Rapport de stage

Assemblage *de novo* des génomes mitochondriaux et chloroplastiques de 10 accessions naturelles d'*Arabidopsis thaliana*

Auteur
Myriam Shafie

Tuteur
Fabienne Granier

Equipe Bioinformatique et Informatique
Institut Jean-Pierre Bourgin
INRA Centre de Versailles-Grignon

Remerciements

Table des matières

1 Présentation de l'organisme d'accueil

L'Institut Jean-Pierre Bourgin est une unité mixte de recherche INRA-AgroParisTech consacrée à l'étude de la biologie végétale. Il comporte environ 350 personnes, réparties en 25 équipes.

Les thématiques étudiées à l'IJPB sont très variées. Elles sont regroupés en 5 grands pôles :

- Morphogenèse, Signalisation, Modélisation
- Dynamique et expression des génomes
- Adaptation des plantes à leur environnement
- Reproduction et Graines
- Paroi végétale, fonction et usage

J'ai réalisé mon stage au sein de l'équipe Bioinformatique et Informatique.

2 Introduction

2.1 Mise en situation biologique

Les plantes vertes possèdent trois compartiments génétiques : le noyau, la mitochondrie (mt) et le chloroplaste (ct). La mitochondrie et le chloroplaste étaient à l'origine des bactéries endosymbiotes ; elles ont été intégrées à la cellule végétale lors de l'évolution et sont devenus des organites cellulaires. Au cours de cette intégration, une grande partie des génomes mt et ct a été transféré vers le noyau.

Cependant, la mitochondrie et le chloroplaste conservent un petit nombre de gènes cruciaux à leur fonctionnement. Ces gènes ne se suffisent bien évidemment pas à eux-mêmes. Les produits codés par le noyau et par les organites doivent absolument interagir entre eux pour que la mitochondrie et le chloroplaste fonctionnement correctement.

Ainsi, une co-évolution du génome nucléaire et cytoplasmique paraît obligatoire, car un changement dans l'un doit forcément être refléter par un changement dans l'autre afin de perpétuer leur interaction. Ce mécanisme fait que les génomes du noyau et des organites vont s'adapter à l'environnement de manière coopérative.

La figure 1 illustre ce principe. Il présente trois étapes de l'histoire évolutive d'une plante. Initialement, la plante possède l'allèle ancestrale N0 pour un gène du noyau et l'allèle ancestrale C0 pour un gène d'un des organites. Les produits de ces deux gènes interagissent correctement au niveau physiologique. Cet interaction contribue à la fitness globale de la plante et à son phénotype.

Le gène cytoplasmique subit alors une mutation. La plante possède l'allèle dérivée C1 pour le gène cytoplasmique mais toujours l'allèle ancestrale N0 pour le gène nucléaire. L'interaction des produits des deux gènes est toujours possible mais elle est un peu moins efficace au niveau physiologique. Cependant, la fitness globale de la plante n'est pas diminuée, par exemple parce que la mutation C1 permet à la plante de s'adapter à un changement dans son environnement. La mutation C1 va donc être fixée.

Enfin, une mutation N1 du gène nucléaire va être positivement sélectionnée. En effet, cette mutation permet de restaurer une interaction optimale entre les produits des gènes nucléaire et cytoplasmique au niveau physiologique. Les gènes nucléaire et cytoplasmique ont donc évolués main dans la main pour assurer l'adaptation de la plante à son nouvel environnement.

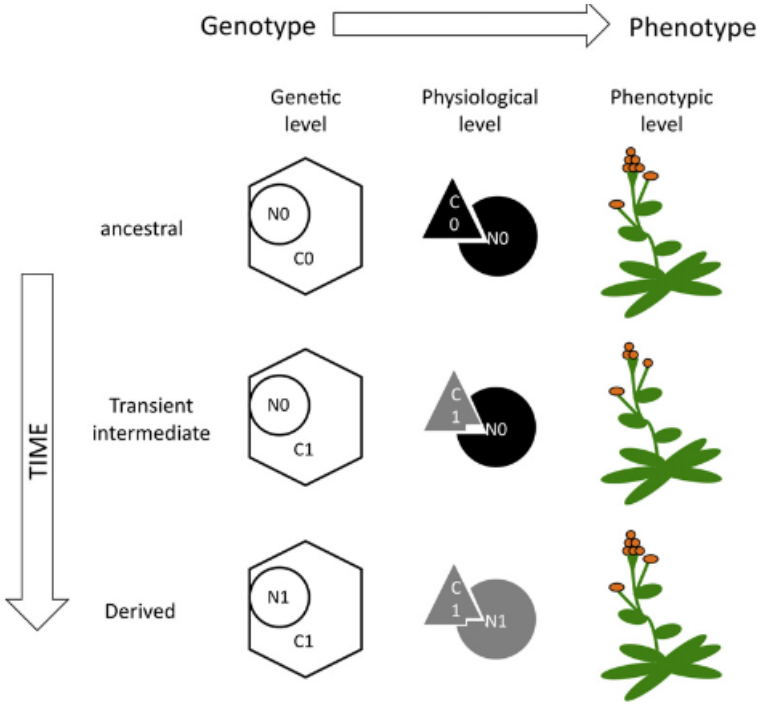


FIGURE 1: La coadaptation nucléo-cytoplasmique [3]

Le meilleur moyen de mettre en évidence expérimentalement la coadaptation nucléo-cytoplasmique est de la briser.

Considérons deux plantes d'une même espèce (donc interfertiles) mais appartenant à deux sous-espèces différentes ayant divergées depuis longtemps (par exemple, une plante poussant en France et une autre aux Etats-Unis). Le cytoplasme et le noyau de ces deux plantes ne sont pas compatibles.

On peut obtenir par croisements une plante qui possède le génome nucléaire de l'une et le génome cytoplasmique de l'autre. On mesure ensuite les caractéristiques phénotypiques qui nous intéressent chez la plante et ses parents. On peut ainsi repérer et quantifier l'influence de la coadaptation nucléo-cytoplasmique sur ces traits.

2.2 Le projet CytoPhéno

Le projet CytoPhéno (signifiant *Co-adaptation nucléo-cytoplasmique et phénotypes adaptatifs des plantes*) a été mis en place en 2012. Il est piloté par Françoise Budar, directrice de l'équipe Organites et Reproduction de l'IJPB, mais il implique également de nombreuses autres partenaires de l'INRA et du CNRS.

Ce projet possède deux volets de recherche, bien évidemment liés : la co-adapatation nucléo-cytoplasmique et le rôle du cytoplasme dans l'adaptation de la plante à son environnement.

La plante choisie pour l'étude de ces thèmes est *Arabidopsis Thaliana*. En effet, outre tous les avantages qui font d'*Arabidopsis* un organisme modèle (petit génome, cycle de vie rapide etc.), cette plante possède une forte diversité génétique et est présente naturellement dans un très grand nombre d'habitats naturels, ce qui en fait une référence pour les études portant sur l'adaptation. De plus, les chercheurs de l'IJBP ont mis en évidence en 2010 l'existence de la co-adaptation nucléo-cytoplasmique chez *Arabidopsis Thaliana*.[5]

Ils ont choisi de concentrer leur étude sur la plus petite core collection (n=8) disponible au *Versailles Arabidopsis Stock Center*. Une core collection est un petit groupe d'accessions naturelles d'*Arabidopsis Thaliana* qui capture au mieux la diversité génétique et morphologique de l'espèce.[4]

A partir de ces 8 accessions naturelles, les chercheurs ont crée 56 cytolignées, représentant toutes les combinaisons noyau/cytoplasme possibles.

Le projet CytoPhéno comporte différentes étapes.

1. l'étude des variantes génétiques mitochondrial et chloroplastique pour les 8 accessions naturelles
2. la production d'un grand nombre de graines pour les accessions naturelles et les cytolignées
3. l'identification des gènes nucléaires et cytoplasmiques impliqués dans la co-adaptation
4. le phénotypage de la core collection et des cytolignées pour différents traits (germination, réponse à différents niveaux de nitrogène etc.)
5. l'analyse statistique des résultats afin d'évaluer l'effet sur le phénotype du noyau, du cytoplasme et de l'interaction nucléo-cytoplasmique

Suite à ces tâches, nous pourrons répondre aux questions suivantes :

Quels phénotypes sont impactés quand la co-adaptation nucléo-cytoplasmique est rompue ? Et quels gènes sont impliqués ?

Quels traits potentiellement adaptatifs sont modifiés par des variations du génome cytoplasmique ? A quel point ces traits sont également impactés par la co-adaptation nucléo-cytoplasmique ?

2.3 La tâche 1 : Analyse bioinformatique

Lors de mon stage, j'ai travaillé sur la première tâche du projet CytoPhéno : l'analyse des génomes mitochondriaux et chloroplastique des 8 accessions naturelles.

Le but de cet tâche est de trouver des polymorphismes non neutres entre les différentes accessions. En effet, les gènes présents dans des versions significativement différentes selon les accessions sont peut-être impliqués dans la rupture de la coadaptation nucléo-cytoplasmique.

Pour trouver ces variants, il va falloir aligner ou assembler les génomes mitochondriaux et chloroplastiques des 8 accessions puis les annoter.

Les mitochondries des plantes évoluent peu par mutations ponctuelles et beaucoup par réarrangements. Des pans entiers de génomes vont donc être réordonnées entre nos 8 accessions naturelles et la référence dont nous disposons (génome mitochondrial de l'accession C24). On privilégie donc l'assemblage *de novo* par rapport à l'alignement pour la mitochondrie.

Le chloroplaste présente lui beaucoup de SNPs et peu de réarrangements. On pourrait donc l'aligner à une référence. Cependant, les données dont nous disposons contiennent à la fois les génomes mt et ct. Séparer les deux génomes est difficile car il y a des insertions du génome chloroplastique dans la mitochondrie et vice-versa. Les bioinformaticiens de l'IJPB ont donc optés pour un assemblage *de novo* simultané des génomes mt et ct.

3 Matériels et méthodes

3.1 Les données brutes

Nous disposons de 10 librairies de reads correspondant chacune à une accession.

Nom	Habitat
Jea	St-Jean Cap Ferrat [France]
Ita-0	Ibel Tazekka (Jebel Tazekka) [Maroc]
Cvi-0	Cap-Vert [Cap-Vert]
Bur-0	Burren [Irlande]
Blh-1	Bulhary [République Tchèque]
Oy-0	Oystese [Norvège]
Sha(hdara)	Shakdara River (Pamir) [Tadjikistan]
Ct-1	Catane [Italie]
Mr-0*	Monterosso [Italie]
Kz-9*	Atasu [Kazakhstan]

TABLE 1: Présentation des 10 accessions étudiées (* : ne fait pas partie de la core collection)

Outre les 8 membres de la core-collection, nous avons également étudié les génomes des accessions Mr-0 et Kz-9 d'*Arabidopsis Thaliana*. Ces deux accessions ont été ajoutées par Françoise Budar, la coordinatrice du projet, car elle présente un intérêt particulier pour son étude de la stérilité mâle cytoplasmique (recherche des gènes cytoplasmiques qui induisent la stérilité mâle chez certaines accessions d'*Arabidopsis Thaliana*).

Les génomes mitochondriaux et chloroplastiques des 10 accessions ont été séquencés en paired-end, Sha et Kz-9 par la technologie Illumina HiSeq 2000 (2X100 bases) et toutes les autres accessions par la technologie Illumina MiSeq 2000 (2X150 bases). Nous avons donc, pour chaque accession, deux fichiers fastq contenant les reads pairés.

3.2 Préparation des séquences

Avant d'aligner ou d'assembler ces reads, il est nécessaire d'effectuer une analyse de qualité. Elle nous permettra de détecter d'éventuelles contaminations de la

librairie ou des problèmes de séquençage (par exemple, une mauvaise qualité d'imagerie sur une flowcell va donner un sous-ensemble de reads de mauvaise qualité).

On effectue l'analyse de qualité avec fastqc. Cet outil possède plusieurs modules qui vont chacune évaluer un indicateur : la qualité des bases (selon leur position sur le read), la longueur des reads etc. Chacune des ces mesures est comparée à ce qui est attendu pour une librairie aléatoire. Si l'observé et le prédit sont proches, le module est validé.

Si le module est rejeté, cela signifie que la librairie étudiée diffère sur ce point d'une librairie aléatoire. Cette différence peut provenir d'une contamination ou d'un biais de séquençage mais elle peut également résulter de la réalité biologique de nos données (qui ne sont, bien évidemment, pas aléatoires).

Il est donc nécessaire de regarder non seulement le résultats des tests (module validé, non validé ou à la limite de validation) mais aussi la sortie graphique de chaque module. fastqc fournit une représentation graphique de chaque indicateur testé : boxplot des qualités de chaque base selon leur position dans le read, abondance des reads en fonction de leur longueur etc.

Si les indicateurs sont normaux pour notre librairie, on peut procéder au nettoyage des reads. Pour cela, on utilise l'outil Trimmomatic.

Trimmomatic va supprimer aux extrémités 3' et 5' les bases dont la qualité est inférieure à 20 (ce qui signifie que la base a une chance sur cent d'avoir été mal identifiée). Puis, il fait glisser une fenêtre de 4 nucléotides sur la séquence : si la qualité moyenne de la fenêtre est inférieure à 15, elle est supprimée. Les reads devenus trop courts (moins de 50 bases) sont ensuite jetés.

Après trimming, on refait une analyse de qualité sur les reads retenus pour vérifier que le nettoyage a bien amélioré la qualité de la librairie.

3.3 Alignement

Avant d'effectuer l'assemblage *de novo* de nos libraries nettoyées, on va d'abord les aligner à une référence : génome nucléaire et chloroplastique de l'accession Col-0 et génome mitochondrial de C24.

Pourquoi cet alignement ?

Il nous permettra :

- de quantifier le degré de contamination nucléaire de notre librairie
- d'évaluer la couverture pour chaque génome et chaque accession, ce qui est essentiel pour l'assemblage

Pour l'alignement, on utilise l'outil bwa. On autorise 2 événements indépendants seulement (par exemple, un mismatch plus un micro-indel). Ce critère stringent est adapté au génome mitochondrial, où il y a très peu de mutations ponctuelles.

Les alignements obtenus sont au format SAM, où chaque read correspond à une ligne. Si un read s'aligne à plusieurs endroits du génome, un seul alignement est choisi au hasard. Cela nous contraint à masquer le chromosome 2 de la référence. En effet, la quasi-totalité de la mitochondrie est insérée à l'intérieur de ce chromosome.[7] Donc, si nous ne le masquions pas, les reads mitochondriaux auraient un best hit sur la mitochondrie et un best hit sur le chromosome 2 et la moitié d'entre eux environ seraient assigné au noyau. L'estimation de la couverture mitochondriale serait alors complètement faussée.

3.4 Réduction des jeux de données

En une fois, on ne séquence par shotgun qu'une fraction aléatoire du génome. Pour être sur que nos reads couvrent 90% et plus du génome, il est nécessaire d'échantilloner le génome un très grand nombre de fois. Certaines régions se retrouveront dans de nombreux échantillons et auront donc une forte couverture. D'autres se retrouveront dans peu d'échantillons et seront peu couvertes. Une minorité de reads seront donc cruciaux et une majorité de reads complètement redondants.

Cette grande quantité de reads inutiles va poser des problèmes à l'assemblage. Le temps et la mémoire nécessaire pour les traiter sont non négligeables. De plus, ces reads contiennent des erreurs de séquençage. Si on les éliminait, la qualité de l'assemblage serait améliorée, particulièrement pour les assembleurs utilisant des graphes de Bruijn, que les noeuds et les arrêtes erronés peuvent rapidement envahir quand les jeux de données sont trop gros.

On va tester deux approches pour réduire le jeu de données.

Premièrement, le ré-échantillonage. Il consiste simplement à tirer aléatoirement un échantillon dans la librairie. On choisit la taille de l'échantillon en fonction des couvertures mitochondrial et chloroplastique qu'on souhaite obtenir.

Par exemple, pour Cvi-0, les couvertures des génomes mitochondrial et chloroplastique ont été estimées par alignement à 60X et 560X pour le jeu de donnée brut. En ne conservant en moyenne qu'un read sur trois, nous devrions obtenir une couverture de 20X sur la mitochondrie et de 190X sur le chloroplaste.

Nous ne connaissons pas *à priori* quelles couvertures des génomes mt et ct vous nous donner les meilleurs résultats. Nous sommes donc obligés de tester plusieurs valeurs, en gardant toutefois en tête que des couvertures très élevées (supérieures à 1000X) ont peu de chances de founir de bons assemblages.

Deuxièmement, la normalisation digitale. Cette procédure utilise la distribution de l'abondance des k-mers pour estimer la couverture par read et éliminer les reads dont la couverture dépasse la limite imposée.[2] Elle présente l'avantage, outre de réduire la taille du jeu de données, d'homogéiniser les couvertures des deux génomes.

On va normaliser les données avec khmer. Ce programme propose également un outil de filtration des données. Il permet de supprimer les reads contenant des k-mers d'abondance inférieure à la limite choisie. Ces k-mers de faible abondance correspondent à des erreurs de séquençage et, dans notre cas, à de la contamination nucléaire. On va donc appliquer une filtration après normalisation pour essayer de supprimer tous les reads correspondant au noyau.

Les seuils de normalisation et de filtration ne sont pas connus au préalable. On est obligé de procéder par tâtonnements en testant plusieurs valeurs des paramètres. On vérifie que les jeux de données obtenus sont valides, c'est-à-dire que la couverture a bien été réduite et homogénéisée et que les données supprimées étaient bien redondantes.

Nous avons pour chaque accession plusieurs jeux de donnée réduits selon différentes méthodes et différents paramètres. Nous devons les assembler afin de comparer la qualité des résultats obtenus pour chaque jeu de donnée.

3.5 Assemblage

3.5.1 Choix d'une stratégie d'assemblage

Le premier problème auquel nous allons être confrontés est que nous n'assemblons pas un seul génome mais deux simultanément. Deux solutions s'offrent à nous :

- utiliser un assembleur classique.

  Cette option n'est valable que les jeux de données pour lesquels les couvertures des génomes mitochondrial et chloroplastique sont les mêmes (jeux de données normalisés).

- utiliser un assembleur de métagénome.

  L'assembleur métagénomique va utiliser les différences de couverture pour distinguer 1) la mitochondrie du chloroplaste 2) les régions répétées des régions non répétées. On ne peut donc pas utiliser de jeux de données normalisé, pour lesquels la couverture a été rabotée en dessous d'une certaine limite. Pour cette approche, on ne se servira que de jeu de donnée ré-échantillonné, où les différentes couvertures ont été réduites tout en maintenant leur stoechiométrie.

Pour l'assemblage classique, nous avons utilisé Velvet et SGA. En effet, ces deux assembleurs sont basés sur des principes différents : les graphes de Bruijn pour Velvet et l'Overlap Consensus Layout pour SGA. SGA identifie toutes les paires de reads qui se recouvrent et construit un graphe avec un noeud pour chaque read et une arrête pour chaque paire de reads qui se recouvre. Le graphe est ensuite simplifiée (on retire les arrêtes redondantes) et stocké efficacement grâce à la transformée de Burrows-Wheeler. [6]

Notre assembleur de métagénome sera MetaVelvet.

A noter que Velvet et MetaVelvet n'ont assuré que la partie contiguage de l'assemblage. La partie scaffolding a été effectuée par Bambus2. Pour SGA, nous avons utilisé son scaffolder intégré (sga scaffold).

3.5.2 Choix des paramètres pour Velvet et MetaVelvet

L'un des paramètres les plus importants pour un assemblage par graphe de Bruijn est la taille de k-mer utilisé pour construire le graphe.

Nous avons construit notre propre script pour lancer Velvet avec de multiples valeurs de k. Les deux options les plus importantes, outre la taille du k-mer, sont *exp_cov* et *cov_cutoff*.

exp_cov permet d'indiquer à Velvet la couverture attendue pour une région unique du génome. On aligne le jeu de données normalisé à la référence et on estime à nouveau les couvertures du chloroplaste et de la mitochondrie. On doit obtenir une valeur unique qu'on fournit à *exp_cov*.

Les noeuds dont la couverture sont inférieure à *cov_cutoff* sont retirés du graphe de Bruijn après sa construction. Cette option permet donc d'éliminer toute trace de contamination nucléaire qui aurait échappé au filtrage.

Pour MetaVelvet, nous avons procédé exactement de la même façon. La seule différence est que l'option *exp_cov* est remplacée par *exp_covs*. On fournit à MetaVelvet les deux pics de couvertures qui correspondent à la mitochondrie et au chloroplaste.

La qualité des assemblages réalisés avec différents k est ensuite comparée à l'aide de Quast. Quast est un script Python qui va calculer les statistiques de chaque assemblage (N50, taille du plus grand contig etc.) et les présenter sous forme d'un rapport HTML pour une comparaison facile. On peut donc rapidement repérer la taille de k-mer qui semble produire le meilleur assemblage.

3.5.3 Choix des paramètres pour Bambus2

Bambus2 est une suite de scripts qui effectuent les différentes étapes du scaffolding.

On commence par créer des liens entre les contigs (clk) puis on rassemble ces liens entre une collection de bord possibles pour les contigs (Bundler). Bambus2 nous permet également de repérer les régions répétées du génome (MarkRepeats). On utilise ces informations pour placer les contigs les uns par rapport aux autres et les orienter (OrientContigs). Les scaffolds obtenus sous forme de graphe sont ensuite linéarisés et enregistrés au format fasta.

Les deux étapes qui proposent des options intéressantes sont MarkRepeats et OrientContigs.

3.5.4 Choix des paramètres pour SGA

SGA propose un ensemble de programmes qui filtrent et trimment les reads, trouvent les recouvrements entre les reads, construisent le String Graph et créent les contigs. Pour effectuer ces étapes, on s'est fortement inspiré d'un des scripts bash proposés par les auteurs de SGA pour assembler le génome de *C. Elegans*.

Pour le scaffolding, on réaligne le jeu de données sur les contigs avec *sga align* et on utilise le fichier BAM obtenu 1) pour avoir une liste des liens entre les contigs à l'aide du script Perl *sga-bam2de* 2) pour calculer pour chaque contig la statistique de Myers avec le script Python *sga-astat*.

Cette statistique sera utilisé par *sga scaffold* pour déterminer quels contigs correspondent à des répétitions : tous les contigs qui ont une statistique de Myers inférieure à un certain seuil sont considérés comme étant répétés. Pour déterminer le bon seuil, on compare les valeurs de la statistique de Myers pour les contigs chloroplastiques uniques et répétées (voir la structure du chloroplaste figure 3). On passe également à *sga scaffold* les options *–strict –remove-conflicting* pour avoir les scaffolds les plus conservatifs possibles.

sga scaffold donne pour chaque scaffold, la liste des contigs qui le constituent et comment ils sont placés les uns par rapport aux autres. *sga scaffold2fasta* utilise ce fichier pour écrire la séquence des scaffolds au format fasta. On se place dans le mode résolution par graphe ; avec l'option *-g best-any*, on tente de résoudre les gaps entre les contigs en parcourant le graphe et en trouvant le chemin qui correspond le mieux aux distances estimées entre les contigs.

3.5.5 Evaluation de la qualité des assemblages obtenus

Comment savoir si nos assemblages sont bons ou pas ?

Pour le chloroplaste, on sait exactement ce que l'on doit obtenir. Le chloroplaste évolue non par réarrangements mais par mutations ponctuelles. Les chloroplastes des 10 accessions naturelles vont donc être très proches du

chloroplaste de référence.

Nous avons donc commencé par étudier la structure du chloroplaste de référence.

On aligne le chloroplaste de référence contre lui-même à l'aide de Nucmer. Nucmer est une pipeline de la suite MUMmer qui permet d'aligner extrêmement rapidement des génomes entiers contre une référence et de représenter ces alignements sous forme de graphe de ressemblance (dotplots).

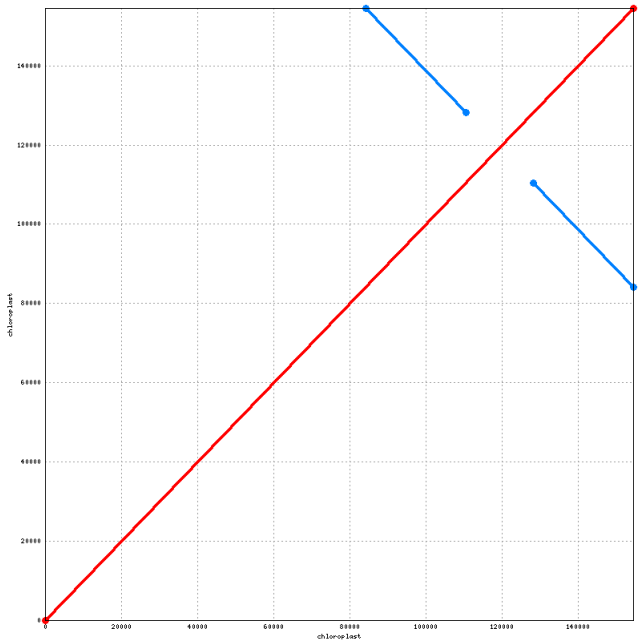


FIGURE 2: Dotplot du chloroplaste de référence aligné contre lui-même

La figure 2 met en évidence qu'une portion du chloroplaste de référence comprise entre 85 et 110kb est répétée et inversée entre 128 et 154kb. On devrait retrouver la même inversion-répétition chez les chloroplastes des dix accessions naturelles.

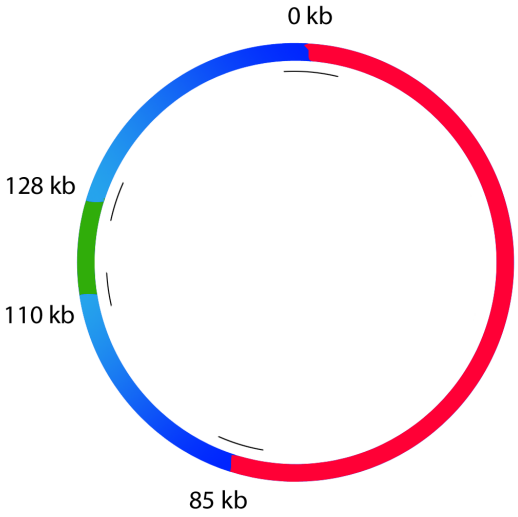


FIGURE 3: Structure du génome chloroplastique chez *Arabidopsis Thaliana*, toutes accessions confondues

Après contigage, on aligne nos contigs contre le chloroplaste de référence à l'aide de Nucmer. On doit obtenir :

- un contig non chimérique qui mappe sur le chloroplaste de référence entre 0 et 85kb dans le sens direct (en rouge sur la figure 3)
- un contig non chimérique qui mappe sur le chloroplaste de référence entre 85 et 110kb dans le sens direct et entre 128 et 154kb dans le sens inverse (en bleu sur la figure 3)
- un contig non chimérique qui mappe sur le chloroplaste de référence entre 110 et 128kb dans le sens direct (en vert sur la figure 3)
- les jonctions entre les 3 contigs (arcs de cercle noirs sur la figure 3)

Le contiguer ne sait pas que la région représentée en bleue est une région répétée. Il voit seulement qu'à ses extrémités existe plusieurs possibilités. Par exemple, l'extrémité bleu foncé de la région répétée est reliée aux deux extremités de la région unique rouge. Le contiguer ne doit pas faire un choix et fusionner les contigs bleu et rouge dans un sens ou dans l'autre. Il doit les garder séparés.

Le scaffolder, en revanche, doit pouvoir détecter les régions répétées et résoudre les répétitions.

Après scaffolding, on doit donc avoir un unique scaffold non chimérique qui recouvre la totalité du chloroplaste de référence et présente la même inversion-répétition.

En alignant nos scaffolds contre le chloroplaste de référence avec Nucmer, on doit retrouver la figure 2.

Pour la mitochondrie, on ne sait pas précisément ce qu'on doit obtenir. En raison des nombreux réarrangements qu'elles ont subis, les mitochondries des 10 accessions naturelles vont être très différentes. [1] Si on retrouve le même contig mitochondrial chez 9 accessions sur 10, cela ne signifie pas que l'assemblage de la dixième accession est mauvais. On peut néanmoins effectuer plusieurs tests afin de savoir si notre assemblage mitochondrial est plausible ou pas.

1. Avec Nucmer, on aligne les contigs ou les scaffolds contre la mitochondrie de référence afin de savoir si notre assemblage couvre la totalité de la mitochondrie.

2. On aligne les données brutes sur les contigs ou les scaffolds et on regarde si les contigs mitochondriaux et chloroplastique ont la couverture attendue (consistence interne).

3. On regarde si des paires s'alignent sur le même contig avec des tailles d'insert incohérentes. S'il y en a beaucoup, cela signifie que la structure interne du contig est mauvaise. On peut ainsi repérer et quantifier les mauvais contigs.

   Pour compter le nombre de "mauvaises" paires dans chaque contig, on utilise la librairie R ContigLink.

ContigLink prend en entrée l'alignement au format bam des données brutes sur les contigs. Il sépare les pairs qui s'alignent sur un seul et même contig et les pairs qui s'alignent sur deux contigs différents. Ces deux types de pairs vont être utilisés pour calculer deux estimations de la taille d'insert et de son écart-type. ContigLink donne ensuite une estimation de la qualité de l'assemblage. Cette estimation contient :

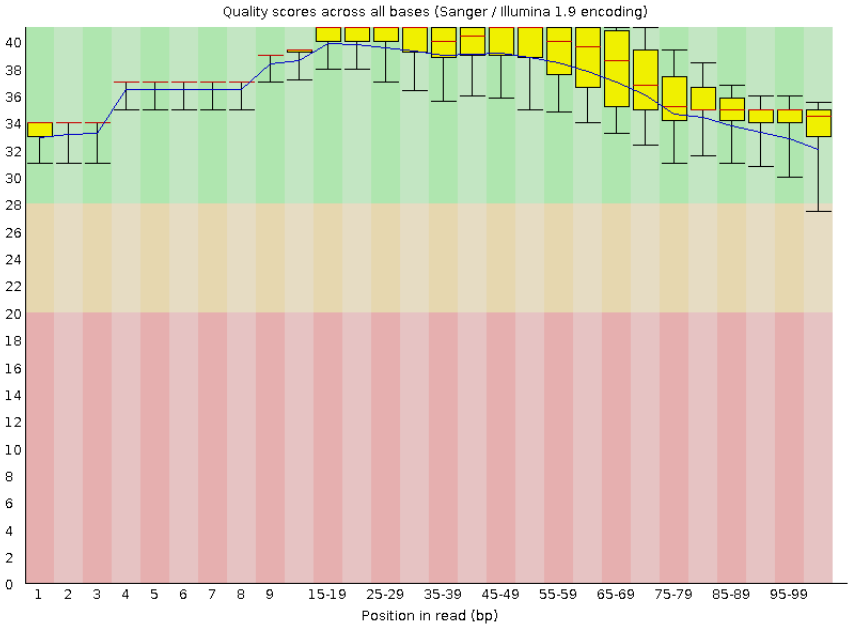
- pour chaque contig, le nombre de pairs s'alignant sur ce contig avec une bonne taille d'insert et une mauvaise taille d'insert (éloignée de son estimation par plus de deux écarts-type)

- pour chaque pair de contigs, le nombre de "bonnes" et mauvaises" paires supportant un lien entre ces contigs.

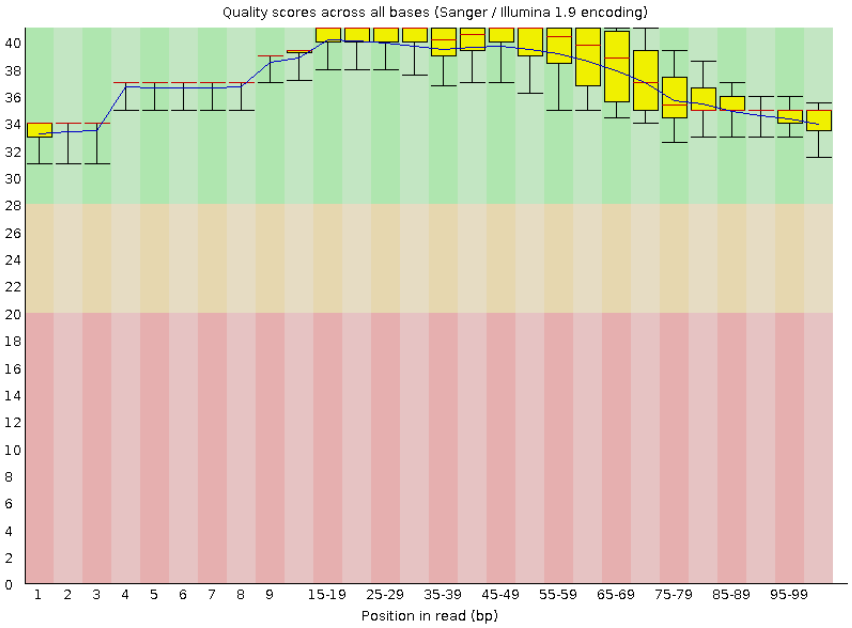
Cette dernière information est très utile pour comprendre comment Bambus2 et sga scaffold vont relier les contigs entre eux pour former des scaffolds.

4 Résultats

4.1 Préparation des séquences



(A) Avant trimming



(B) Après trimming

FIGURE 4: Qualité des bases en fonction de leur position sur les reads : exemple de Shahdara

La qualité de nos reads était excellente à la base : elle n'est donc que légèrement améliorée par le trimming.

La quasi-totalité des séquences ont conservé après trimming une taille proche de leur taille initiale (autour de 100 bases pour Shahdara et Kz-9 et de 150 bases pour les autres accessions).

Le taux de duplication de nos libraries est considéré comme trop élevé par fastqc. Cependant, comme notre librarie contient contient des données mitochondriales, où les séquences répétées sont fréquentes, nous n'avons pas à nous alarmer de ce résultat.

4.2 Alignement

	Chloroplaste	Mitochondrie	Noyau	Non aligné
Blh-1	8.08	64.26	15.26	12.40
Bur-0	16.16	9.78	42.79	31.26
Ct-1	16.09	6.32	45.20	32.39
Cvi-0	22.86	5.73	40.02	31.39
Ita-0	14.70	43.72	21.48	20.09
Jea	12.83	21.90	38.84	26.43
Kz-9	15.86	21.02	44.15	18.98
Mr-0	8.04	66.05	13.07	12.84
Oy-0	15.86	11.04	42.97	30.12
Sha	19.35	33.42	32.85	14.37

TABLE 2: Pourcentage de reads mappés et non mappés pour les 10 accessions (référence sans chromosome 2)

Nous observons d'abord un pourcentage non négligeable de reads non mappés - jusqu'à 32% pour l'accession Ct-1.

Certains reads non mappés appartiennent au chromosome 2 hors insertions mitochondriales. Comme le chromosome 2 est entièrement masqué, ils ne peuvent s'aligner nulle part.

D'autres correspondent à des régions où la variabilité entre l'accession étudiée et la référence dépasse le seuil fixé. On rappelle qu'avec les paramètres qu'on a fixé, pour qu'un read s'aligne à la référence, ils doivent être séparés par moins de deux événements indépendants. Or, on a trouvé dans certains reads non mappés jusqu'à 7 SNPs de différence par rapport à la référence.

Enfin, les autres reads non mappés ont une faible complexité et correspondent probablement à des régions répétées.

A partir du nombre de reads mappés sur chaque génome, on peut avoir une estimation de la couverture par une formule très simple :

couverture $\approx$ nombre de reads mappés X taille des reads / taille du génome

	Chloroplaste	Mitochondrie	Noyau
Kz-9	2845.83	1587.55	10.60
Sha	2764.37	2010.17	6.28
Blh-1	218.26	730.76	0.55
Bur-0	466.26	118.79	1.65
Ct-1	348.44	57.65	1.31
Cvi-0	558.13	58.94	1.31
Ita-0	396.71	496.72	0.78
Jea	326.96	235.05	1.33
Mr-0	237.68	822.04	0.52
Oy-0	458.24	134.26	1.66

TABLE 3: Les couvertures par accessions (référence sans chromosome 2)

Pour toutes les accessions, la couverture du noyau est faible devant celles de la mitochondrie et du chloroplaste. Cependant, le génome nucléaire d'*Arabidopsis Thaliana* est 300 fois plus grand que le génome mitochondrial et 1000 fois plus grand que le génome chloroplastique. En nombre de bases, la contamination nucléaire est donc non négligeable et nous devrons nous efforcer de l'éliminer du jeu de données avant l'assemblage.

Les couvertures mitochondriale et chloroplastique ne sont pas homogènes pour une même accession. Cela va nous poser problème car la taille de k-mer optimale pour l'assemblage par graphe de Bruijn dépend de la couverture. On n'aura donc pas le même k-mer optimal pour les génomes mitochondrial et chloroplastique. Nous prendrons en compte cette disparité de couverture pour notre stratégie d'assemblage.

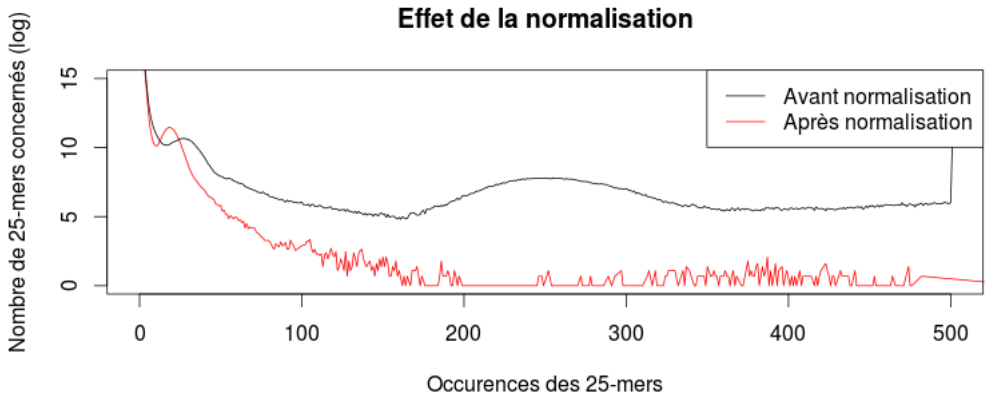
4.3 Réduction des jeux de données

Exemple pour Cvi-0 normalisé à 30X

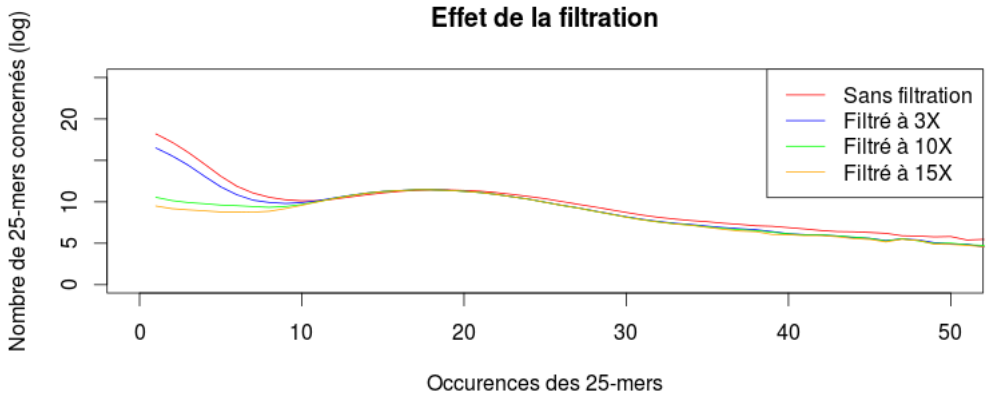
La figure 5a illustre l'effet de la normalisation sans filtration. On vérifie que la normalisation ramène bien le chloroplaste et la mitochondrie a un même niveau de couverture.

La figure 5b met en évidence l'effet de différents niveaux de filtration. Nous avions commencé avec un seuil de filtration de 3X afin d'éliminer le noyau, dont la couverture est de 1.3X environ pour Cvi-0. Cependant, après assemblage de notre jeu de données normalisé et filtré, la plupart des contigs obtenus étaient faiblement couverts et s'alignaient sur le noyau. Un seuil de filtration faible est donc insuffisant pour éliminer la majorité des reads du noyau.

Nous avons testé des valeurs de plus en plus grandes pour le seuil de filtration (jusqu'à 15X pour des données normalisées à 30X!). Néanmoins, comme le montre la figure 5b, de telles valeurs du seuil de normalisation ne dégradent pas la qualité du jeu de données.



(A) Avant normalisation, on constate 2 pics, situés autour de 20 et 250 sur l'axe des abscisses. Cela signifie qu'on trouve dans notre jeu de données un grand nombre de 25-mers qui apparaissent autour de 20 fois et de 150 fois. Ce sont probablement des 25-mers qui appartiennent à la mitochondrie et au chloroplaste respectivement. Après normalisation, nous n'avons plus qu'un seul pic à 20 sur l'axe des abscisses. Les 25-mers mitochondriaux et chloroplastiques apparaissent donc désormais en moyenne 20 fois dans le jeu de données normalisé. Les couvertures de la mitochondrie et du chloroplaste ont été homogéinisées.



(B) La filtration permet de diminuer le nombre de 25-mers de très faible abondance sans dégrader les données. En effet, le pic d'abscisse 20 constaté à la figure précédente conserve la même forme. Les 25-mers mitochondriaux et chloroplastiques sont donc épargnés par la filtration. Ils apparaissent toujours entre 10 et 30 fois dans le jeu de données.

FIGURE 5: Histogrammes des abondances des 25-mers pour Cvi-0. On représente le nombre y de 25-mers qui apparaissent x fois dans le jeu de données.

4.4 Assemblage

4.4.1 Contigage avec Velvet

Exemple pour Cvi-0 normalisé à 20X et filtré à 10X

On assemble pour des tailles de k-mer comprises entre 61 et 91, par pas de 10.

Statistics without reference	Cvi_norm_20X_filt_10X_k_61_co.	Cvi_norm_20X_filt_10X_k_71_co.	Cvi_norm_20X_filt_10X_k_81_co.	Cvi_norm_20X_filt_10X_k_91_co...
# contigs	44	42	38	33
Largest contig	80 710	128 342	128 361	128 381
Total length	497 844	498 186	498 664	495 060
N50	29 005	29 860	37 397	28 833

FIGURE 6: Les statistiques d'assemblages pour 4 valeurs de k (calculés pour les contigs de taille supérieure à 500 bases)

L'assemblage qui a le plus petit nombre de contigs et le plus grand N50 est l'assemblage réalisé avec k=81. La taille de k-mer optimale va donc se situer autour de 81. On assemble à nouveau le jeu de donnée pour des tailles de k-mer comprises entre 75 et 85, par pas de 2.

On trouve une taille de k-mer optimale de 79. L'assemblage a alors 38 contigs d'au moins 500 bases et un N50 de 39kb. Le contig le plus long a une taille de 149kb.

Avec Nucmer, on compare les contigs obtenus pour une taille de k-mer de 79 avec le chloroplaste de référence et la mitochondrie de référence. Le plus grand contig de l'assemblage est un contig chimérique. Il s'agit d'un contig chloroplastique qui contient trois grands morceaux d'ADN mitochondrial (de taille 13, 17 et 18kb). Chez la référence, il existe des insertions d'ADN mitochondrial dans le chloroplaste mais ces insertions sont de petite taille (inférieure à 1kb). Plusieurs grandes insertions d'ADN mitochondrial dans le chloroplaste ont plus de chance de correspondre à un mauvais assemblage qu'à une différence entre les chloroplaste des accessions.

Pour s'en assurer, on regarde l'assemblage pour k=83, le deuxième meilleur assemblage d'après ses statistiques. Son contig le plus long fait 128kb. Il n'est pas chimérique.

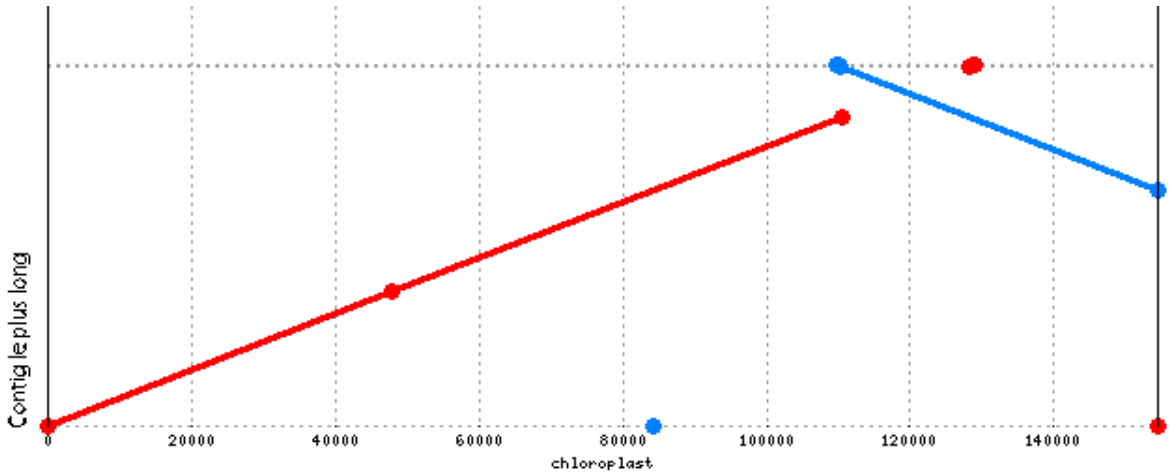


FIGURE 7: Graphe de ressemblance entre le plus grand contig obtenu pour k=83 (en ordonnée) et le chloroplaste de référence (en abscisse)
Nous n'avons pas représenté les autres contigs afin de ne pas surcharger la figure.

Sur la figure 7, on voit que le plus grand contig de l'assemblage s'aligne au chloroplaste de référence dans le sens direct de 0 à 110kb puis dans le sens inverse de 110 à 128kb. Il contient donc la longue région unique du chloroplaste,

la première copie de sa région inversée-répétée et la petite région unique du chloroplaste dans le sens inverse (en rouge, bleu et vert sur la figure 3). Les trois régions du chloroplaste, qui auraient dû rester distinctes, ont été fusionnées en un seul contig par Velvet. Cet assemblage n'est donc pas bon.

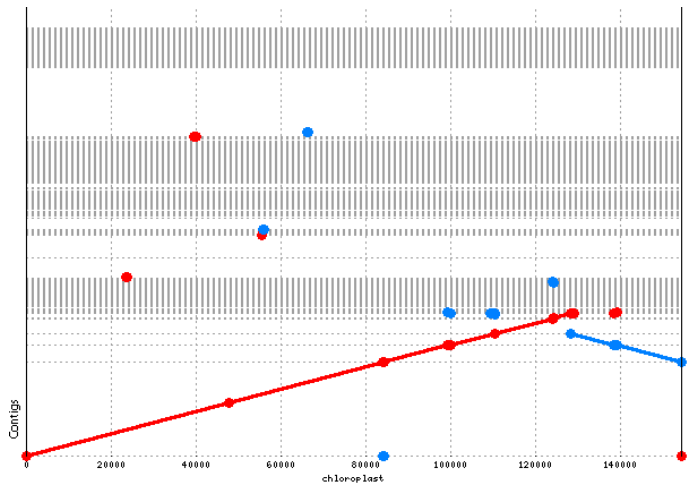
Nous avons testé différents niveaux de normalisation et de filtration mais nous n'avons jamais réussi à obtenir les 3 contigs chloroplastiques attendus.

4.4.2 Contigage avec MetaVelvet

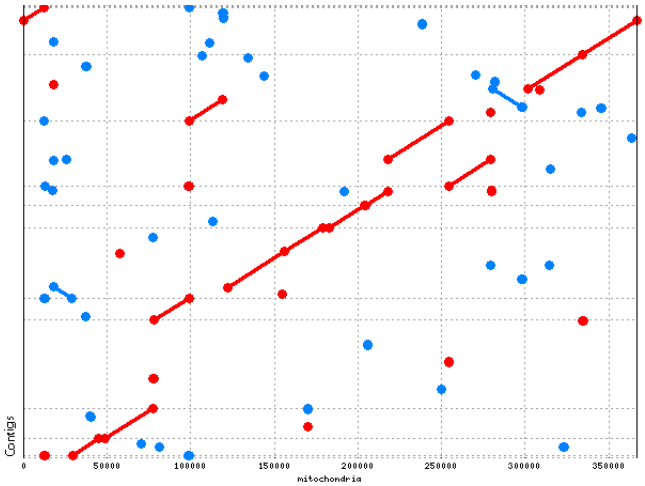
Exemple pour Cvi-0

Nous avons assemblés plusieurs jeux de données ré-échantillonés (qui ne contiennent que la moitié, le tiers, le quart des données etc.) mais les meilleurs résultats ont été obtenus pour les données brutes simplement filtrées à 10X afin de retirer la contamination nucléaire. Nous les présentons ici.

On effectue la recherche de la taille de k-mer optimale comme précédemment. On choisit k=71. Le N50 est alors de 44kb. Il y a 90 contigs d'au moins 500 bases et le contig le plus long fait 85kb.



(A) Graphe de ressemblance entre le chloroplaste de référence (en abscisse) et les contigs obtenus (en ordonnée)



(B) Graphe de ressemblance entre la mitochondrie de référence (en abscisse) et les contigs obtenus (en ordonnée)

FIGURE 8: Comparaison entre les contigs obtenus et la référence. Tous les contigs ne figurent pas.

En figure 8a, on voit qu'on retrouve dans notre assemblage la longue région unique, la région répétée et la petite région unique du chloroplaste. Ces trois régions correspondent à différents contigs et sont donc bien distinctes. On n'a cependant pas 3 contigs différents mais 5 (la petite région unique et la région répétée sont chacune séparée en deux contigs). La figure 8b nous montre qu'on a également dans notre assemblage des contigs qui recouvrent entièrement la mitochondrie de référence.

	Nom	Taille (bp)	Couverture (X)
Longue région unique	Noeud 1379	84093	554
Région répétée	Noeud 8	813	933
	Noeud 297	15209	1117
	Noeud 331	10265	906
Petite région unique	Noeud 34	3990	502
	Noeud 360	13798	594

(A) Les contigs chloroplastiques

Nom	Taille (bp)	Couverture (X)
Noeud 7	62590	60
Noeud 105	21238	62
Noeud 138	20691	66
Noeud 222	67121	56
Noeud 223	455	76
Noeud 227	15628	53
Noeud 296	44684	67
Noeud 514	18462	65
Noeud 541	62127	60
Noeud 3153	28796	61
Noeud 6544	335	66

(B) Les contigs mitochondriaux

FIGURE 9: Les contigs s'alignant sur le chloroplaste ou la mitochondrie de référence et possédant une couverture cohérente

On réaligne les données brutes sur les contigs. On voit que de nombreux contigs présent sur les graphes de ressemblances 8a et 8b possèdent bien la couverture attendue : environ 60X pour la mitochondrie et 560X pour le chloroplaste (le double pour sa région répétée). La liste de ces contigs est donnée dans les tables 9a et 9b.

Cependant, on trouve également de nombreux contigs qui ne passent pas l'étape de consistence interne. Il s'agit :

1. de contigs qui s'alignent sur le noyau de l'accession de référence d'après nucmer ou Blast mais sont extrêmement couverts par les reads bruts

   D'après des recherches dans Blast, ces contigs correspondent à des unités répétées d'ADN ribosomique, d'où leur forte couverture.

2. de contigs qui s'alignent sur la mitochondrie ou le chloroplaste de référence d'après nucmer ou Blast mais dont la couverture n'est pas celle qui est attendue

   Trois contigs apparaissent plusieurs fois dans le génome mitochondrial de C24 d'après Blast. Ces contigs peuvent également correspondre à des répétitions chez Cvi-0.

   A l'opposé, on a également trois petits contigs qui s'alignent sur la mitochondrie de référence mais dont la couverture est pratiquement nulle. Ces petits contigs, présents chez la mitochondrie de C24, ont peut-être été supprimés du génome mitochondrial de Cvi-0.

Nom	Taille (bp)	Couverture (X)
Noeud 1122	2089	94
Noeud 1668	2163	108
Noeud 2098	508	90

(A) Des contigs possiblement répetées

Nom	Taille (bp)	Couverture (X)
Noeud 1243	251	0.6
Noeud 4426	230	0
Noeud 4766	200	0.75

(B) Des séquences appartenant à la mitochondrie de C24 mais peut-être pas à celle de Cvi-0

FIGURE 10: Des possibles répétitions et délétions dans le génome mitochondrial de Cvi-0

Il existe également des séquences chloroplastiques dont la couverture est trop faible mais on ignore encore pourquoi.

Nous avons essayé d'utiliser ContigLink pour répérer les contigs mal formés mais sans succès. En effet, si on considère un contig qu'on sait être bien formé, par exemple le noeud 1379 correspondant à la longue région unique du chloroplaste (voir table 9a), on voit que 3629 paires s'alignent sur ce contig avec une bonne taille d'insert et 72692 avec une mauvaise taille d'insert, ce qui représente un ration "mauvaises paires"/"bonnes paires" de 0.05.

Or, si on étudie maintenant un contig qu'on sait être mal formé, par exemple l'unique contig chloroplastique trouvé par Velvet à la section précédente, on voit que ce mauvais contig est supporté par 7413 "mauvaises paires" pour 132255 "bonnes paires" d'où un ratio de 0.06. On ne peut donc pas utiliser le nombre de paires s'alignant sur un contig avec une taille d'insert incohérente pour distinguer les bons et les mauvais contigs.

En revanche, si on visualise l'alignement des données brutes sur ce contig mal formé à l'aide d'IGV, on voit bien qu'on a, à 110 et 128kb, des colonnes de reads dont le compagnon mappe sur le même contig avec une taille d'insert incohérente. Ce qui compte n'est donc pas pas la proportion des "mauvaises" paires par rapport aux "bonnes" mais la localisation des reads de ces "mauvaises" paires. S'ils sont concentrés en un point précis, le contig doit être cassé à cette endroit ; il est donc mal formé. Malheureusement, on ne peut pas visualiser sous IGV tous les contigs.

4.4.3 Scaffolding avec Bambus2

4.4.4 Assemblage avec SGA

5 Discussion

5.1 Comparaison entre les assembleurs

Nous avons essayé de comprendre pourquoi Velvet ne nous donne jamais les trois régions distinctes du chloroplaste pour les données normalisées.

Nous avons remarqué que, même lorsqu'on demande à Velvet ou MetaVelvet de ne pas scaffolder (avec l'option *-scaffolding no*), Velvet/MetaVelvet lance tout de même Pebble, son module de scaffolding. Pebble connecte les contigs uniques en utilisant les informations apportées par les reads pairés.

Comme dans la section 4.4.1, on utilise Velvet pour assembler les données de Cvi-0 normalisées à 20X et filtrées à 10X pour k=83 mais, cette fois, en désactivant complètement Pebble (on dit à Velvet que nos données pairées ne le sont pas).

On obtient un assemblage où les trois régions du chloroplaste sont parfaitement distinctes et représentées chacune par un contig. Cependant, les statistiques de l'assemblage se sont considérablement dégradées. Le N50 est passé de 37kb à 16kb. Le nombre de contigs d'au moins 500 bases est passé de 35 à 55.

On en déduit que, même lorsqu'on lui demande de ne pas scaffolder, Velvet exploite les informations sur les paires si elles sont présentes en utilisant Pebble pour fusionner certaines contigs. Ce premier round de Pebble donne globalement un meilleur assemblage mais il arrive que Pebble fasse des erreurs comme c'est le cas ici pour le chloroplaste.

Lorsqu'on assemble les données brutes avec MetaVelvet, le premier round de Pebble est également lancé sur chacun des sous-graphes de MetaVelvet. Pourtant, comme nous l'avons vu à la section 4.4.2, les trois régions du chloroplaste sont distinctes. Donc, Pebble commet l'erreur de relier les trois régions du chloroplaste uniquement pour les données normalisées mais pas pour les données brutes.

Comme on l'a dit précédemment, Pebble connecte uniquement des noeuds **uniques**. Or, pour les données normalisées, il fusionne le contig qui correspond à la région répétée du chloroplaste avec les contigs correspondant aux deux régions uniques. Cela signifie que le contig de la région répétée est considéré par Velvet comme étant unique.

Pour distinguer les noeuds uniques des noeuds répétés, Velvet, comme Bambus2 et SGA, utilise la statistique de Myers.

5.2 Un scaffolder sachant scaffolder ne scaffolde jamais sans ses liens

5.3 Comparaison entre les accessions

6 Conclusion

References

[1] Maria P Arrieta-Montiel, Vikas Shedge, Jaime Davila, Alan C Christensen, and Sally A Mackenzie. Diversity of the arabidopsis mitochondrial genome occurs via nuclear-controlled recombination activity. *Genetics*, 183(4):1261–1268, 2009.

[2] C Titus Brown, Adina Howe, Qingpeng Zhang, Alexis B Pyrkosz, and Timothy H Brom. A reference-free algorithm for computational normalization of shotgun sequencing data. *arXiv preprint arXiv:1203.4802*, 2012.

[3] Françoise Budar and Sota Fujii. *Mitochondrial Genome Evolution*, volume 63. Academic Press, 2012.

[4] Heather I McKhann, Christine Camilleri, Aurélie Bérard, Thomas Bataillon, Jacques L David, Xavier Reboud, Valérie Le Corre, Christophe Caloustian, Ivo G Gut, and Dominique Brunel. Nested core collections maximizing genetic diversity in arabidopsis thaliana. *The Plant Journal*, 38(1):193–202, 2004.

[5] Michaël Moison, Fabrice Roux, Martine Quadrado, Romain Duval, Muriel Ekovich, Duc-Hoa Lê, Marie Verzaux, and Françoise Budar. Cytoplasmic phylogeny and evidence of cyto-nuclear co-adaptation in arabidopsis thaliana. *The Plant Journal*, 63(5):728–738, 2010.

[6] Jared T Simpson and Richard Durbin. Efficient de novo assembly of large genomes using compressed data structures. *Genome research*, 22(3):549–556, 2012.

[7] Steve Rounsley Terrance P. Shea Maria-Ines Benito Christopher D. Town Claire Y. Fujii Tanya Mason Cheryl L. Bowman Mary Barnstead Xiaoying Lin, Samir Kaul. Sequence and analysis of chromosome 2 of the plant arabidopsis thaliana. *Nature*, (6763):761–768, 1999.