- Puisque au plus un cycle peut être ajouté par une inversion

- Inverser un intervalle orienté crée une adjacence
- Une adjacence est un cycle trivial

Inverser un intervalle orienté augmente de $+1$ le nombre de cycles .

- La permutation identité sur l'ensemble $\{0, 1, 2, \dots, n\}$ est la seule avec n cycles, tous des adjacences.
 - Puisque au plus un cycle peut être ajouté par une inversion
- ⇒ Une première **borne inférieure** pour la distance d'inversion :

$$d(\pi) \geq n - c(\pi)$$

Si I et J se chevauchent, alors inverser l'intervalle I change l'orientation de J , puisque une seule des extrémités de J change de signe.

Si I et J se chevauchent, alors inverser l'intervalle I change l'orientation de J , puisqu'une seule des extrémités de J change de signe.

Quand deux intervalles J et K chevauchent un intervalle I , l'effet de l'inversion de I est de **complémenter** la relation de chevauchement entre J et K :

- si J et K se chevauchaient avant l'inversion, ils ne se chevauchent plus après ;
- si J et K ne se chevauchaient pas avant, ils se chevauchent après.

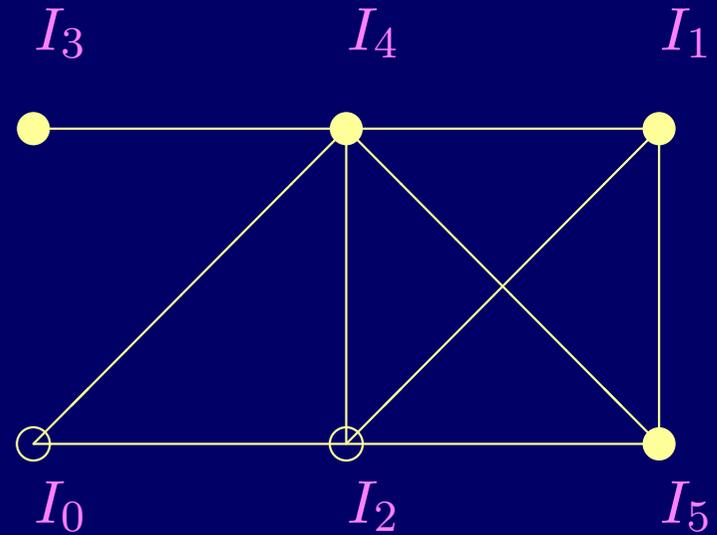
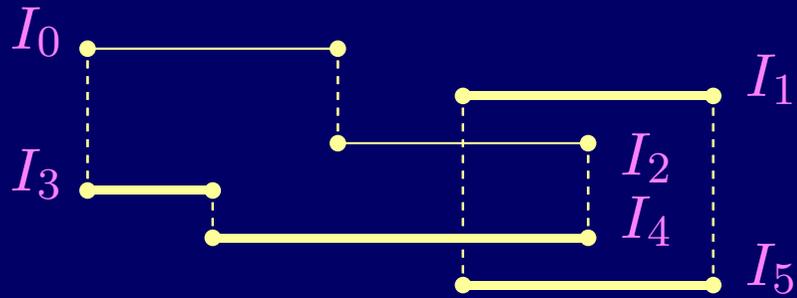
Formellement :

Soit G_I le sous-graphe du graphe de chevauchement formé par le sommet I et ses sommets voisins.

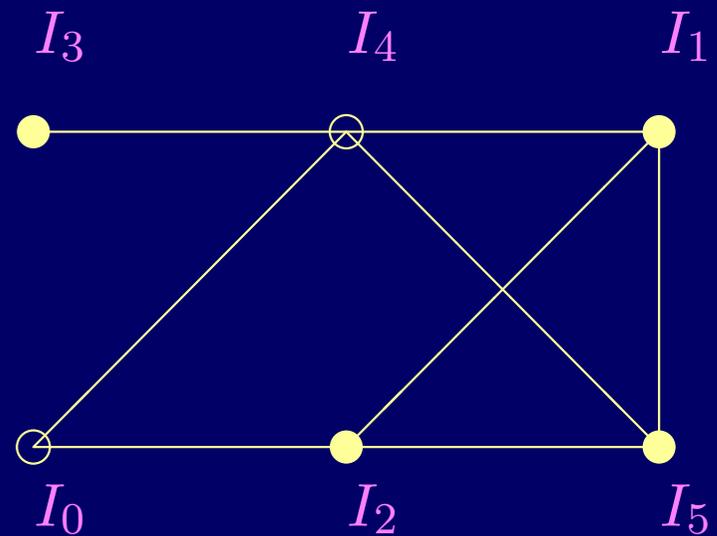
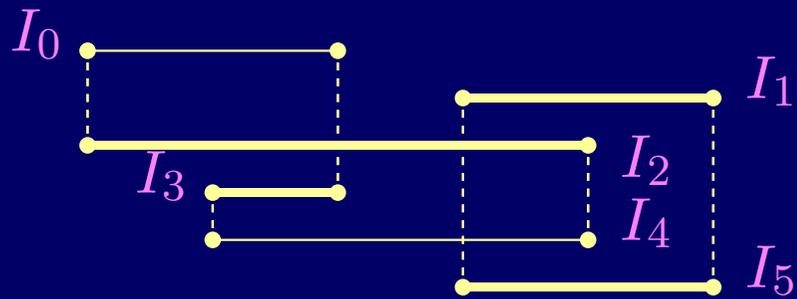
Considérons l'inversion de l'intervalle élémentaire I .

1. Si I est non orienté, les effets sur le graphe de chevauchement sont de changer la couleur de tous les sommets dans $G_I - \{I\}$, et de complémententer les arêtes de $G_I - \{I\}$.
2. Si I est orienté, les effets sur le graphe de chevauchement sont de changer la couleur de tous les sommets de G_I , et de complémententer les arêtes de G_I .

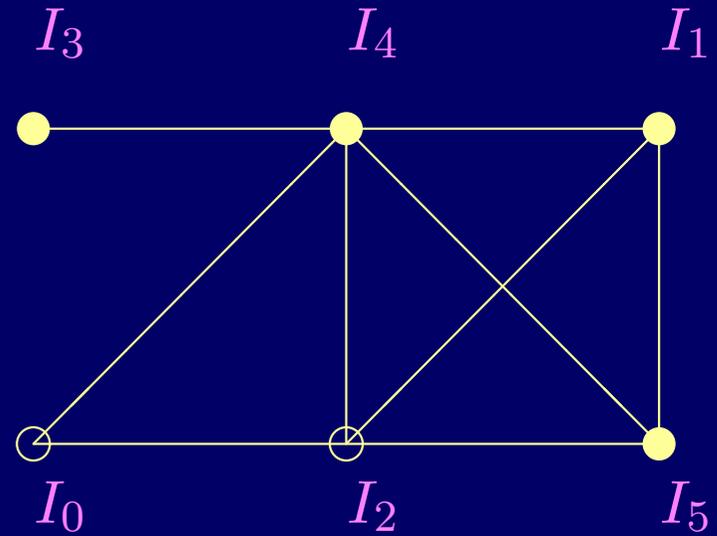
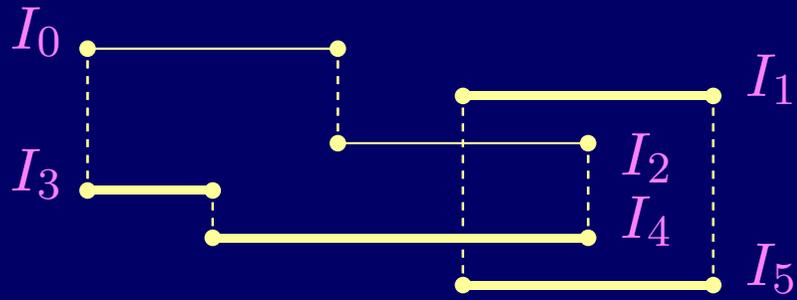
(0 4 -3 1 -5 -2 6)



(0 3 -4 1 -5 -2 6)



(0 4 -3 1 -5 -2 6)



Les points, les intervalles élémentaires et les cycles sont organisés en structure d'ordre supérieur appelés **composants**.

Un **composant** de P est un intervalle de i à $(i + j)$ ou de $-(i + j)$ à $-i$, pour $j > 0$, dont l'ensemble des éléments non-signés est $\{i, \dots, i + j\}$, et tel qu'il ne soit pas l'union de tels intervalles.

Les points, les intervalles élémentaires et les cycles sont organisés en structure d'ordre supérieur appelés **composants**.

Un **composant** de P est un intervalle de i à $(i+j)$ ou de $-(i+j)$ à $-i$, pour $j > 0$, dont l'ensemble des éléments non-signés est $\{i, \dots, i+j\}$, et tel qu'il ne soit pas l'union de tels intervalles.

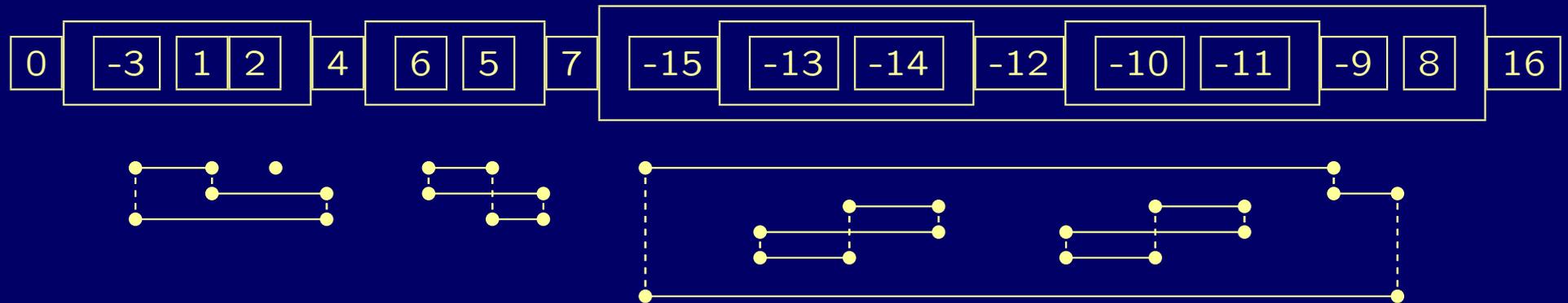
Exemple

$$\pi_2 = (0 \quad -3 \quad 1 \quad 2 \quad 4 \quad 6 \quad 5 \quad 7 \quad -15 \quad -13 \quad -14 \quad -12 \quad -10 \quad -11 \quad -9 \quad 8 \quad 16).$$

π_2 a six composants : $(0 \dots 4)$, $(4 \dots 7)$, $(7 \dots 16)$, $(1 \dots 2)$, $(-15 \dots -12)$ et $(-12 \dots -9)$.

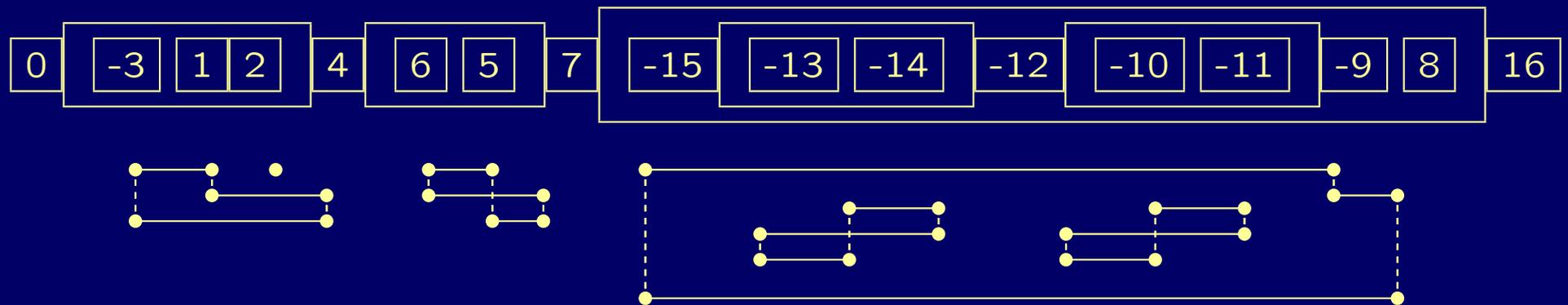
Les composants d'une permutation peuvent être représentés par un **diagramme de boîtes**

$$\pi_2 = (0 \quad -3 \quad 1 \quad 2 \quad 4 \quad 6 \quad 5 \quad 7 \quad -15 \quad -13 \quad -14 \quad -12 \quad -10 \quad -11 \quad -9 \quad 8 \quad 16).$$



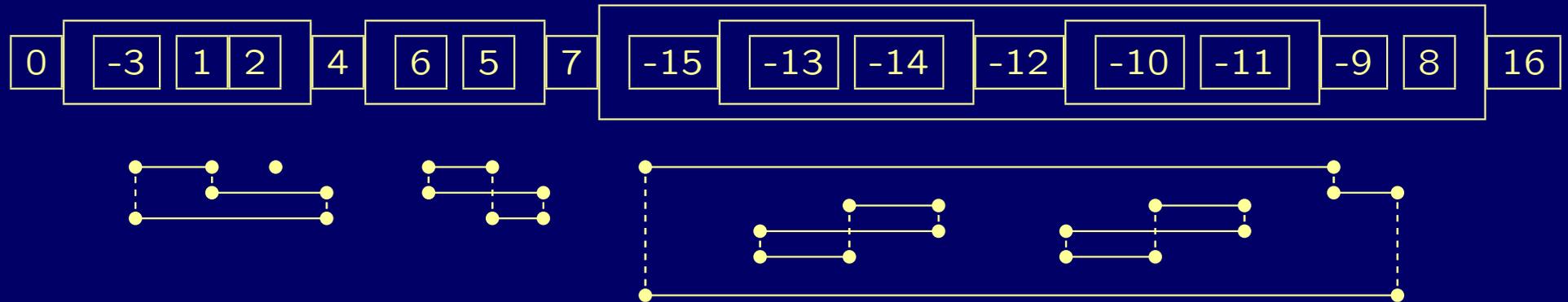
Les composants d'une permutation peuvent être représentés par un **diagramme de boîtes**

$$\pi_2 = (0 \quad -3 \quad 1 \quad 2 \quad 4 \quad 6 \quad 5 \quad 7 \quad -15 \quad -13 \quad -14 \quad -12 \quad -10 \quad -11 \quad -9 \quad 8 \quad 16).$$



- Un point $p \cdot q$ appartient au plus petit composant qui contient à la fois p et q .
 - Les extrémités d'un intervalle élémentaire appartiennent au même composant,
- ⇒ donc tous les points d'un cycle appartiennent au même composant.

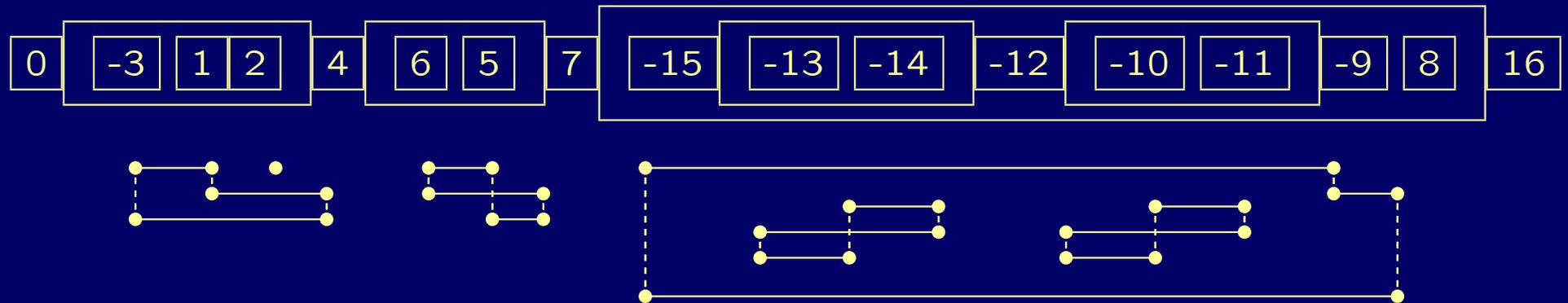
$$\pi_2 = (0 \quad -3 \quad 1 \quad 2 \quad 4 \quad 6 \quad 5 \quad 7 \quad -15 \quad -13 \quad -14 \quad -12 \quad -10 \quad -11 \quad -9 \quad 8 \quad 16).$$



Le **signe** d'un point $p \cdot q$ est positif si à la fois p et q sont positifs, il est négatif si à la fois p et q sont négatifs.

Un composant est **non orienté** si il a un ou plusieurs breakpoints, et que tous ont le même signe, sinon le composant est **orienté**.

$$\pi_2 = (0 \quad -3 \quad 1 \quad 2 \quad 4 \quad 6 \quad 5 \quad 7 \quad -15 \quad -13 \quad -14 \quad -12 \quad -10 \quad -11 \quad -9 \quad 8 \quad 16).$$



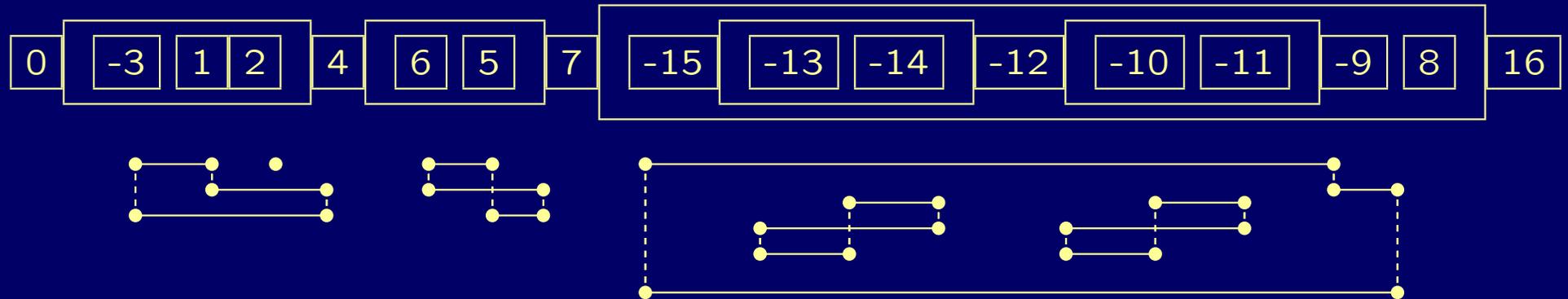
Le **signe** d'un point $p \cdot q$ est positif si à la fois p et q sont positifs, il est négatif si à la fois p et q sont négatifs.

Un composant est **non orienté** si il a un ou plusieurs breakpoints, et que tous ont le même signe, sinon le composant est **orienté**.

Exemple :

- $(4 \dots 7)$, $(-15 \dots -12)$ et $(-12 \dots -9)$ sont non orientés
- $(0 \dots 4)$, $(7 \dots 16)$ sont orientés

$$\pi_2 = (0 \quad -3 \quad 1 \quad 2 \quad 4 \quad 6 \quad 5 \quad 7 \quad -15 \quad -13 \quad -14 \quad -12 \quad -10 \quad -11 \quad -9 \quad 8 \quad 16).$$



Le **signe** d'un point $p \cdot q$ est positif si à la fois p et q sont positifs, il est négatif si à la fois p et q sont négatifs.

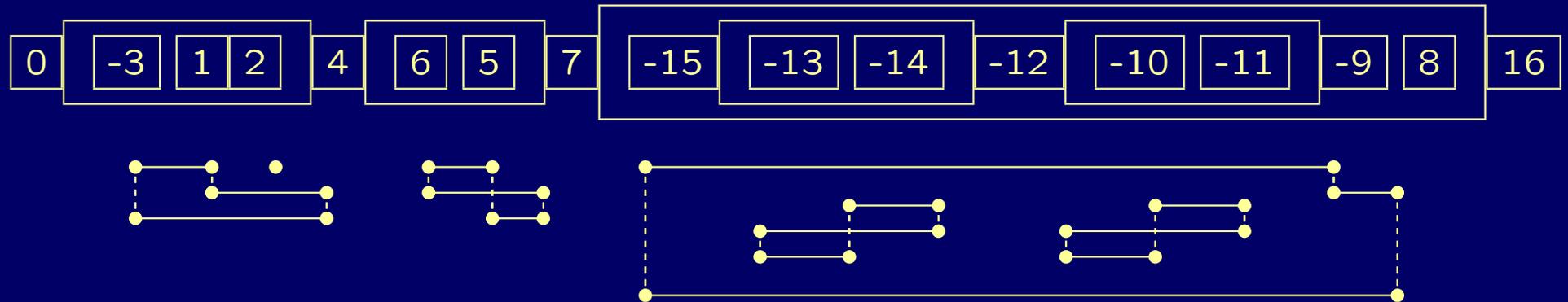
Un composant est **non orienté** si il a un ou plusieurs breakpoints, et que tous ont le même signe, sinon le composant est **orienté**.

Exemple :

- $(4 \dots 7)$, $(-15 \dots -12)$ et $(-12 \dots -9)$ sont non orientés
- $(0 \dots 4)$, $(7 \dots 16)$ sont orientés

Impossible de créer une adjacence dans un composant non orienté

$$\pi_2 = (0 \ -3 \ 1 \ 2 \ 4 \ 6 \ 5 \ 7 \ -15 \ -13 \ -14 \ -12 \ -10 \ -11 \ -9 \ 8 \ 16).$$

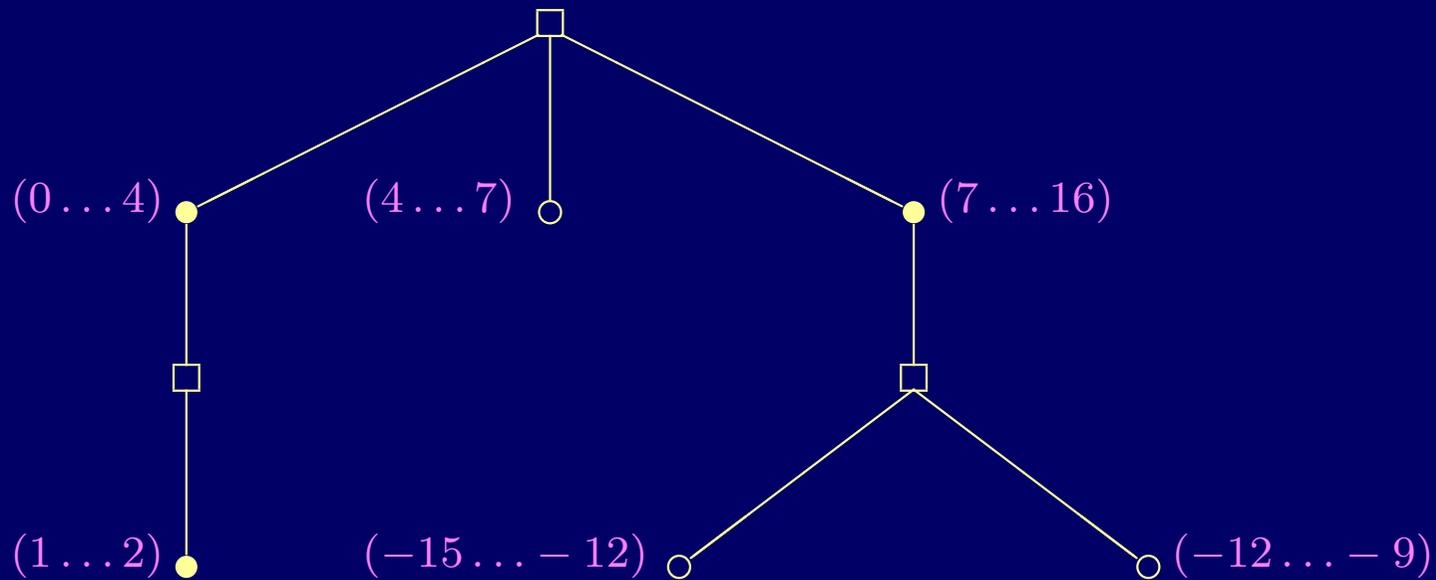
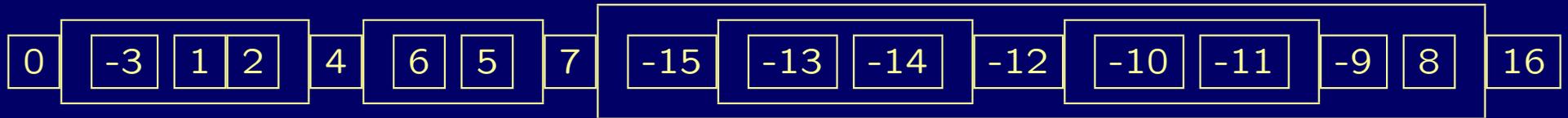


Deux composants distincts d'une permutation sont soit **disjoints**, soit **emboîtés** avec des extrémités différentes, soit ils **se chevauchent d'un élément**.

- Quand deux composants se chevauchent d'un élément, on dit qu'ils sont **liés**.
- Une suite de composants liés forme une **chaîne**.
- Une chaîne qui ne peut pas être étendue par la gauche ou par la droite est dite **maximale**.

Soit une permutation π sur l'ensemble $\{0, 1, \dots, n\}$ et ses composants, l'arbre T_π est défini par la construction suivante :

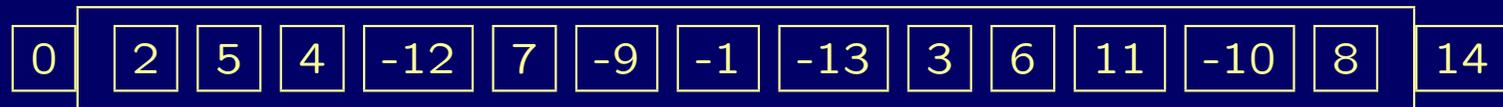
1. Chaque composant est représenté par un nœud rond.
2. Chaque chaîne maximale est représentée par un nœud carré dont les fils (ordonnés) sont les nœuds ronds qui représentent les composants de la chaîne.
3. Un nœud carré est le fils du plus petit composant qui contient cette chaîne.



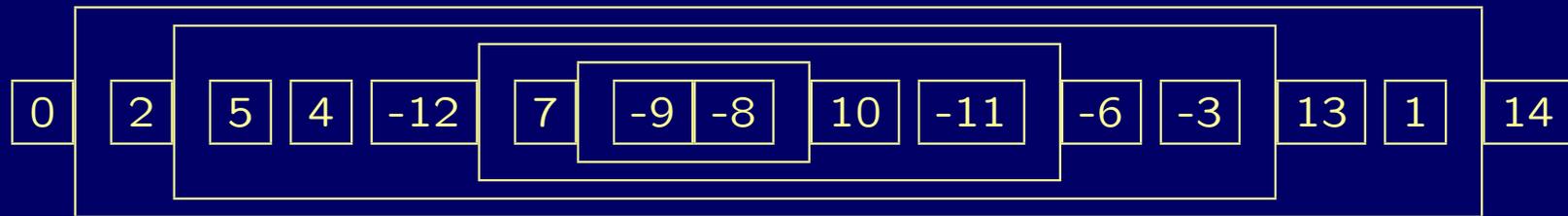
L'arbre T_π associé à la permutation π .

- Une inversion peut créer ou détruire un nombre quelconque de composants.

Exemple Inversion de $(-1 \dots 8)$ dans la permutation suivante :



mène à une permutation avec 4 nouveaux composants :



- Créer des composants orientés, ou des adjacences, est généralement considéré comme un pas vers le tri optimal d'une permutation.
- Cependant, la création de composants non orientés devrait être évité. Heureusement peu d'inversions ont cet effet.

- **Hurdles** : un certain type de composants non orientés
- Une hurdle peut être **simple** si son élimination fait diminuer le nombre total de hurdles ; c'est une **super** hurdle sinon
- **Forteresse** : une permutation qui a un nombre impair de hurdles et qui sont toutes des super hurdles

- Introduction
- Anatomie d'une permutation signée
- Les effets d'une inversion
- **Les algorithmes**
 - Les différents algorithmes
 - Quelques mots sur l'algorithme de Bergeron
- Conclusion et perspectives

1. Le problème SBR

- Berman et Hannenhalli 1996, $O(n^2\alpha(n))$
- Kaplan, Shamir et Tarjan 1999, $O(n^2)$
Happy cliques
- Bergeron 2001, $O(n^2)$ ou $O(n^3)$
- Tannier et Sagot 2004, $O(n\sqrt{n\log n})$

2. Le problème de la distance d'inversion

- Bader, Moret et Yan 2001, $O(n)$

1. Élimination des composants non orientés

→ « Hurdles cutting »

→ « Hurdles merging »

2. Trier les composants orientés

→ Trouver les « safe inversions » ($d(\pi \cdot \rho) = d(\pi) - 1$)

3. Calcul de la distance d'inversion

HurdleCutting

Si un composant C est non orienté, l'inversion d'un intervalle élémentaire dont les extrémités appartiennent à C oriente C , et laisse le nombre de cycles de la permutation inchangés.

HurdleMerging

Une inversion qui a ses deux extrémités dans des composants différents A et B détruit, ou oriente, tous les composants sur le chemin de A à B dans T_π , sans créer de nouveaux composants non orientés.

-
- Question Comment choisir des inversions sûres ? (càd celles qui ne crée par de nouveaux composants non orientés)

- Question Comment choisir des inversions sûres ? (càd celles qui ne crée par de nouveaux composants non orientés)

Le **score** d'une inversion est défini comme le nombre d'intervalles orientés dans la permutation résultante.

- **Question** Comment choisir des inversions sûres ? (càd celles qui ne crée par de nouveaux composants non orientés)

Le **score** d'une inversion est défini comme le nombre d'intervalles orientés dans la permutation résultante.

- **Théorème** [Bergeron, 2001] L'inversion d'un intervalle orienté de score maximum ne crée pas de nouveaux composants non orientés.

- **Question** Comment choisir des inversions sûres ? (càd celles qui ne crée par de nouveaux composants non orientés)

Le **score** d'une inversion est défini comme le nombre d'intervalles orientés dans la permutation résultante.

- **Théorème** [Bergeron, 2001] L'inversion d'un intervalle orienté de score maximum ne crée pas de nouveaux composants non orientés.
- **Corollaire** Si une permutation π n'est constituée que composants orientés,

$$d(\pi) = n - c(\pi)$$

- On travaille sur l'arbre T_π

- On travaille sur l'arbre T_π

Une **couverture** \mathcal{C} de T_π est un ensemble de chemins joignant tous les composants non orientés de π , et tel que chaque nœud extrémité d'un chemin appartient à un seul chemin.

- On travaille sur l'arbre T_π

Une **couverture** \mathcal{C} de T_π est un ensemble de chemins joignant tous les composants non orientés de π , et tel que chaque nœud extrémité d'un chemin appartient à un seul chemin.

- « HurdleCutting » et « HurdleMerging » \Rightarrow chaque couverture de T_π décrit une séquence d'inversions qui oriente de P

- On travaille sur l'arbre T_π

Une **couverture** \mathcal{C} de T_π est un ensemble de chemins joignant tous les composants non orientés de π , et tel que chaque nœud extrémité d'un chemin appartient à un seul chemin.

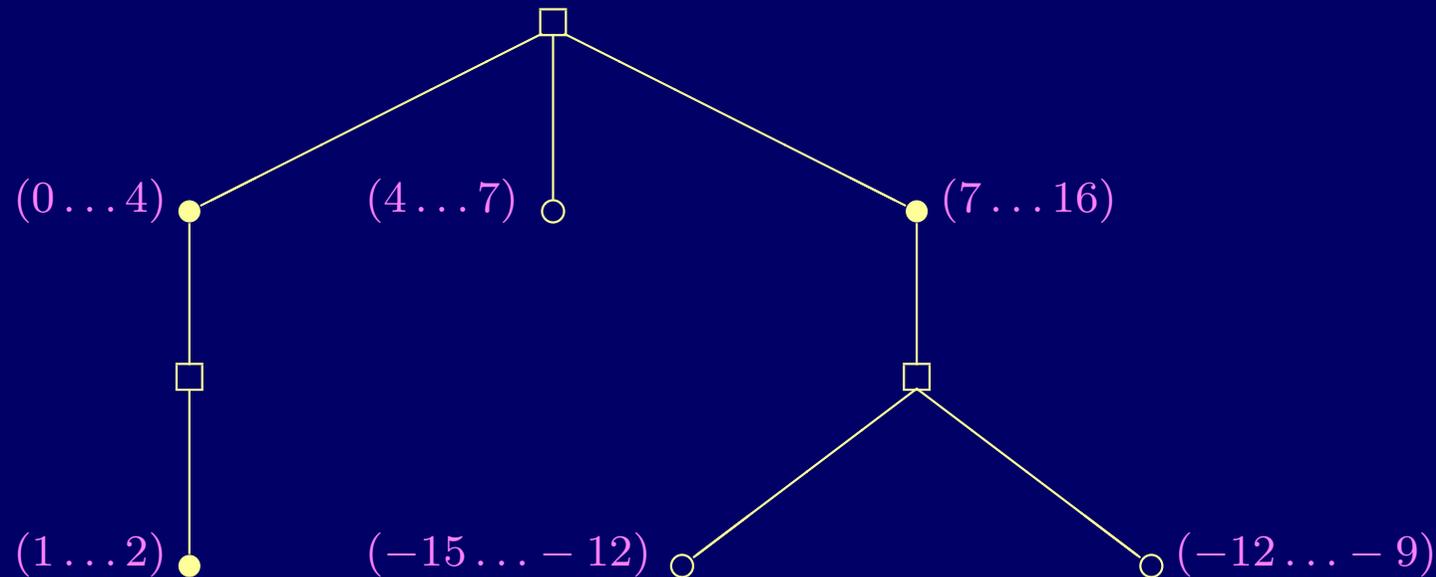
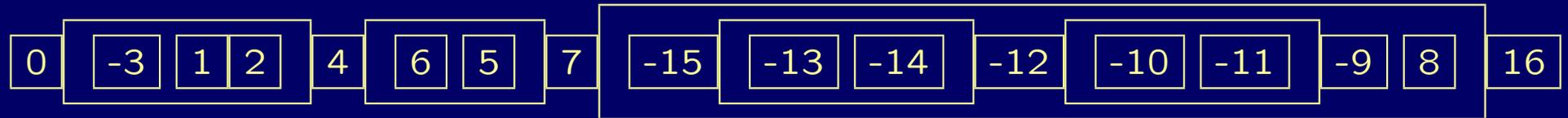
- « HurdleCutting » et « HurdleMerging » \Rightarrow chaque couverture de T_π décrit une séquence d'inversions qui oriente de P
- Un chemin est **long** s'il contient au moins 2 composants non orientés, et son **coût** est 2
- Un chemin est **court** sinon son **coût** est 1
- Le **coût** d'une couverture est la somme du coût de ses chemins ; une couverture est **optimale** si son coût est minimal (soit t ce coût)

- On travaille sur l'arbre T_π

Une **couverture** \mathcal{C} de T_π est un ensemble de chemins joignant tous les composants non orientés de π , et tel que chaque nœud extrémité d'un chemin appartient à un seul chemin.

- « HurdleCutting » et « HurdleMerging » \Rightarrow chaque couverture de T_π décrit une séquence d'inversions qui oriente de P
- Un chemin est **long** s'il contient au moins 2 composants non orientés, et son **coût** est 2
- Un chemin est **court** sinon son **coût** est 1
- Le **coût** d'une couverture est la somme du coût de ses chemins ; une couverture est **optimale** si son coût est minimal (soit t ce coût)

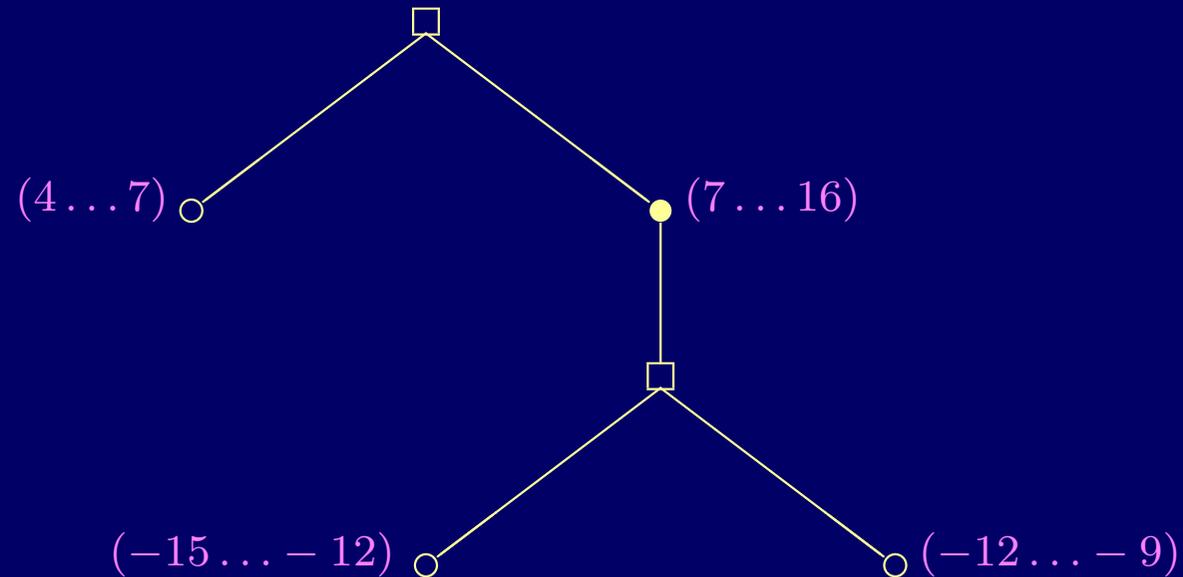
$$d(\pi) = n - c(\pi) + t$$



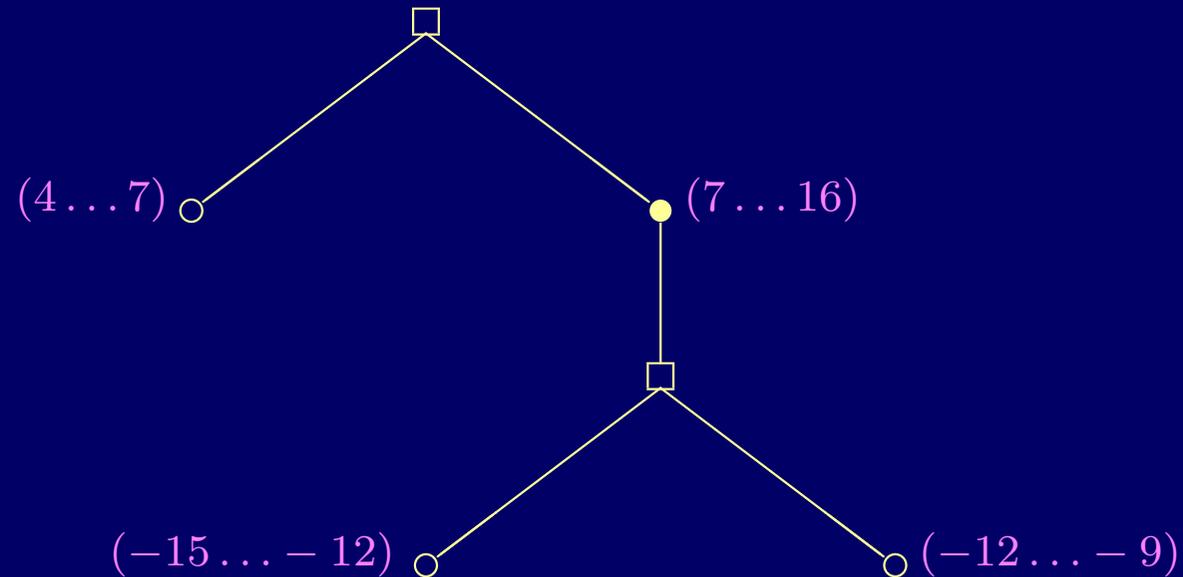
Le long chemin joignant les composants $(4 \dots 7)$ et $(-12 \dots -9)$ détruirait ces composants, ainsi que $(7 \dots 16)$.

-
- Soit T' le plus petit sous-arbre non enraciné de T_π qui contient tous les composants non orientés
 - Toutes les feuilles de T' sont donc des composants non orientés, mais les nœuds ronds internes peuvent être des composants orientés

- Soit T' le plus petit sous-arbre non enraciné de T_π qui contient tous les composants non orientés
- Toutes les feuilles de T' sont donc des composants non orientés, mais les nœuds ronds internes peuvent être des composants orientés

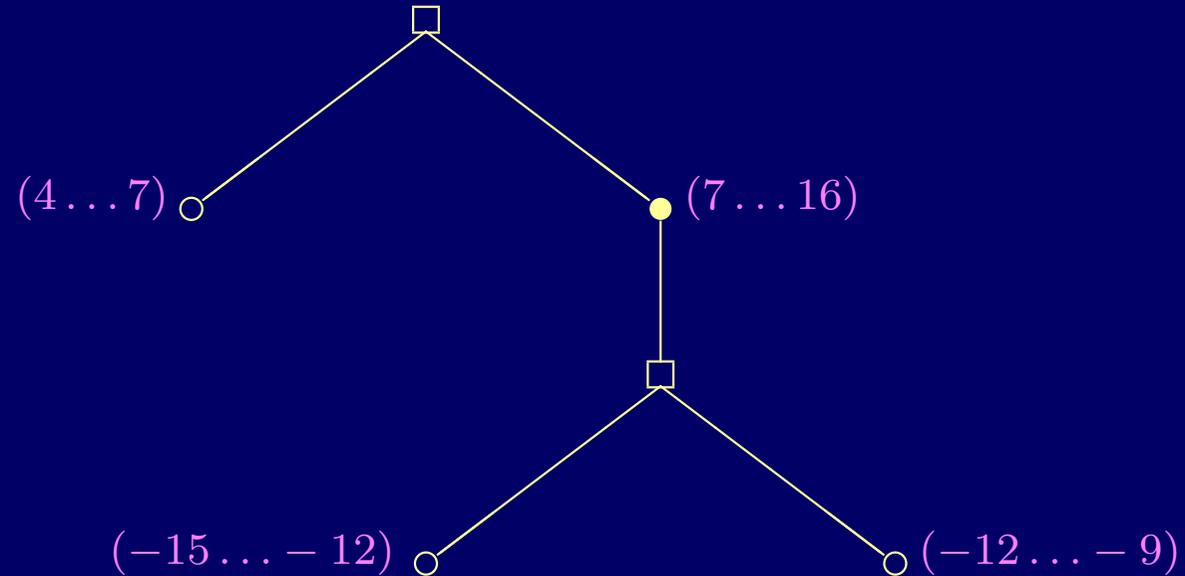


- Soit T' le plus petit sous-arbre non enraciné de T_π qui contient tous les composants non orientés
- Toutes les feuilles de T' sont donc des composants non orientés, mais les nœuds ronds internes peuvent être des composants orientés

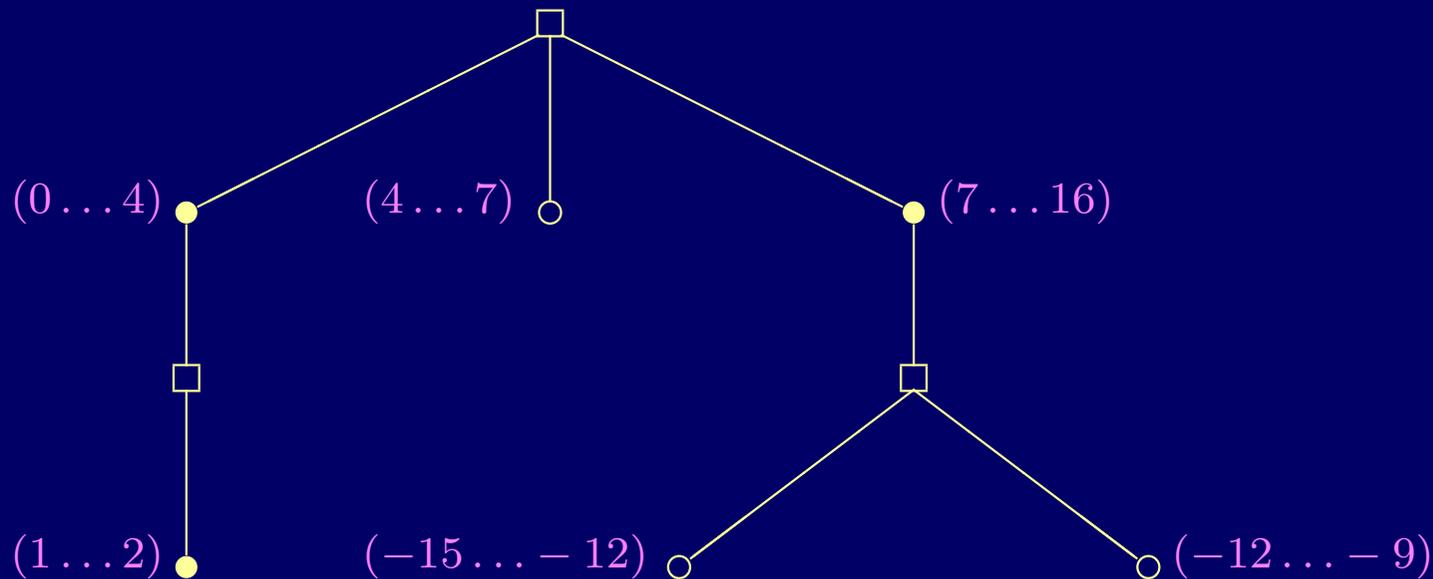
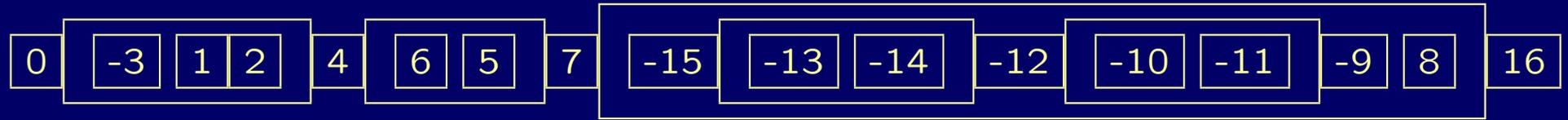


- Une **branche** est l'ensemble des nœuds sur le chemin montant d'une feuille jusqu'à un nœud de degrés ≥ 3 , mais excluant celui-ci
Une branche est **courte** si elle contient 1 seul composant non orienté ; elle est **longue** sinon

1. Si T' a $2k$ feuilles, alors $t = 2k$.
2. Si T' a $2k + 1$ feuilles, et au moins une d'entre elles est sur une branche courte, alors $t = 2k + 1$.
3. Si T' a $2k + 1$ feuilles, et aucune d'entre elles est sur une branche courte, alors $t = 2k + 2$.

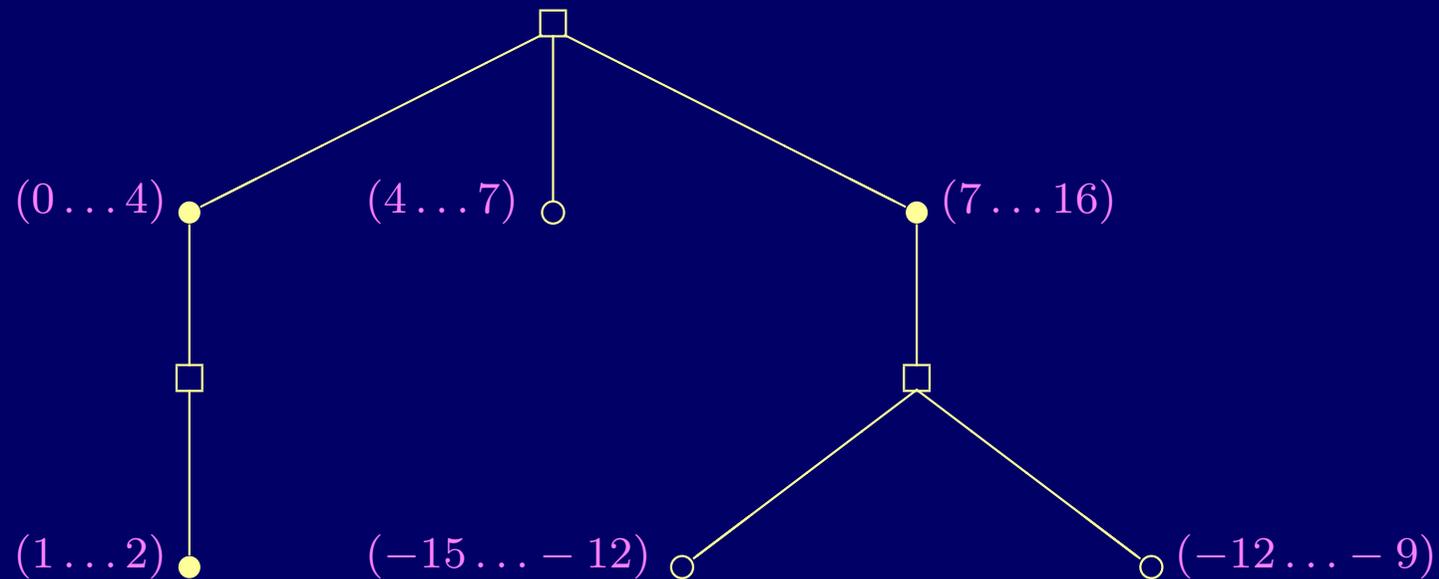
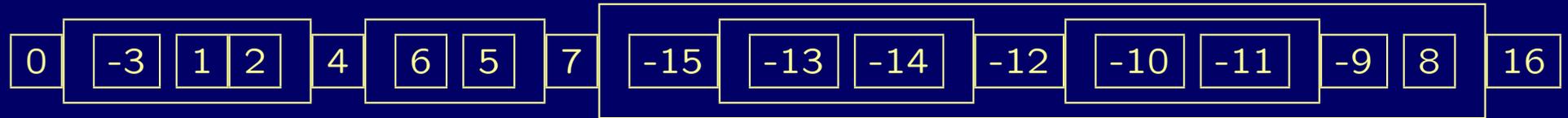


3 feuilles, chacune sur des branches courtes $\Rightarrow t = 3$



Pour cette permutation π :

$$n = 16, c(\pi) = 6, t = 3, \text{ donc } d(\pi) =$$



Pour cette permutation π :

$$n = 16, c(\pi) = 6, t = 3, \text{ donc } d(\pi) = 13$$

$$\pi_3 = (0 \ 2 \ 4 \ 3 \ 5 \ 1 \ 6 \ 8 \ 10 \ 9 \ 11 \ 7 \ 12 \ 14 \ 16 \ 15 \ 17 \ 13 \ 18)$$

- Introduction
- Anatomie d'une permutation signée
- Les effets d'une inversion
- Les algorithmes
- Perspectives

- Autre simplification de la théorie ?
- SBR en temps linéaire ?
- Tri avec d'autres opérations de réarrangements
 - Distance génomique (inversions, translocations, fusions, fissions)
 - Avec transpositions ? (difficile)
- Un nombre exponentiel de scénarios parcimonieux
 - ⇒ discriminer parmi ces scénarios avec d'autres critères

2004 Figeac et Varré

"Scénarios parfaits"

2004 Bérard, Bergeron et Chauve

"Scénarios optimaux et commutants"

2005 Sagot et Tannier

"Scénarios optimaux et parfaits"

2005 Bérard, Bergeron, Chauve et Paul

"Scénarios parfaits (décomposition modulaire de graphes)"