Un siècle de réarrangements génomiques

Anne Bergeron, Krister M. Swenson

Chapitre **1**

Un siècle de réarrangements génomiques

Anne BERGERON, Krister SWENSON

LACIM, LIRRM

1.1. Introduction

En février 1920, le prestigieux journal PNAS publiait une courte lettre intitulée "*The Evidence for the Linear Order of the Genes*'" (Morgan *et al*. 1920), alimentant le débat entre deux théories opposées sur l'organisation des marqueurs génétiques sur un chromosome. D'une part, Morgan et ses collègues soutenaient que, dans une espèce, les gènes étaient disposés les uns à la suite des autres, dans une structure linéaire. Leur rival scientifique, William Castle, proposait plutôt une organisation tri-dimensionnelle qui, selon lui, expliquait mieux les *distances* observées entre les différents marqueurs (Castle 1919).

Les découvertes subséquentes du 20ème siècle ont finalement donné raison à l'équipe de Morgan. On leur doit les premières cartes génétiques, l'identification d'inversions dans l'ordre des marqueurs entre différentes espèces, et la construction d'arbres qui retracent l'histoire de ces événements.

Il faudra encore quelques décennies, avec les travaux de Crick et Watson (Crick *et al*. 1954), pour établir que les molécules d'ADN forment le support physique de cette organisation linéaire. Au tournant du 21ème siècle, les avancées des techniques de séquençage ont permis la comparaison des génomes d'espèces de plus en plus nombreuses, mettant en lumière une grande diversité de types de réarrangements :

de biologique, le problème d'expliquer la nature de ces réarrangements est devenu algorithmique.

Parallèlement, l'hypothèse de Castle sera partiellement justifiée par la mise en évidence de la structure tri-dimensionnelle adoptée par les chromosomes à l'intérieur du noyau des cellules (Lieberman-Aiden *et al*. 2009). Oui, les réarrangements se produisent entre des molécules linéaires, mais les positions affectées par ces réarrangements sont souvent déterminées par leur emplacement relatif dans l'espace (Nikiforova *et al*. 2000, Yaffe *et al*. 2010, Véron *et al*. 2011, Swenson and Blanchette 2019).

Ce chapitre développe les techniques mathématiques et algorithmiques qui permettent de traiter les problèmes de réarrangements de *gènes*, autant dans leur sens primitif de marqueurs de traits hérités, que dans leur interprétation moderne comme familles de transcrits.

1.2. L'ordre linéaire des gènes et les opérations de réarrangement

Pour étudier les réarrangements, nous avons besoin de définitions abstraites et simplifiées de gènes et de chromosomes. Abstraites, au sens où chaque gène correspondra à un segment de chromosome, identifié par un nombre et une orientation relative. Simplifiées, car les découvertes des dernières décennies ont considérablement revu et corrigé l'idée initiale d'un gène comme marqueur discret d'un trait héréditaire [voir, par exemple, Pearson (2006)].

Lorsque l'on compare l'ordre des gènes de deux espèces, on fait face à plusieurs problèmes dont le principal est de déterminer si deux gènes sont *égaux*. Cette question relève de la biologie : le concept de base est celui de gènes *homologues*, définis comme étant des gènes qui partagent un ancêtre commun. Quand une cellule se divise, elle transmet une copie de son ADN à chacune de ses cellules-filles, et ainsi de suite pour chacune des nouvelles cellules : théoriquement toutes les descendantes de la cellule ancestrale ont des gènes identiques.

En pratique, plusieurs événements peuvent altérer l'ADN de certaines filles, comme des erreurs dans le mécanisme de copie, ou encore l'exposition à des facteurs environnementaux. Si l'ancêtre commun est relativement récent, disons quelques millions d'années, on peut reconnaitre assez facilement les gènes homologues car ils sont encore presque identiques, et de nombreux outils bio-informatiques ont automatisé ce processus. Nous supposerons donc, dans ce chapitre que le problème de décider si deux gènes sont égaux est résolu.

Lorsque l'on compare le contenu en gènes de deux organismes, plusieurs problèmes apparaissent : certains gènes peuvent être répétés dans un même

organisme, on parle alors de *duplications* ; certains gènes peuvent être présents dans un organisme et absents dans l'autre, on parle alors d' *insertions* et de *délétions* ; enfin, même en l'absence de duplications, d'insertions ou de délétions, la composition des chromosomes, ou l'ordre dans lequel apparaissent les gènes dans un chromosome, peut varier, on parle alors de *réarrangements*.

Ce chapitre se concentre sur le problème de réarrangements entre différentes espèces, et nous supposerons que les génomes comparés ont le même contenu en gènes, sans duplications.

1.2.1. *Représentations et définitions de base*

Les techniques modernes de séquençage et d'annotation de génomes permettent d'identifier, pour chaque gène, sa position sur le chromosome ainsi que son orientation – positive ou négative. Ces informations sont souvent rendues par des diagrammes tel qu'illustré dans la Figure 1.1 où les gènes sont représentés par des flèches orientées, et étiquetées par un identifiant qui permet de reconnaitre les gènes homologues.

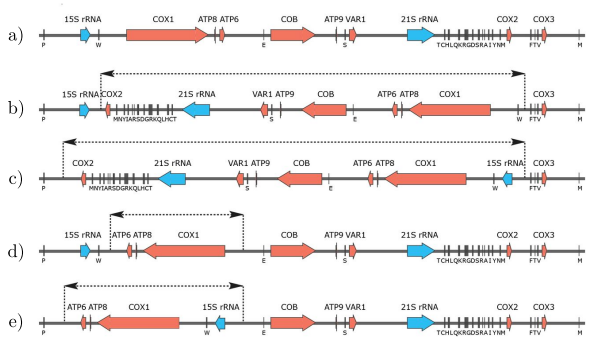

Figure 1.1. *Représentation typique de segments de chromosomes montrant des réarrangements, ici dans les mitochondries de différentes espèces de levures. Les gènes COX1 et 21sRNA, par exemple, apparaissent dans des orientations relatives différentes quand on compare les espèces a et d. Tiré de De Chiara* et al. *(2020).*

Lorsque l'on étudie les problèmes théoriques de réarrangements, la représentation des génomes est encore plus simplifiée : les identifiants des gènes sont des nombres entiers, avec un signe positif ou négatif donnant leurs orientations *relatives* par rapport à l'un des génome, arbitrairement choisi, dont les gènes sont identifiés dans l'ordre croissant. La Figure 1.2 montre un exemple de deux génomes ayant chacun deux chromosomes, avec neuf gènes.

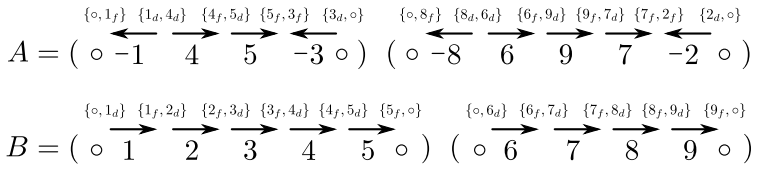

Figure 1.2. *Représentations linéaires des génomes $A =$ (∘ -1 4 5 -3 ∘) (∘ -8 6 9 7 -2 ∘) et $B =$ (∘ 1 2 3 4 5 ∘) (∘ 6 7 8 9 ∘), avec leurs ensembles d'adjacences. On remarque que seule l'adjacence conservée $\{4_f, 5_d\}$ est commune aux deux ensembles.*

Dans cet exemple, le génome B est représenté par

$$B = \text{ (∘ 1 2 3 4 5 ∘) (∘ 6 7 8 9 ∘)}$$

où le symbole ∘ indique l'extrémité d'un chromosome, et le génome A par

$$A = \text{(∘ -1 4 5 -3 ∘) (∘ -8 6 9 7 -2 ∘).}$$

Cette notation indique, par exemple, que les gènes 3 et 4 se retrouvent dans des positions différentes et des orientations relatives opposées dans les génomes A et B.

Les gènes étant orientés, on en distingue les deux *extrémités* : le *début* et la *fin*. Les extrémités d'un gène g seront représentées par les symboles g_d et g_f respectivement. Une *adjacence* d'un génome est définie comme une paire $\{u, v\}$ d'extrémités (notons que les adjacences $\{u, v\}$ et $\{v, u\}$ sont les mêmes). Lorsque qu'une extrémité e d'un gène correspond aussi à l'extrémité d'un chromosome, appelée télomère, on note l'adjacence $\{e, \circ\}$.

Par exemple, chaque génome de la Figure 1.2 comporte 11 adjacences, dont 7 sont *internes*, et 4 situées au bout des chromosomes. Ces adjacences sont toutes différentes d'un génome à l'autre sauf une, $\{4_f, 5_d\}$, qui correspond au fait que, dans les deux génomes, les gènes 4 et 5 se suivent, dans la même orientation. Dans ce dernier cas, on parle alors d'adjacence *conservée*.

La notion d'adjacences conservées permet de formaliser une dernière notion fondamentale, celle de *génomes équivalents*. En effet, dans la réalité, les chromosomes ne sont pas ordonnés et n'ont pas une orientation fixée. La définition suivante permettra de décider si deux génomes sont identiques.

DEFINITION 1.1.– *Deux génomes sans duplications et ayant le même contenu en gènes sont* équivalents *si et seulement si leurs ensembles d'adjacences sont égaux.*

Par exemple, les génomes $A = (\circ\ 1\ 2\ \circ)\ (\circ\ 3\ 4\ \circ)$ et $B = (\circ$ -4 -3 $\circ)\ (\circ\ 1\ 2\ \circ)$ sont équivalents puisque leurs adjacences sont, dans chaque cas, $\{\{\circ, 1_d\}, \{1_f, 2_d\}, \{2_f, \circ\}, \{\circ, 3_d\}, \{3_f, 4_d\}, \{4_f, \circ\}\}$.

1.2.2. *Les opérations DCJ et le graphe des cassures*

La plupart des réarrangements qui préservent le contenu en gènes d'un génome peuvent être modélisés par une opération simple, introduite dans Yancopoulos *et al.* (2005) qui agit sur un ou deux brins d'ADN, et qui est basée sur des mécanismes biologiques naturels qui permettent de réparer la rupture accidentelle d'un chromosome, suite à un accident chimique, radiologique ou mécanique.

Lors d'une rupture simple, ces mécanismes font, la plupart du temps, leur travail correctement, comme dans le cas d'une rupture d'un tuyau de plomberie. Il s'agit de repérer le lieu de la rupture, et de couper la section défectueuse, et de joindre les deux bouts.

Si deux brins d'ADN rapprochés sont victimes d'un même accident, la stratégie précédente peut donner lieu à des aberrations, comme le raccordement de la sortie d'eau chaude avec l'entrée d'eau froide. Avec le formalisme introduit dans la section précédente, si deux adjacences $\{u, v\}$ et $\{r, s\}$ sont coupées, il existe trois façons de les raccorder :

1) $\{u, s\}, \{r, v\}$,
2) $\{u, r\}, \{v, s\}$,
3) $\{u, v\}, \{r, s\}$.

Seuls les deux premiers raccordements modifient la structure d'un génome : nous désignons l'une ou l'autre de ces opérations par l'acronyme *DCJ*, de l'anglais *Double-Cut-and-Join*.

Par exemple, si l'on effectue une opération DCJ sur les adjacences $\{1_f, 2_d\}$ et $\{3_f, 4_d\}$, du génome $A = (\circ\ 1\ 2\ \circ)\ (\circ\ 3\ 4\ \circ)$, on obtient soit :

$$A' = (\circ\ 1\ 4\ \circ)\ (\circ\ 3\ 2\ \circ),$$

ou encore :

$$A'' = (\circ\ 1\ \text{-}3\ \circ)\ (\circ\ \text{-}4\ 2\ \circ).$$

évidemment, il est possible que les deux adjacences coupées appartiennent à un même chromosome. Par exemple, une opération DCJ sur les adjacences $\{1_f, 2_d\}$ et $\{3_f, 4_d\}$, du génome $A = (\circ\ 1\ 2\ 3\ 4\ \circ)$ peut produire le génome :

$$A' = (\circ\ 1\ \text{-}3\ \text{-}2\ 4\ \circ).$$

Exercice 1 : Quel serait, dans ce cas, le génome obtenu avec l'opération DCJ alternative ? [Aide : Oui, il est possible qu'un chromosome n'ait aucun télomère, on parle alors de chromosome *circulaire*].

Étant donné deux génomes, notre objectif immédiat est de déterminer la nature et le nombre minimal d'opérations DCJ qui permettent de transformer l'un en l'autre. Pour y arriver, nous définissons une structure classique, le *graphe de cassures*, qui permet de comparer les ensembles d'adjacences des deux génomes.

DEFINITION 1.2 (Graphe de cassures).– *Soient A et B deux génomes avec n gènes. Un* graphe de cassures *est un graphe non-orienté dont les arêtes sont colorées en gris ou en noir. Les sommets du graphe sont les extrémités des n gènes, ainsi que le symbole $\circ$: les arêtes* noires *sont les adjacences du génome A, et les arêtes* grises *sont les adjacences du génome B.*

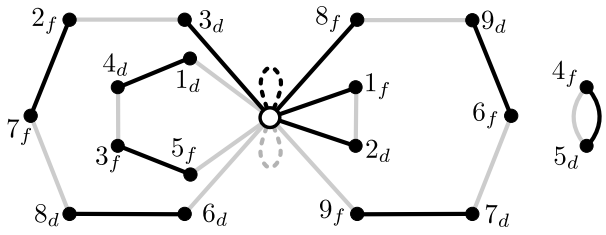

Figure 1.3. *Graphe de cassures des génomes $A = (\circ$ -1 4 5 -3 $\circ)$ $(\circ$ -8 6 9 7 -2 $\circ)$ et $B = (\circ$ 1 2 3 4 5 $\circ)$ $(\circ$ 6 7 8 9 $\circ)$.*

Le graphe de cassures des génomes $A = (\circ$ -1 4 5 -3 $\circ)$ $(\circ$ -8 6 9 7 -2 $\circ)$ et $B = (\circ$ 1 2 3 4 5 $\circ)$ $(\circ$ 6 7 8 9 $\circ)$ est illustré dans la Figure 1.3. [Note : Les deux arêtes en pointillé peuvent être ignorées en première lecture. Elles serviront éventuellement au dénombrement de scénarios de réarrangements.]

Le graphe de cassures permet de facilement calculer la *distance* entre deux génomes, notée d_{DCJ}, qui correspond au nombre minimal d'opérations DCJ permettant de transformer un génome en un autre.

Un cycle *élémentaire* d'un graphe est un cycle qui ne contient pas deux fois le même sommet. Dans un graphe de cassures, un cycle est *équilibré* s'il a le même nombre d'arêtes grises et noires. Le graphe de la Figure 1.3 a trois cycles élémentaires et équilibrés, dont deux contiennent le sommet $\circ$.

Un cycle élémentaire **et** équilibré est dit *complet* si tous ses sommets sont des extrémités de gènes, à l'inverse il est dit *incomplet* s'il contient le sommet $\circ$. La

distance DCJ est donnée par la formule

$$d_{DCJ} = n - (c + i/2)$$

où c est le nombre de cycles complets et i le nombre de cycles incomplets. Cette formule ne semble pas dépendre du nombre de cycles élémentaires non-équilibrés : il y en deux dans la Figure 1.3, $(5_f, 3_f, 4_d, 1_d, \circ)$ et $(1_f, 2_d, \circ)$. Ce n'est qu'une illusion, car il existe des dépendances entre le nombre de cycles non-équilibrés et les valeurs de n, c et i qui font que plusieurs formules différentes sont équivalentes (Bergeron and Stoye 2013).

Exercice 2 : Il est intéressant de tracer le graphe de cassures de génomes équivalents. Quel est le graphe de cassures des génomes $A =$ (∘ 1 2 ∘) (∘ 3 4 ∘) et $B =$ (∘ -4 -3 ∘) (∘ 1 2 ∘) ? En déduire que $d_{DCJ}(A, B) = 0$.

Dans la prochaine section, nous aurons besoin de calculer **toutes** les suites d'opérations DCJ de longueur minimale. Nous terminons donc cette section en montrant, avec un exemple simple, comment on peut calculer au moins **une** suite de longueur minimale. Une telle suite sera appelée *scénario optimal*.

Considérons les génomes $A =$ (∘ 1 -3 5 2 -4 6 ∘) et $B =$ (∘ 1 2 3 4 5 6 ∘). Le graphe de cassures correspondant est illustré dans la Figure 1.4. Il contient 2 cycles complets, 2 cycles incomplets, et aucun cycle non-équilibré, donc la distance entre les deux génomes est de $6 - (2 + 2/2) = 3$ opérations DCJ.

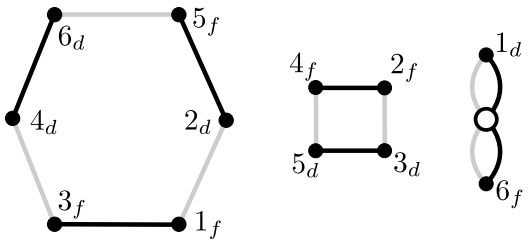

Figure 1.4. *Graphe de cassures des génomes $A =$ (∘ 1 -3 5 2 -4 6 ∘) et $B =$ (∘ 1 2 3 4 5 6 ∘).*

Pour réduire la distance $n - (c + i/2)$ entre deux génomes, il suffit d'augmenter le nombre de cycles complets ou incomplets. Notre objectif sera, pour cet exemple, d'augmenter le nombre de cycles complets.

Dans le cas du cycle de longueur 4, il est facile de le scinder en deux cycles de longueur 2 avec une simple opération DCJ, tel qu'illustré dans la Figure 1.5. Ce

réarrangement correspond à couper les adjacences noires $\{2_f, 4_f\}$ et $\{3_d, 5_d\}$ du génome A, pour construire les adjacences grises $\{2_f, 3_d\}$ et $\{3_d, 4_f\}$ du génome B, tel qu'illustré dans la Figure 1.5. Cette opération transforme le génome $A = (\circ\ 1\ {-3}\ 5\ 2\ {-4}\ 6\ \circ)$ en le génome $A' = (\circ\ 1\ {-3}\ {-2}\ {-5}\ {-4}\ 6\ \circ)$.

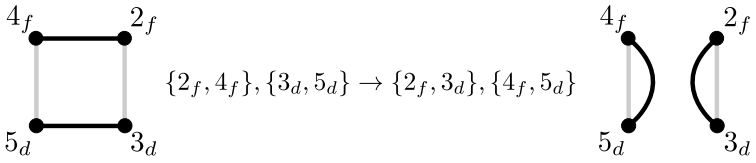

Figure 1.5. *Division d'un cycle avec une opération DCJ*

Pour les cycles plus longs, comme dans le cas du cycle de longueur 6, plusieurs stratégies sont possibles. En effet, trois opérations DCJ permettent ici de scinder le cycle en un cycle de longueur 2 et un cycle de longueur 4. Il s'agit de :

$$\{1_f, 3_f\}, \{2_d, 5_f\} \rightarrow \{1_f, 2_d\}, \{3_f, 5_f\},$$

$$\{1_f, 3_f\}, \{4_d, 6_d\} \rightarrow \{3_f, 4_d\}, \{1_f, 6_d\},$$

$$\{2_d, 5_f\}, \{4_d, 6_d\} \rightarrow \{2_d, 4_d\}, \{5_f, 6_d\}.$$

Pour compléter un scénario, chacune de ces opérations est suivie d'une opération DCJ qui scinde le carré résultant en deux. On a donc trois scenarios optimaux différents pour résoudre le cycle de longueur 6, ce qui nous donne un total de 9 scénarios différents puisque les opérations DCJ sur les deux cycles initiaux peuvent être effectués de manière indépendante.

1.3. Dénombrement de scénarios

Dans cette section, nous allons aborder le problème d'énumérer l'ensemble de tous les scénarios optimaux qui permettent de transformer un génome en un autre. Lorsque le graphe de cassures est composé de cycles de longueur 2, ou de cycles complets, le problème est relativement simple. C'était le cas, par exemple, des génomes de la Figure 1.4. Le principe de base sera toujours le même. Pour augmenter le nombre de cycles équilibrés, il faut diviser un cycle en deux : aucune opération DCJ qui fait intervenir deux cycles équilibrés ne peut augmenter le nombre de cycles. Il nous faudra donc énumérer tous les scénarios qui résolvent un cycle équilibré, puis énumérer toutes les décompositions en tels cycles du graphe de cassures.

1.3.1. *Les scénarios possibles pour un cycle équilibré de longueur* $2m$

Si on choisit deux arêtes noires $\{u, v\}$ et $\{r, s\}$ dans un cycle de longueur $2m$, une seule des deux operations DCJ possibles scindera le cycle en deux. Pour s'en convaincre, il suffit d'appliquer l'opération alternative au petit cycle de la Figure 1.5. Une opération DCJ dans un scénario optimal est donc caractérisée par une paire d'arêtes noires, et les quatre extrémités impliquées forment un quadrilatère si le cycle est dessiné sans croisements. L'effet de l'opération DCJ sera de diviser le cycle en deux *à travers* les arêtes noires.

Comme chaque opération DCJ scinde un cycle en deux, on a le résultat suivant :

PROPOSITION 1.1.– *Le nombre d'opérations DCJ pour résoudre un cycle de longueur* $2m$ *est* $m - 1$.

Si l'on trace les quadrilatères correspondant à toutes les opérations d'un scénario optimal pour un cycle, nous obtenons une *quadrangulation* du polygone à $2m$ sommets, c'est-à-dire une partition d'un polygone en rectangles non-chevauchants. Deux exemples de quadrangulation sont illustrés dans la Figure 1.6 pour un cycle de longueur 2*8.

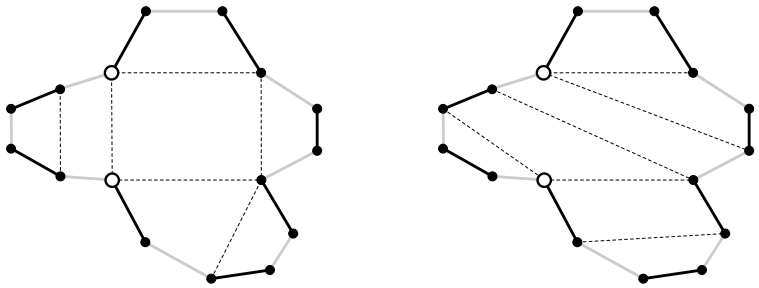

Figure 1.6. *Deux quadrangulations parmi les 7752 possibles d'un polygone à 16 sommets.*

À l'inverse, étant donné une quadrangulation, celle-ci nous donne tous les scénarios optimaux possibles dont les opérations DCJ partagent les mêmes paires d'arêtes.

Heureusement, on sait comment énumérer les quadrangulations d'un polygone à $2m$ sommets, et leur nombre est donné par la formule $\frac{1}{2m-1}\binom{3m-3}{m-1}$, (Baryshnikov 2001).

1.3.2. *Les (nombreuses) décompositions en cycles du graphe des cassures*

Maintenant que nous connaissons toutes les façons résoudre un cycle équilibré, notre prochain objectif est de calculer toutes les façons de décomposer le graphe de cassures en cycles équilibrés, tout en préservant la distance $d_{DCJ} = n - (c + i/2)$.

Les cycles élémentaires et équilibrés feront partie de toute décomposition en cycles équilibrés. Le problème se pose avec les cycles non-équilibrés comme, par exemple, les cycles $(5_f, 3_f, 4_d, 1_d, \circ)$ et $(1_f, 2_d, \circ)$ de la Figure 1.3.

Un cycle non-équilibré est caractérisé par la couleur, grise ou noire, des ses deux arêtes incidentes au sommet $\circ$: nous distinguerons donc l'ensemble G des cycles *gris* et l'ensemble N des cycles *noirs*, selon cette couleur.

Une première façon de construire des cycles équilibrés est de se servir des arêtes en pointillé pour équilibrer un cycle non-équilibré. Ces arêtes en pointillé du graphe de cassures permettent de modéliser le fait qu'une opération DCJ peut modifier le nombre de chromosomes, donc le nombre de télomères d'un génome, tout en conservant le même contenu en gènes. [Note : Un de ces réarrangements les plus célèbres concerne le chromosome 2 de l'humain qui résulterait de la fusion de deux chromosomes chez un ancêtre commun de l'humain et des néandertaliens (IJdo *et al*. 1991)].

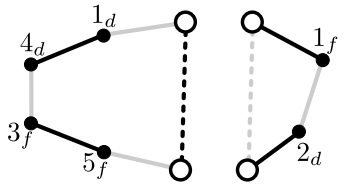

Figure 1.7. *Un cycle non-équilibré gris ou noir peut être fusionné avec un cycle pointillé de la couleur opposée, noir ou gris, pour créer un cycle équilibré.*

Chaque cycle non-équilibré est fusionné avec le *bon* cycle pointillé pour créer un cycle équilibré. La Figure 1.7 montre ces fusions pour les deux cycles $(5_f, 3_f, 4_d, 1_d, \circ)$ et $(1_f, 2_d, \circ)$.

La distance est préservée, puisque chaque opération DCJ qui scinde l'un de ces cycles en deux donne soit un cycle complet et un cycle non-équilibré, soit deux cycles incomplets. Par exemple, le cycle $(5_f, 3_f, 4_d, 1_d, \circ)$ sera scindé, selon les trois quadrangulations possibles du cycle augmenté, en :

1) $(3_f, 4_d,)(5_f, 1_d, \circ)$ ou

2) $(5_f, \circ)(3_f, 4_d, 1_d, \circ)$ ou
3) $(1_d, \circ)(5_f, 3_f, 4_d, \circ)$.

Un seconde façon de créer des cycles équilibrés est de fusionner un cycle gris $g \in G$ avec un cycle noir $n \in N$. Le résultat est un cycle équilibré qui passe deux fois par le sommet $\circ$. Dans l'exemple de Figure 1.3, les ensembles G et N contiennent un seul cycle. La Figure 1.8 montre les deux fusions possibles – selon le choix des arêtes adjacentes au télomère – pour ces deux cycles. Pour des raisons de clarté, on choisit de dessiner ces cycles sous forme de *cercles* où le sommet $\circ$ est répété deux fois.

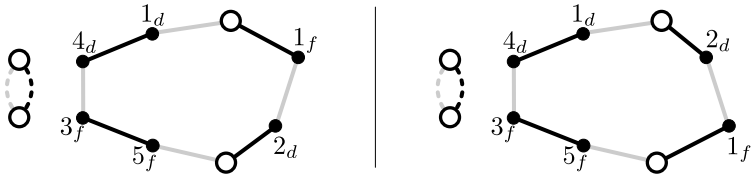

Figure 1.8. *Deux façons de fusionner le cycle gris $(5_f, 3_f, 4_d, 1_d, \circ)$ et le cycle noir $(1_f, 2_d, \circ)$ pour créer un cycle équilibré. (Les cycles en pointillé ne sont pas utilisés.)*

Encore ici, la distance est préservée, puisque chaque opération DCJ qui scinde l'un de ces cycles en deux donne soit un cycle complet et un cycle non-équilibré, soit deux cycles incomplets.

1.3.2.1. *Calcul du nombre de décompositions*

Nous sommes maintenant en mesure d'estimer le nombre de décompositions différentes du graphe des cassures. En effet, comme les cycles équilibrés appartiennent à toute décomposition, il suffit de compter le nombre de façons de fusionner les éléments de G et de N.

Soit j le nombre d'éléments de G, et ℓ le nombre d'éléments de N. On peut supposer que $j \leq \ell$, car le résultat final sera symétrique par rapport à j et ℓ. Une décomposition du graphe de cassures est caractérisée par les choix suivants.

1) Le choix de k cycles de G qui seront fusionnés avec k cycles de N. Ce choix peut s'effectuer de $\binom{j}{k}$ façons, où $0 \leq k \leq j$.

2) Le choix d'une orientation pour chaque fusion, on a donc 2^k possibilités.

3) Les choix, pour chacun des k cycles, d'un cycle de N comme partenaire de fusion. Il y en a $(\ell)(\ell-1)\ldots(\ell-(k-1)) = k!\binom{\ell}{k}$.

Ce qui nous donne la formule :

$$\sum_{k=0}^{j} 2^k k! \binom{j}{k} \binom{\ell}{k}.$$

Cette somme donne bien les 3 décompositions de la Figure 1.8, puisque $j = \ell = 1$ dans cet exemple. Malheureusement, ce nombre croît très rapidement ce qui exclut la possibilité d'examiner chaque scenario optimal en un temps raisonnable, même pour des petites valeurs de j. Par exemple, quand $j = 16$, qui est une valeur plausible si l'on pense que j est du même ordre de grandeur que le nombre de chromosomes, la valeur de cette somme est strictement plus grande que $\sum_{k=0}^{j} 2^k k!$, qui est égale à 1 415 527 220 320 869 563.

La bonne nouvelle est que le nombre de cycles différents qui peuvent apparaître dans un scénario demeure restreint. En effet, le nombre de cycles différents obtenus de la fusion d'un cycle $g \in G$ avec un cycle $n \in N$ est $2j\ell$, et le nombre de cycles différents obtenus par complétion est $j + \ell$. Pour $j = \ell = 16$, ceci donne 480 cycles, ce qui reste très raisonnable.

Nous verrons, dans la section suivante, que certains calculs sur les différents scénarios optimaux ne dépendent que du nombre de cycles possible, ce qui les rendra applicables en pratique.

1.4. Données de contact et scénarios pondérés

Jusqu'ici nous avons traité le problème de réarrangements de manière très abstraite, au sens où les opérations DCJ peuvent se produire arbitrairement. En réalité, les chromosomes d'une cellule habitent un espace en trois dimensions, et la probabilité d'un réarrangement entre deux brins peut varier beaucoup, selon qu'un chromosome est entortillé sur lui-même, ou cohabite étroitement avec un autre.

Pour revenir à l'analogie des tuyaux de plomberie, raccorder par erreur la sortie d'eau chaude de Madame Tremblay de Verdun avec l'entrée d'eau froide de son voisin d'en haut, est beaucoup plus plausible que raccorder la sortie d'eau chaude de Madame Tremblay avec l'entrée d'eau froide de Monsieur Segura de Montpellier.

Dans cette section, nous introduisons une notion de *poids* qui va nous permettre de mesurer la plausibilité d'un scénario optimal. Comme dans plusieurs problèmes d'optimisation, une stratégie gloutonne qui consisterait à choisir la *meilleure* opération à chaque étape ne garantit pas que le scénario obtenu soit le meilleur globalement. En effet, nous savons qu'une opération DCJ sur un cycle réduit considérablement le choix des opérations subséquentes, puisque toutes les opérations subséquentes devront impérativement se produire à l'intérieur d'un des deux cycles nouvellement créés.

Nous présentons ici un algorithme polynomial qui permet d'explorer tous les scénarios optimaux possibles.

1.4.1. *Modélisation des données de contact*

La Figure 1.9 illustre une configuration tri-dimensionnelle fictive des génomes $A = (\circ$ -1 4 5 -3 $\circ)$ $(\circ$ -8 6 9 7 -2 $\circ)$ et $B = (\circ$ 1 2 3 4 5 $\circ)$ $(\circ$ 6 7 8 9 $\circ)$ de la Figure 1.3. Les adjacences sont représentées, dans cette figure, par des segments de chromosome en pointillé. On y voit, par exemple, que certaines paires d'adjacences sont beaucoup plus proches entre elles que d'autres.

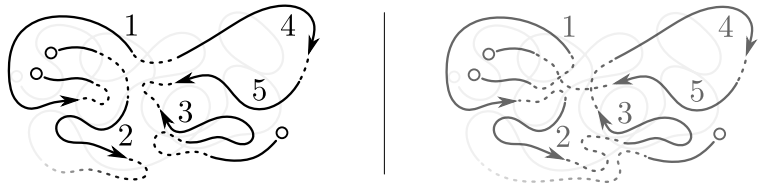

Figure 1.9. *Dans le panneau de gauche on voit le premier chromosome du génome A, ainsi qu'une partie du second chromosome. On remarque que l'adjacence $\{1_d, 4_d\}$ est beaucoup plus proche de l'adjacence $\{3_f, 5_f\}$ que de l'adjacence $\{1_f, \circ\}$.*

Une approche proposée pour mesurer la proximité relative de segments d'ADN est basée sur le *nombre de contacts* entre les segments (Lieberman-Aiden *et al*. 2009), mesuré grâce à des expériences en laboratoire. Ces données sont ensuite transformées en valeurs réelles, sous l'hypothèse que la distance physique entre deux segments est inversement proportionnelle à leur nombre de contacts.

Pour intégrer ces informations à notre modèle, nous associons à chaque adjacence $\{u, v\}$, le segment x d'ADN correspondant. Une opération DCJ sera modélisée comme dans les sections précédentes, en rajoutant l'information quant au sort des segments impliqués. Par exemple, l'opération $\{u, v\}, \{r, s\} \rightarrow \{u, r\}, \{v, s\}$ deviendra :

$$(\{u, v\}, x), (\{r, s\}, y) \rightarrow (\{u, r\}, x), (\{v, s\}, y) \quad \text{ou}$$
$$(\{u, v\}, x), (\{r, s\}, y) \rightarrow (\{u, r\}, y), (\{v, s\}, x),$$

selon que les adjacences créées héritent du segment x ou du segment y. La Figure 1.10 donne un exemple d'une telle opération.

Le *coût* d'une opération DCJ impliquant l'adjacence $(\{u, v\}, x), (\{r, s\}, y)$ est donné par la distance physique entre les segments x et y. Le *coût* d'un scénario est la

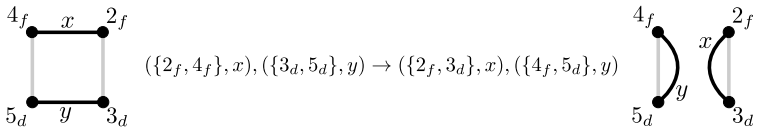

Figure 1.10. *Division d'un cycle avec une opération DCJ étiquetée*

somme des coûts de ses opérations DCJ. Le problème algorithmique que nous voulons résoudre est le suivant :

Étant donné deux génomes, et une fonction de coût pour chaque opération DCJ, calculer le coût d'un scénario de coût minimal parmi tous les scénarios optimaux.

Supposons, pour l'instant, que nous connaissons le coût minimal pour résoudre chaque cycle possible des diverses décompositions du graphe de cassures – il y en a $c+i+2j\ell+j+\ell$. On peut alors trouver un scénario de coût minimal avec l'algorithme suivant :

1) Calculer les coûts des c cycles complets et de i cycles incomplets.

2) Calculer le coût des $2j\ell + j + \ell$ cycles fusionnés possibles.

3) Trouver un couplage des cycles gris et noirs de coût minimal.

4) Calculer le coût total des cycles retenus.

Trouver un couplage des cycles gris et noirs de coût minimal se ramène à un problème classique de la théorie des graphes : *"trouver un couplage parfait de coût minimum dans un graphe biparti dont les arêtes sont pondérées"*. Martello (2010) en retrace l'historique qui remonte au XIX^e Siècle. Plusieurs algorithmes polynomiaux ont été proposés depuis, notamment ceux basés sur la *méthode hongroise* (Kuhn 1955).

Nous esquissons ici la façon de construire le graphe biparti dont les sommets sont *à gauche* ou *à droite*. Les sommets de gauche sont les j cycles de G, ainsi que ℓ cycles pointillé gris ; les sommets de droite sont les ℓ cycles de N, ainsi que j cycles pointillé noirs. Soit $g \in G$, g_p un cycle pointillé gris, $n \in N$ et n_p un cycle pointillé noir. Les coûts associés aux différentes arêtes reliant ces sommets sont :

1) $c(g, n_p)$ et $c(g_p, n)$ sont les coûts associés aux cycles fusionnés avec des arêtes en pointillé, comme illustré dans la Figure 1.7.

2) $c(g, n)$ est le minimum des coûts des deux fusions possibles de g et n, comme illustré dans la Figure 1.8.

3) $c(g_p, n_p) = 0$, puisque ces cycles ne contiennent pas de gènes.

Un couplage *parfait* est un ensemble d'arêtes qui joignent un et un seul sommet de gauche du graphe avec un sommet de droite. Donc une solution au problème de

couplage de coût minimum nous donnera une façon de fusionner les cycles non-équilibrés du graphe de cassures à coût minimal.

Dans la prochaine section, nous résolvons le problème du calcul du coût minimal d'un cycle.

1.4.2. *Arbres planaires et algorithme d'exploration*

Comme nous l'avons vu, tous les scénarios optimaux étiquetés pour résoudre un cycle de longueur $2m$ ont la même longueur $m-1$: ce qui les distingue au niveau de leur coût est l'ensemble des $m-1$ paires d'étiquettes impliquées dans chaque opération DCJ. En effet, le coût d'un cycle sera la somme des coûts de ces paires d'étiquettes.

Dans cette section, nous établissons une bijection entre les scénarios optimaux étiquetés pour résoudre un cycle de longueur $2m$ et les *arbres planaires* à $m-1$ branches, qui vont représenter un ensemble de paires d'étiquettes compatibles avec un scénario d'opérations DCJ. Puis nous allons montrer comment construire efficacement un arbre de coût minimal.

Nous commençons par le rappel de certaines propriétés des arbres et des arbres planaires.

1.4.2.1. *Arbres planaires*

Un *arbre* est un graphe connexe sans cycles. Un arbre à m sommets a toujours $m-1$ arêtes, ici appelées *branches*. Une *feuille* est un sommet de degré 1 : un arbre avec au moins une branche a toujours au moins deux feuilles.

Pour les besoins du texte, nous dirons qu'un arbre est *planaire* si ses sommets sont l'intervalle $[0..m-1]$, sont placés, dans cet ordre, sur des points équidistants d'un cercle, et dont les branches *ne se croisent pas*. Notons que, dans ce cas, les branches d'un arbre planaire sont toutes des cordes du cercle, on peut donc associer aux branches leur *longueur*, et ordonner celles-ci en ordre non-décroissant. La Figure 1.11 montre un arbre planaire à 8 sommets. Les branches les plus courtes de cet arbre sont $\{1,2\}$, $\{3,4\}$ et $\{6,7\}$. Les plus longues sont $\{0,3\}$ et $\{0,5\}$.

Un intervalle de sommets consécutifs sur le cercle à m sommets sera noté $[i..j]$. Dans le contexte d'un cercle, cette notation doit être interprétée comme l'ensemble des sommets allant de i à j, en parcourant le cercle dans le sens des aiguilles d'une montre. Par exemple, dans la Figure 1.11, l'intervalle $[1..3]$ désigne l'ensemble $\{1,2,3\}$, alors que l'intervalle $[3..1]$ désigne l'ensemble $\{3,4,5,6,7,0,1\}$. Chaque branche $\{i,j\}$ d'un arbre planaire P à m branches détermine deux sous-arbres de P, notés $P[i..j]$ et $P[j..i]$, qui contiennent les sommets i et j ainsi que les sommets entre i et j de part et d'autre de la branche.

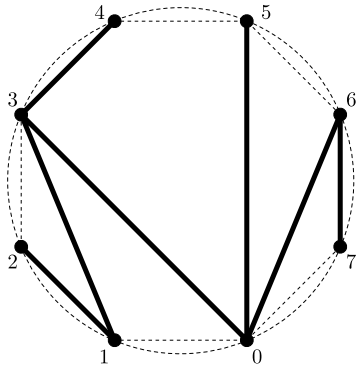

Figure 1.11. *Arbre planaire à 8 sommets, inscrit dans un cercle.*

Enfin, nous aurons besoin de la proposition suivante :

PROPOSITION 1.2.– *Dans un arbre planaire, il existe au moins une feuille qui est adjacente à une de ses voisines sur le cercle.*

DÉMONSTRATION 1.0.– Soit i une feuille de l'arbre P, j l'unique sommet adjacent à i, et tel que la branche $\{i, j\}$ est de longueur minimale. Comme $\{i, j\}$ est une branche, $P[i..j]$ et $P[j..i]$ seront tous les deux des arbres. Si l'un de ces arbres ne contient qu'une seule branche, alors la feuille i est adjacente à l'une de ses deux voisines sur le cercle.

Sinon, supposons que le nombre de sommets de $P[i..j]$ est inférieur ou égal à celui de $P[j..i]$, alors toutes les branches de $P[i..j]$ sont plus courtes que la branche $\{i, j\}$. Comme l'arbre $P[i..j]$ a au moins deux feuilles, l'une d'entre elles appartient à l'arbre P, ce qui contredit l'hypothèse que la branche $\{i, j\}$ est de longueur minimale. □

1.4.2.2. *Bijection entre scénarios étiquetés et arbres planaires*

Notre objectif est de trouver un scénario optimal de coût minimal pour un cycle équilibré de longueur $2m$. Comme le coût d'un scénario est égal à la somme des coûts des paires d'étiquettes impliquées dans une opération DCJ, nous considérons comme équivalents deux scénarios qui impliquent le même ensemble de paires d'étiquettes. Dans cette section nous donnons une bijection entre les ensembles de paires d'étiquettes et les arbres planaires.

Soit un cycle équilibré de longueur $2m$, étiqueté par les segments d'ADN $(x_0, x_1, \ldots, x_{m-1})$. Le choix de x_0 est arbitraire, mais les autres doivent respecter l'ordre induit par le cycle.

Étant donné un scénario optimal étiqueté, donc de longueur $m-1$, on lui associe l'arbre planaire sur m sommets de la façon suivante : si une opération DCJ implique la paire d'étiquettes $\{x_i, x_j\}$, alors on ajoute la branche $\{i, j\}$. L'arbre correspondant ne dépend que de l'ensemble des paires d'étiquettes. Le résultat est un arbre planaire car deux branches croisées impliqueraient que l'une des opérations DCJ du scénario agirait sur deux cycles différents.

Inversement, étant donné un arbre planaire, on construit récursivement un scénario de longueur $m-1$ en remarquant que si une feuille i est adjacente à une de ses voisines $j = (i+1) \mod m$ ou $j = (i-1) \mod m$, on peut toujours effectuer une opération DCJ étiquetée qui crée un nouveau cycle de longueur 2 avec l'étiquette x_i. Le long cycle restant est de longueur $2(m-1)$, et aucune des opérations DCJ nécessaires pour le résoudre n'implique l'étiquette x_i, puisque cette dernière était une feuille. La Proposition 1.2 garantit l'existence d'une telle feuille.

1.4.2.3. *Construction d'un arbre de coût minimal*

Les données de contact nous permettent de calculer le coût $c(x_i, x_j)$ associé à une opération DCJ impliquant les étiquettes x_i et x_j, donc le coût $c(i, j)$ associé à une branche $\{i, j\}$ d'un arbre planaire. Par exemple, en agrandissant le génome fictif de la Figure 1.9, on obtient des distances relatives entre les paires d'adjacences. Les adjacences impliquées correspondent au cycle du panneau de gauche de la Figure 1.8.

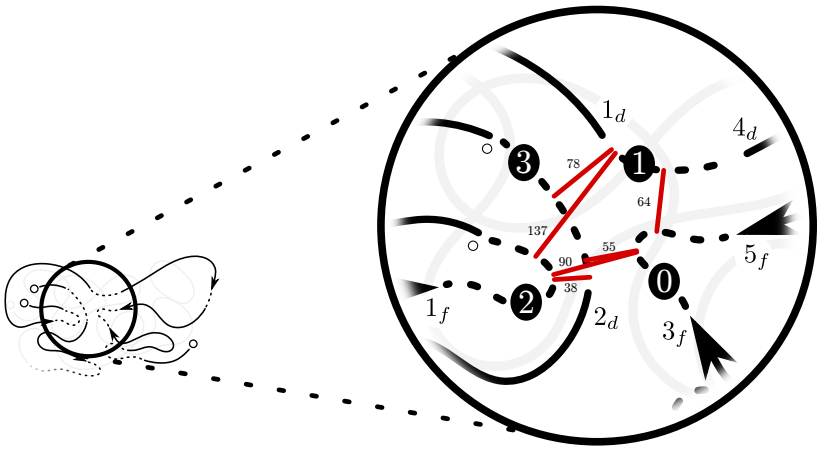

Figure 1.12. *Un gros plan des adjacences des génomes fictifs. On note, par exemple, que la distance entre l'adjacence $\{3_f, 5_f\}$, étiquetée par $i = 0$, et l'adjacence $\{1_d, 4_d\}$, étiquetée par $j = 1$, est de 64. On en déduit le coût $c(0, 1) = 64$.*

Donc nous réduisons le problème de trouver le coût minimal pour un cycle de longueur $2m$ à celui de trouver un arbre planaire P de coût minimal c_{min} dans l'ensemble $\mathcal{P}_{[0..m-1]}$ de tous les arbres planaires avec les sommets dans l'intervalle $[0..m-1]$. En général, si $[k..\ell]$ est un sous-intervalle de $[0..m-1]$, nous noterons $c_{min}(\mathcal{P}_{[k..\ell]})$ le coût minimal d'un arbre planaire dont les sommets sont $[k..\ell]$ dans le cercle de taille m.

Soit $\{i,j\}$ une branche d'un arbre planaire P. Le coût $c(P)$ peut être calculé au moyen de la formule suivante qui nous sera utile pour construire une formule de récurrence :

$$c(P) = c(P[i..j]) + c(P[j..i]) - c(i,j).$$

En scindant le cercle en deux le long de la branche $\{i,j\}$ on peut ramener le calcul du coût d'un arbre dont l'ensemble des sommets est $[0..m-1]$ au coût de deux arbres sur des cercles plus petits. Ces deux arbres sont des éléments des ensembles $\mathcal{P}_{[i..j]}$ et $\mathcal{P}_{[j..i]}$ *et qui contiennent nécessairement la branche* $\{i,j\}$. Le problème se réduit donc au calcul du coût minimal d'un arbre P dans $\mathcal{P}_{[i..j]}$, *sachant* que la branche $\{i,j\}$ appartient à P, ce que nous noterons $c_{min}(\mathcal{P}_{[i..j]}|\{i,j\})$.

En testant chacune des cordes $\{i,j\}$ possibles du cercle dont les sommets sont $[k..\ell]$, on en déduit l'équation :

$$c_{min}(\mathcal{P}_{\mathcal{I}=[k..\ell]}) = \min_{i\neq j\in\mathcal{I}} \{c_{min}(\mathcal{P}_{[i..j]\cap\mathcal{I}}|\{i,j\}) + c_{min}(\mathcal{P}_{[j..i]\cap\mathcal{I}}|\{i,j\}) - c(i,j)\}$$

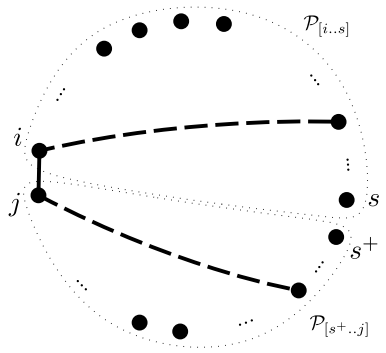

Figure 1.13. *Décomposition de l'arbre contenant la branche $\{i,j\}$ en deux sous-arbres $P[i..s]$ et $P[s^+..j]$. Un des deux sous-arbres peut potentiellement être vide.*

Comme les sommets i et j sont voisins dans $\mathcal{P}_{[i..j]\cap[k..\ell]}$, un arbre planaire contenant cette branche peut se décomposer en deux sous-arbres $P[i..s]$ et $P[s^+..j]$,

où s^+ est le sommet suivant s dans le cercle $[k..\ell]$, donc $s+1$ sauf si $s=\ell$, et on calcule le coût minimal sur toutes les valeurs de s :

$$c_{min}(\mathcal{P}_{[i..j]\cap\mathcal{I}}|\{i,j\}) = \min_{s\in[i..j-1]\cap\mathcal{I}}\{c(i,j)+c_{min}(\mathcal{P}_{[i..s]})+c_{min}(\mathcal{P}_{[s^+..j]})\},$$

sachant que $c_{min}(\mathcal{P}_{[i..i]})=0$.

Une façon efficace de calculer $c_{min}(\mathcal{P}_{[0..m-1]})$ est de calculer les valeurs de $c_{min}(\mathcal{P}_{[i..j]\cap[k..\ell]}|\{i,j\})$ et de $c_{min}(\mathcal{P}_{[i..j]})$ pour toutes les cordes $\{i,j\}$ du cercle, des plus courtes jusqu'aux plus longues. Les calculs sont élémentaires, et pour donner une idée du processus, nous montrons deux extraits des calculs pour le cycle $(5_f, 3_f, 4_d, 1_d, \circ, 1_f, 2_d, \circ)$, et en utilisant la matrice coûts $c(i,j)$ donnée par les valeurs de la Figure 1.12. La valeur de $c_{min}(\mathcal{P}_{\mathcal{I}=[0..3]})$ est donnée par :

$$c_{min}(\mathcal{P}_{\mathcal{I}=[0..3]}) = \min\begin{cases} c_{min}(\mathcal{P}_{[0..1]\cap\mathcal{I}}|\{0,1\}) + c_{min}(\mathcal{P}_{[1..0]\cap\mathcal{I}}|\{0,1\}) - c(0,1) \\ c_{min}(\mathcal{P}_{[0..2]\cap\mathcal{I}}|\{0,2\}) + c_{min}(\mathcal{P}_{[2..0]\cap\mathcal{I}}|\{0,2\}) - c(0,2) \\ c_{min}(\mathcal{P}_{[0..3]\cap\mathcal{I}}|\{0,3\}) + c_{min}(\mathcal{P}_{[3..0]\cap\mathcal{I}}|\{0,3\}) - c(0,3)* \\ c_{min}(\mathcal{P}_{[1..2]\cap\mathcal{I}}|\{1,2\}) + c_{min}(\mathcal{P}_{[2..1]\cap\mathcal{I}}|\{1,2\}) - c(1,2) \\ c_{min}(\mathcal{P}_{[1..3]\cap\mathcal{I}}|\{1,3\}) + c_{min}(\mathcal{P}_{[3..1]\cap\mathcal{I}}|\{1,3\}) - c(1,3) \\ c_{min}(\mathcal{P}_{[2..3]\cap\mathcal{I}}|\{2,3\}) + c_{min}(\mathcal{P}_{[3..2]\cap\mathcal{I}}|\{2,3\}) - c(2,3) \end{cases}$$

Puisque les sommets 0 et 3 sont voisins, $c_{min}(\mathcal{P}_{[3..0]\cap\mathcal{I}}|\{0,3\}) = c(0,3)$, donc l'expression étoilée est égale à $c_{min}(\mathcal{P}_{[0..3]\cap[0..3]}|\{0,3\})$, donc à :

$$c_{min}(\mathcal{P}_{[0..3]\cap[0..3]}|\{0,3\}) = \min\begin{cases} c(0,3)+c_{min}(\mathcal{P}_{[0..0]})+c_{min}(\mathcal{P}_{[1..3]}) \\ c(0,3)+c_{min}(\mathcal{P}_{[0..1]})+c_{min}(\mathcal{P}_{[2..3]})* \\ c(0,3)+c_{min}(\mathcal{P}_{[0..2]})+c_{min}(\mathcal{P}_{[3..3]}) \end{cases}$$

De nouveau, l'expression étoilée se calcule facilement et donne $c(0,3)+c(0,1)+c(2,3) = 55+64+38 = 157$, ce qui est le minimum recherché, et l'arbre planaire correspondant contient les branches $\{0,3\}$, $\{0,1\}$ et $\{2,3\}$. La Figure 1.14 montre ce scénario de poids minimal, l'arbre planaire reliant les arêtes étiquetées par $[0,1,2,3]$, ainsi que le scénario correspondant sur les génomes.

1.5. Conclusion

Ce chapitre a couvert diverses techniques qui permettent de comparer les positions des gènes communs de deux espèces, autant dans leur organisation linéaire que tri-dimensionnelle. La comparaison de deux génomes apporte une quantité limitée d'informations, comme le nombre minimal d'opérations DCJ qui séparent deux espèces, mais la nature et l'ordre de ces opérations reste généralement non résolu : pour y arriver, il faut comparer un grand nombre d'espèces simultanément, ou introduire plus d'informations biologiques.

Pendant la majeure partie du XXième siècle les données sur l'ordre des gènes étaient pratiquement inexistantes. Les expériences du groupe de Morgan avaient déjà

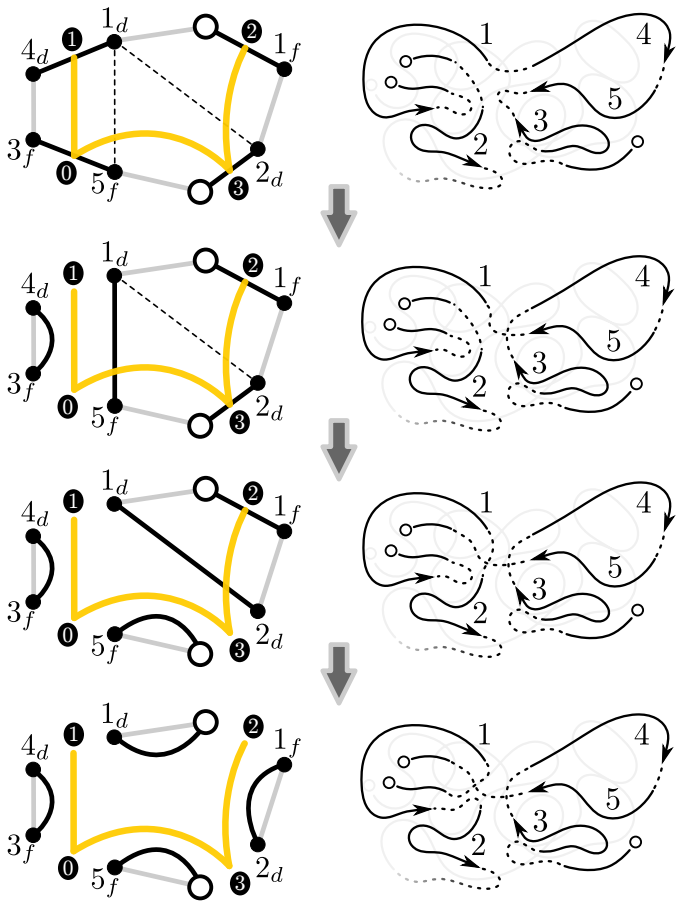

Figure 1.14. *Le scénario de poids minimal pour le cycle* $(5_f, 3_f, 4_d, 1_d, \circ, 1_f, 2_d, \circ)$. *L'arbre planaire associé est composé des arêtes* $\{0,1\}$, $\{0,3\}$ *et* $\{2,3\}$. *La colonne de droite montre l'effet des opérations DCJ sur les génomes* A *et* B.

mis en lumière les relations possibles de l'organisation des gènes de quelques espèces de mouches drosophiles de Californie. Comme le nombre de réarrangements était restreint, ils ont pu reconstruire les relations phylogénétiques entre ces espèces avec des techniques élémentaires (Sturtevant and Dobzhansky 1936).

Une des premières études systématiques impliquant plusieurs espèces, parue dans le journal PNAS en 1992, compare l'ordre des gènes dans 16 mitocondries d'insectes et donne un premier cadre formel pour cette problématique (Sankoff *et al*. 1992). Cet

article a constitué le coup d'envoi de la course aux premiers algorithmes efficaces pour résoudre le problème de la *distance* d'inversions entre deux génomes (Hannenhalli and Pevzner 1995). Ces calculs de distance permettaient dès lors d'appliquer des techniques rapides de construction d'arbres phylogénétiques (voir Chapitre 6). Les modèles de réarrangements génomiques se sont multipliés et raffinés au cours des trente dernières années : en plus des inversions, toute une panoplie d'opérations biologiques a été définie pour s'adapter aux différentes branches de l'arbre de la vie. Nous conseillons au lecteur de se référer à Fertin *et al*. (2009) pour une étude approfondie du sujet.

Aujourd'hui, le séquençage du génome d'une espèce est presque devenu une routine, et les données biologiques s'accumulent beaucoup plus vite que les outils d'analyse. Il reste d'autre part un important travail à faire avant que les techniques développées dans ce chapitre puissent être appliquées à grande échelle. Une première difficulté vient du fait que plusieurs génomes sont reconstruits à partir de courtes séquences en utilisant un génome d'une espèce proche comme référence, ce qui tend à masquer certains réarrangements récents. Nous avons aussi supposé, en début de chapitre, que le problème de décider si deux gènes sont homologues était résolu, et que les génomes ne comportaient pas de gènes dupliqués ou manquants. Dans la réalité biologique, hélas, ces hypothèses sont rarement vérifiées.

L'arsenal pour attaquer ces problèmes est à la fois technologique et informatique. Le séquençage des génomes devient de plus en plus performant, et l'assemblage de courtes séquences fait place à la lecture directe de très longues séquences d'ADN (Wenger *et al*. 2019). L'identification de gènes homologues intègre peu à peu des informations qui complètent la simple ressemblance, comme le voisinage d'un gène dans son génome, ou sa généalogie individuelle par rapport à celle de son espèce (Chauve *et al*. 2013). Lorsque le nombre de gènes dupliqués est restreint, les techniques de programmation linéaire permettent de maîtriser l'explosion combinatoire des calculs nécessaires (Shao *et al*. 2015, Simonaitis *et al*. 2019).

À long terme, l'objectif est de comprendre quand, comment et pourquoi les réarrangements ont contribué à la diversité biologique qu'on observe. Nous avons au moins un autre siècle pour y arriver.

1.6. Bibliographie

Baryshnikov, Y. (2001), *On Stokes Sets*, Springer Netherlands, Dordrecht, pp. 65–86.

Bergeron, A., Stoye, J. (2013), *The Genesis of the DCJ Formula*, Springer London, London, pp. 63–81.

Castle, W. E. (1919), Is the arrangement of the genes in the chromosome linear ?, *Proceedings of the National Academy of Sciences*, 5(2), 25–32.

Chauve, C., El-Mabrouk, N., Guéguen, L., Semeria, M., Tannier, E. (2013), Duplication, rearrangement and reconciliation : A follow-up 13 years later, *in* Models and Algorithms for Genome Evolution, Springer, pp. 47–62.

Crick, F. H. C., Watson, J. D., Bragg, W. L. (1954), The complementary structure of deoxyribonucleic acid, *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 223(1152), 80–96.

De Chiara, M., Friedrich, A., Barré, B., Breitenbach, M., Schacherer, J., Liti, G. (2020), Discordant evolution of mitochondrial and nuclear yeast genomes at population level, *BMC Biology*, 18(1), 49.

Fertin, G., Labarre, A., Rusu, I., Vialette, S., Tannier, E. (2009), *Combinatorics of genome rearrangements*, MIT press.

Hannenhalli, S., Pevzner, P. A. (1995), Transforming men into mice (polynomial algorithm for genomic distance problem), *in* 36th Annual Symposium on Foundations of Computer Science, Milwaukee, Wisconsin, USA, 23-25 October 1995, IEEE Computer Society, pp. 581–592.

IJdo, J. W., Baldini, A., Ward, D. C., Reeders, S. T., Wells, R. A. (1991), Origin of human chromosome 2 : an ancestral telomere-telomere fusion., *Proceedings of the National Academy of Sciences*, 88(20), 9051–9055.

Kuhn, H. W. (1955), The hungarian method for the assignment problem, *Naval Research Logistics Quarterly*, 2(1-2), 83–97.

Lieberman-Aiden, E., van Berkum, N. L., Williams, L., Imakaev, M., Ragoczy, T., Telling, A., Amit, I., Lajoie, B. R., Sabo, P. J., Dorschner, M. O., Sandstrom, R., Bernstein, B., Bender, M. A., Groudine, M., Gnirke, A., Stamatoyannopoulos, J., Mirny, L. A., Lander, E. S., Dekker, J. (2009), Comprehensive mapping of long-range interactions reveals folding principles of the human genome, *Science*, 326(5950), 289–293.

Martello, S. (2010), Jenőegerváry : from the origins of the hungarian algorithm to satellite communication, *Central European Journal of Operations Research*, 18(1), 47–58.

Morgan, T. H., Sturtevant, A. H., Bridge, C. B. (1920), The evidence for the linear order of the genes, *Proceedings of the National Academy of Sciences*, 6(4), 162–164.

Nikiforova, M. N., Stringer, J. R., Blough, R., Medvedovic, M., Fagin, J. A., Nikiforov, Y. E. (2000), Proximity of chromosomal loci that participate in radiation-induced rearrangements in human cells, *Science*, 290(5489), 138–141.

Pearson, H. (2006), What is a gene ?, *Nature*, 441(7092), 398–401.

Sankoff, D., Leduc, G., Antoine, N., Paquin, B., Lang, B. F., Cedergren, R. (1992), Gene order comparisons for phylogenetic inference : evolution of the mitochondrial genome, *Proceedings of the National Academy of Sciences*, 89(14), 6575–6579.

Shao, M., Lin, Y., Moret, B. M. (2015), An exact algorithm to compute the double-cut-and-join distance for genomes with duplicate genes, *Journal of Computational Biology*, 22(5), 425–435.

Simonaitis, P., Chateau, A., Swenson, K. (2019), A general framework for genome rearrangement with biological constraints, *Algorithms for Mol. Biol.*, 14(15).

Sturtevant, A., Dobzhansky, T. (1936), Inversions in the third chromosome of wild races of drosophila pseudoobscura and their use in the study of the history of the species, *Proceedings of the National Academy of Sciences*, 22, 448–450.

Swenson, K. M., Blanchette, M. (2019), Large-scale mammalian genome rearrangements coincide with chromatin interactions, *Bioinformatics*, 35(14), i117–i126.

Véron, A. S., Lemaitre, C., Gautier, C., Lacroix, V., Sagot, M.-F. (2011), Close 3D proximity of evolutionary breakpoints argues for the notion of spatial synteny., *BMC Genomics*, 12, 303.

Wenger, A. M., Peluso, P., Rowell, W. J., Chang, P.-C., Hall, R. J., Concepcion, G. T., Ebler, J., Fungtammasan, A., Kolesnikov, A., Olson, N. D. *et al*. (2019), Accurate circular consensus long-read sequencing improves variant detection and assembly of a human genome, *Nature biotechnology*, 37(10), 1155–1162.

Yaffe, E., Farkash-Amar, S., Polten, A., Yakhini, Z., Tanay, A., Simon, I. (2010), Comparative analysis of DNA replication timing reveals conserved large-scale chromosomal architecture., *PLoS Genet*, 6(7), e1001011.

Yancopoulos, S., Attie, O., Friedberg, R. (2005), Efficient sorting of genomic permutations by translocation, inversion and block interchange, *Bioinformatics*, 21(16), 3340–3346.