



UNIVERSITÉ DE ROUEN, NORMANDIE UNIVERSITÉ

UFR DES SCIENCES ET TECHNIQUES - CENTRE DE FORMATION PAR APPRENTISSAGE

MASTER 2 DE BIOINFORMATIQUE

2015-2016 - MI-PARCOURS

FRANÇOIS-XAVIER BABIN

**AMÉLIORATION D'UNE STRATÉGIE DE CORRECTION DES LECTURES
NANOPORE.**



CENTRE NATIONAL DE SÉQUENÇAGE (GÉNOSCOPE)

COMMISSARIAT À L'ÉNERGIE ATOMIQUE ET AUX ÉNERGIES ALTERNATIVES (CEA)

Sous la responsabilité de Monsieur JEAN-MARC AURY



CENTRE UNIVERSITAIRE DE
FORMATION CONTINUE
ET PAR ALTERNANCE





UNIVERSITÉ DE ROUEN, NORMANDIE UNIVERSITÉ

UFR DES SCIENCES ET TECHNIQUES - CENTRE DE FORMATION PAR APPRENTISSAGE

MASTER 2 DE BIOINFORMATIQUE

2015-2016 - MI-PARCOURS

FRANÇOIS-XAVIER BABIN

**AMÉLIORATION D'UNE STRATÉGIE DE CORRECTION DES LECTURES
NANOPORE.**



CENTRE NATIONAL DE SÉQUENÇAGE (GENOSCOPE)

COMMISSARIAT À L'ÉNERGIE ATOMIQUE ET AUX ÉNERGIES ALTERNATIVES (CEA)

Sous la responsabilité de Monsieur JEAN-MARC AURY



CENTRE UNIVERSITAIRE DE
FORMATION CONTINUE
ET PAR ALTERNANCE



REMERCIEMENTS :

Je tiens tout d'abord à remercier toute l'équipe R&D en BioInformatique du Génoscope pour leur accueil chaleureux.

Je souhaite remercier plus particulièrement Jean-Marc Aury et Benjamin Istace pour le temps et l'aide précieuse qu'ils m'ont apporté durant cette première année d'alternance.

Sommaire

1	Introduction	1
1.1	Le Laboratoire d’Informatique Scientifique	1
1.2	Contexte Scientifique et Technologique	1
1.3	Objectifs de mon travail d’Alternance	5
2	Environnement Informatique, outils informatiques et nature des données	6
2.1	Les données tests	6
2.1.1	Librairie et Séquençage Illumina	6
2.1.2	Librairie et Séquençage Nanopore	6
2.1.3	Constitution du jeu de données test	6
2.2	Environnement Informatique	7
2.2.1	Cluster de Calcul	7
2.2.2	Poste Personnel	7
2.3	Méthodologie de travail	7
2.3.1	Veille Bibliographique	8
2.3.2	Langages et commandes utilisées	8
2.4	Description et fonctionnement de NaS	8
2.5	La Stratégie KMC2 - Cookiecutter	12
2.6	La Stratégie Minimap	13
3	Résultats	15
3.1	Architecture logicielle	15
3.2	Visualisation pour la comparaison de données	15
3.3	Implémentation de KMC2 - Cookiecutter	16
3.4	Implémentation parallélisée de Minimap	18
3.5	Implémentation globale de Minimap	19
4	Discussions	23
4.1	Évolution de la technologie Nanopore	23
4.2	Veille technologique	24
5	Conclusions et Perspectives	26

Abréviations :

CEA : Commissariat à l'Énergie Atomique et aux Énergies Alternatives

CPU : Central Processing Unit (un processeur)

NGS : Next Generation Sequencing

ADN : Acide DésoxyriboNucléique

CRT : Cyclic Reversible Termination

NaS : Nanopore Synthetic (lecture nanopore synthétique)

OLC : Overlap Layout Consensus

SSH : Secure Shell

Glossaire :

Base Calling : conversion d'un signal (électrique ou chimique) en une base

Benchmarking : comparaison des performances entre des programmes (temps CPU , utilisation de mémoire ...)

Wrapper : programme chargé d'exécuter d'autres programmes les uns à la suite des autres

Parallélisation : exécution simultanée de plusieurs processus identiques (souvent par l'utilisation de plusieurs processeurs)

Multithreading : utilisation simultanée de plusieurs processeurs pour un même traitement

Indel : insertions délétions

Scaffoldeur : outil permettant d'ordonner et d'orienter des contigs

Gapcloseur : outil permettant de combler les trous lors de l'assemblage

Flowcell : support physique pour le séquençage

Contig : séquence résultant de l'assemblage de lectures

Scaffold : séquence résultant de l'ordonnancement de contigs

1 Introduction

1.1 Le Laboratoire d’Informatique Scientifique

Le Laboratoire d’Informatique Scientifique (LIS) est l’un des laboratoires du Genoscope. Véritable plate-forme de traitement des données de séquençage, son activité s’oriente autour de l’exploitation et l’analyse bio-informatique des données issues des projets annuels (projets France Génomique). Ce laboratoire est subdivisé en trois équipes : Système et réseaux, Développement, Recherche et Développement en BioInformatique et Séquençage (R&D BioSeq).

Encadrée par Jean-Marc Aury, l’équipe R&D BioSeq assure le contrôle de la qualité des séquences, l’assemblage et l’annotation. Les personnes affectées à la thématique assemblage calibrent les outils d’assemblage (assembleur, scaffoldeur et gapcloseur) en fonction des caractéristiques génomiques de l’espèce étudiée et effectuent une veille technologique (benchmarking et application de nouveaux outils d’assemblage). Avec l’arrivée d’une nouvelle technologie de séquençage, le minION (commercialisé par la compagnie Oxford Nanopore Technologies), un nouveau travail de correction des lectures Nanopore est venu s’ajouter à ceux déjà réalisés par l’équipe. C’est dans le cadre de ce travail que je réalise mon alternance au sein du Genoscope.

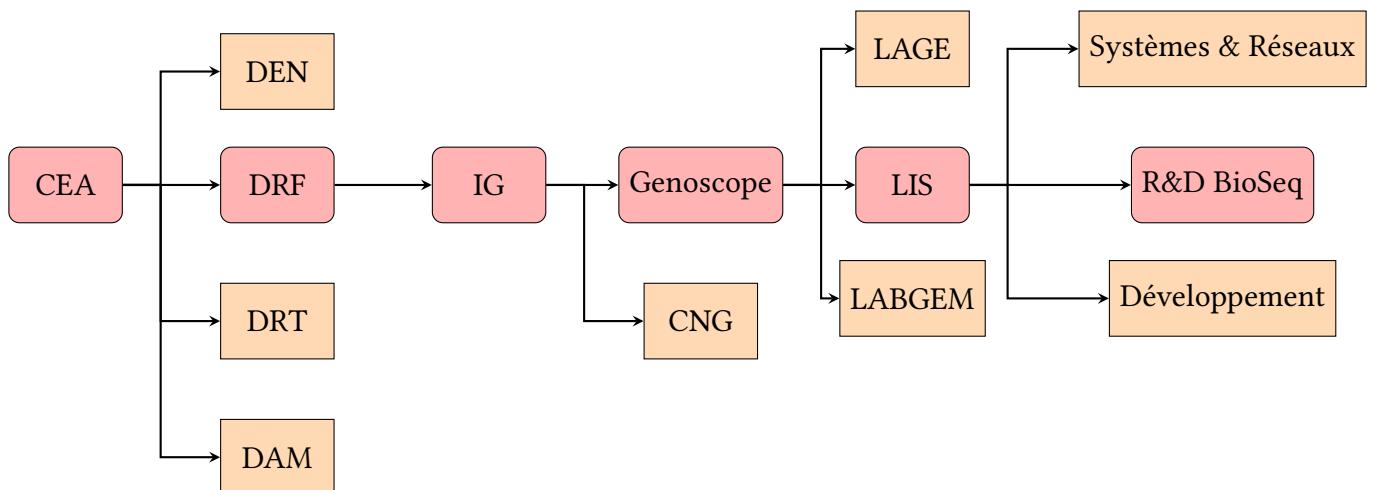


FIGURE 1: Organigramme situant l’équipe R&D BioInformatique et Séquençage au sein du CEA

1.2 Contexte Scientifique et Technologique

Le séquençage de l’ADN est apparu dans les années 1970 avec l’apparition de la chimie de Sanger [1] (synthèse enzymatique). Cette chimie est à l’origine des séquenceurs automatiques utilisés lors du projet génome humain. Malgré des séquences obtenues de très bonne qualité, une

limite de débit et de coût de séquençage a encouragé le développement d'une nouvelle génération de séquenceurs (NGS).

A partir de 2005, de nouvelles technologies (NGS) ont fait leur apparition. Elles permettent de séquencer l'ADN à moindre coût, dans un temps plus court et avec un débit élevé. Ces technologies sont basées sur une fragmentation préalable de l'ADN en fragments courts, d'une longueur de 100 à 800 paires de bases (pb).

Dans le cas de la technologie Illumina, majoritairement utilisée aujourd'hui, les fragments d'ADN vont ensuite être reliés à des adaptateurs, fixées à un support (flowcell), amplifiées et séquencées. Pour obtenir la séquence lors de l'amplification, l'ajout de chaque base va générer une phosphorescence spécifique à la base ajoutée et cette lumière émise va être détectée par le séquenceur pour reconstituer la séquence du fragment d'ADN. Le séquençage Illumina permet d'obtenir des lectures courtes (100 pb) de très bonne qualité (avec moins de 0,1% d'erreurs [2]).

Ces lectures reconstituées peuvent ensuite être assemblées (regroupées et ordonnées selon leur similarité de séquences) pour reconstruire la (ou les) séquence(s) commune(s) à un ensemble des lectures. Cette séquence commune porte le nom de contig (figure 2 a). Lorsque plusieurs contigs sont assignés à une région génomique (exemple : un chromosome), ces derniers peuvent être assemblés ensemble pour reconstituer cette région. Pour cela, une première étape de scaffolding (figure 2 b) peut permettre de relier (ordonner et orienter) deux contigs à condition qu'une lecture (dont les séquences aux extrémités sont plus distantes, librairie Mate Pair) couvre les deux contigs. Lorsque des scaffolds se retrouvent avec des zones sans assignations de bases (trous), une dernière étape, le gapclosing (figure 2 c), permet de combler ces trous (en partie ou complètement).

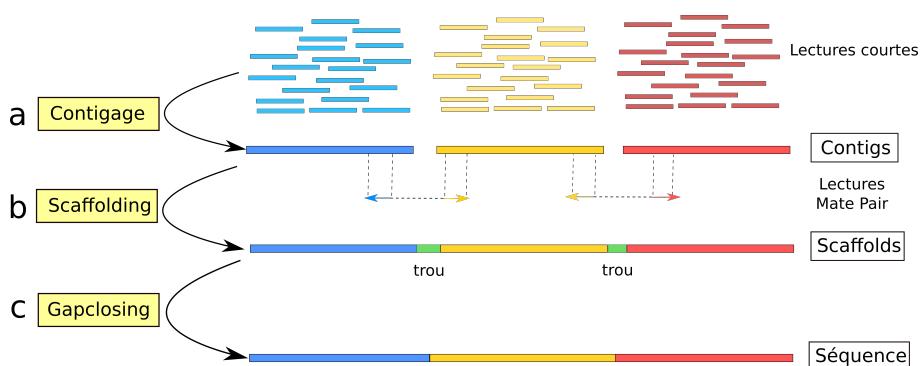


FIGURE 2: Schéma des étapes d'assemblage : le Contigage (a), le Scaffolding (b) et le Gapclosing (c)

Bien que le séquençage courtes lectures (exemple : Illumina) ait permis l'obtention de nombreux génomes de référence, une difficulté dans l'assemblage de certaines séquences persiste du fait de leur complexité génomique. En effet, les régions répétées constituent une véritable difficulté lors de l'assemblage. Ces dernières ont tendance à être regroupées à cause de leur forte similarité et à être co-assemblées en une seule séquence et ainsi fragmenter l'assemblage (figure 3). La séquence obtenue est donc incomplète et morcelée. Une autre difficulté est liée au phasage des haplotypes (connaître l'allèle qui correspond à un chromosome).

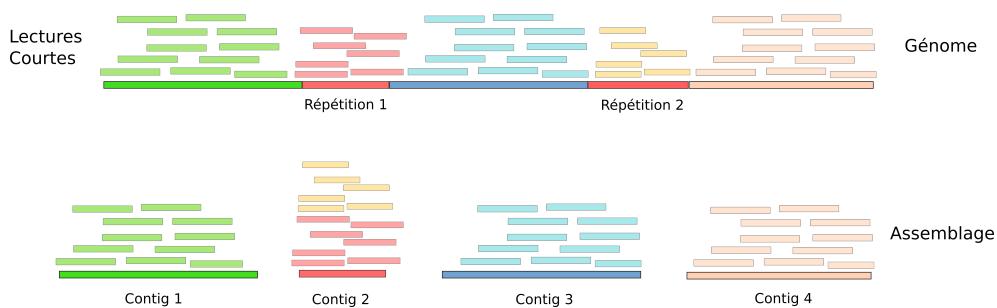


FIGURE 3: Assemblage de régions répétées

Pour pallier à ces deux problèmes, il faut utiliser des lectures de plus grandes tailles, capables de recouvrir entièrement les régions répétées ou la région correspondant à un haplotype. A l'heure actuelle, il existe trois technologies de séquençage longues lectures : Illumina TrueSeq (moleculo), Pacific BioSciences (PacBio) et Oxford Nanopore Technologies (ONT).

Bien que PacBio a été le premier à proposer des lectures longues, le coût de séquençage, d'achat de l'appareil (750 000 \$ pour le PacBio RS II ou 350 000 \$ pour le Sequel) et d'encombrement (1 tonne) rendent son utilisation quasi exclusive à de grosses structures. Un peu plus récemment (2013), ONT démocratise le séquençage de longue lectures par la création d'un séquenceur de la taille d'un harmonica relié en usb à un ordinateur et capable de séquencer des longues lectures avec un coût relativement faible (1000 \$ l'appareil).

Pour réaliser le séquençage Nanopore, l'ADN est préalablement découpé en fragments de 8 Kb ou 20 Kb (taille théorique du kit fourni). Les brins sens et antisens des fragments sont ensuite reliés entre eux par une molécule en épingle à cheveux (hairpin) pour permettre le passage des deux brins l'un à la suite de l'autre (figure 4) et liés à une molécule qui va les guider jusqu'au pore. Dans le cas du passage d'un seul des deux brins, la lecture obtenue est dite 1D (une dimension). Si le brin template et complement sont séquencés alors une lecture consensus est construite, elle est

dite 2D (deux dimensions). Lors du séquençage, le Nanopore (molécule membranaire de passage) est parcourue par un flux d'ions. Le passage des nucléotides 6 bases à la fois dans le pore va engendrer des perturbations du courant électrique parcourant la membrane. Cette perturbation est caractéristique d'un 6-mer, ce qui permet de reconstituer la séquence d'origine.

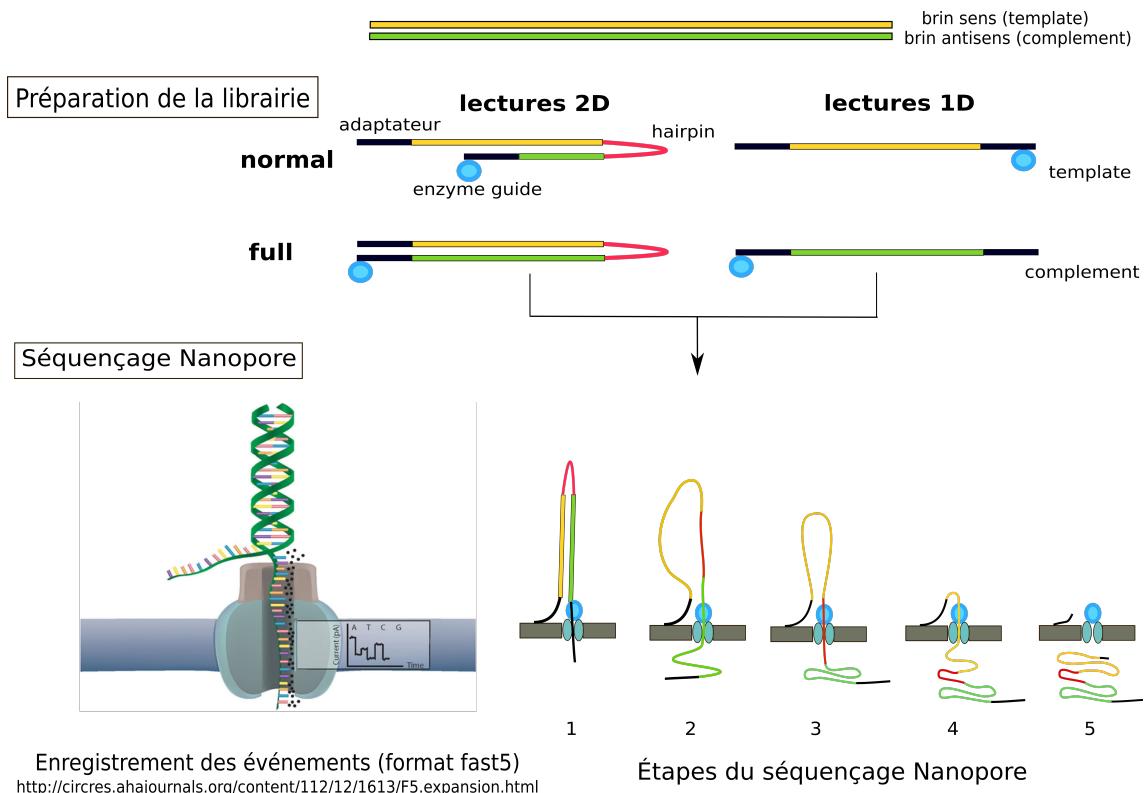


FIGURE 4: Le séquençage Nanopore

La préparation de la librairie permet de relier les deux brins à l'aide de l'hairpin (si c'est le cas des lectures 2D sont obtenues), puis les adaptateurs sont ajoutés aux extrémités du brin ainsi qu'une enzyme guide. Lors du séquençage chacune des parties du fragment va transiter à travers le pore ce qui va permettre l'émission d'évènements électriques.

L'enregistrement des événements possède cependant un défaut, il varie fortement en temps d'enregistrement ainsi il est compliqué de distinguer un ou deux événements identiques qui se suivent. Par conséquent, la détection des homopolymères est difficile avec cette technologie. Du fait de ces deux particularités, un taux important d'erreurs (environ 15%) est observé. Or ce taux d'erreur compliquant l'assemblage des séquences, une étape de correction des lectures Nanopore devient nécessaire pour pouvoir les exploiter au mieux. Plusieurs outils ont été développés dans ce sens pour corriger les lectures tel proovread [3], LoRDEC [4], ECTools [5], Nanocorr [6] et

NaS [7] (stratégie de correction développée au Genoscope). Cette stratégie hybride utilise des lectures illumina alignées sur chaque lecture Nanopore pour créer des lectures synthétiques (NaS ou Nanopore Synthetic) longues et de bonne qualité. Les lectures obtenues permettent d'effectuer des assemblages de grande qualité où les régions répétées sont prises en compte.

Cette stratégie possède cependant un gros défaut, elle est lente (15h pour corriger les lectures Nanopore d'une bactérie, *Acinetobacter baylyi ADP1*). Ceci ne permet actuellement pas (ou très difficilement) une application sur des génomes dont la taille excède 100 Mb.

1.3 Objectifs de mon travail d'Alternance

Dans le cadre du travail de l'équipe assemblage et du fait des limitations techniques (temps CPU) de la stratégie de correction des lectures Nanopore développé au Genoscope (NaS), l'objectif de mon alternance est de trouver une ou des stratégies alternatives nous permettant à la fois de garder le gain de qualité dû à la correction et d'obtenir des performances qui permettent d'envisager un traitement sur des génomes plus grands. Au cours de mon alternance, j'ai accès aux données des projets de séquençage en cours (levures) à partir desquels je vais constituer un jeu de données test qui me permet de tester les modifications apportées au fur et à mesure du développement.

De ces travaux deux problématiques majeures ressortent :

- quelle sont les étapes limitantes du logiciel ?
- quels sont les outils ou stratégies alternatives existantes qui permettent à la fois de garder la même qualité de correction et d'augmenter les performances du pipeline ?

2 Environnement Informatique, outils informatiques et nature des données

2.1 Les données tests

2.1.1 Librairie et Séquençage Illumina

Pour la préparation de la librairie Illumina, l'ADN génomique d'une levure, *Saccharomyces cerevisiae* (30-100 ng) a subi une sonication (avec un Covaris E210 sonicator) qui a aboutit à des fragments de 100 à 800 pb. L'extrémité des fragments a été réparée (adénylation en 3') et les adaptateurs ont été ajoutés (NEBNext Sample Reagent Set). Les produits de ligation ont été purifiés (Ampure XP) et les fragments de plus de 200 pb ont été amplifiés par PCR. Les librairies obtenues ont ensuite été séquencées sur un séquenceur Illumina MiSeq en utilisant un hit de séquençage de 300 pb en mode paired-end. Des fichiers au format fastq ont été obtenus.

2.1.2 Librairie et Séquençage Nanopore

Deux librairies Nanopore ont été préparées : une de 8 Kb et une autre de 20 Kb. L'ADN de *Saccharomyces cerevisiae* a été fragmenté en utilisant différentes vitesses de centrifugation pour obtenir les deux tailles de librairies. Les extrémités de fragments obtenus ont été réparées puis des adaptateurs et l'hairpin (molécule en épingle à cheveux qui relie le brin sens et le brin antisens) ont été ajoutés grâce au kit fourni par Oxford Nanopore. Les librairies obtenues ont ensuite été séquencées avec le séquenceur MinION.

La lecture des événements a été réalisé par le logiciel de contrôle MinKNOW (version 0.50.1.15 à 0.51.1.62) et le base calling a été fait avec le logiciel Metrichor (version 2.26.1 à 2.38.3). Les données générées par le logiciel du MinION ont été stockées et organisées en utilisant un format de données hiérarchiques (HDF5). Trois types de lectures ont été obtenus : template, complement et two-directions (2D). Les lectures template et complement sont combinées pour obtenir une lecture consensus (2D). Les fichiers HDF5 ont été convertis au format FASTA à l'aide de poretools [8]. Le contrôle qualité a été effectué par alignement avec LAST (version 588 [9]) des lectures sur le génome de référence de *S. cerevisiae* S288C.

2.1.3 Constitution du jeu de données test

Dans l'objectif d'améliorer le workflow de correction des lectures Nanopore, un jeu de données test a été réalisé pour comparer les temps entre les différentes évolutions du programme. Ce

dernier est constitué de l'ensemble des lectures Illumina (150X soit 8,36 millions de lectures), et de 1000 lectures Nanopore sélectionnées aléatoirement et ayant une taille supérieure à 500 pb.

2.2 Environnement Informatique

2.2.1 Cluster de Calcul

Le Genoscope, pour effectuer ses différents traitements, dispose d'un cluster de calcul composé de nœuds de 24 ou 36 coeurs pour un total de 748 coeurs et 13 To de RAM. Dans le cas de calcul plus conséquents, les traitements sont effectués au Centre de Calcul Recherche et Technologie (CCRT) qui dispose d'un cluster de 3000 coeurs. L'ensemble du parc informatique du Genoscope étant constitué de systèmes d'exploitation windows 7 à 64 bits, la connexion au cluster de calcul s'effectue via ssh (protocole crypté de connexion à un réseau) par le biais de MobaXterm (ce logiciel simule le fonctionnement d'un terminal UNIX sur Windows).

Deux utilitaires de gestion de ressources existent au Genoscope : Slurm et Lsf. Au vu de la forte transition des nœuds et de l'utilisation du personnel de Lsf vers Slurm, ce dernier utilitaire a été favorisé. Lors du lancement d'un traitement (ou job), la commande jobify permet de gérer la demande de ressource relative au traitement (exemple : on demande 20 coeurs). Dans un soucis de comparaisons efficace et pour éviter toute influence d'autres traitements sur les tests, il a été décidé que le benchmarking serait effectué à raison de 24 coeurs sur des nœuds de 24 coeurs. Le nœud entier est ainsi utilisé (afin d'éviter l'influence d'autres traitements).

2.2.2 Poste Personnel

Lors de cette alternance, l'ensemble des travaux ont été réalisés sur un environnement Windows 7 64 bits. La connexion au cluster est effectuée via ssh par le biais de MobaXterm. Ce dernier simule le fonctionnement d'un terminal linux, ce qui permet d'effectuer l'ensemble des traitements et tests nécessaires sur le cluster. Mon poste personnel est composé d'un dual core (intel core I3) cadencé à 3.5 GHz et de 4 Go de RAM.

2.3 Méthodologie de travail

Mon travail d'alternance se décompose en deux phases : test minimal de nouvel outil (benchmarking) et implémentation (dans le cas de résultats positifs). Dans cet objectif un premier travail de recherche bibliographique a été effectué pour trouver des outils similaires à ceux utilisés dans

le traitement limitant. Une fois l'outil trouvé, ce dernier est testé sur un exemple minimal. Si ce dernier montre des performances encourageantes et une bonne qualité de résultats, alors il est mis en œuvre. L'impact en terme de performance sur le traitement total de correction est ensuite observé.

2.3.1 Veille Bibliographique

La veille bibliographique a été assurée durant ce stage à l'aide de PubMed et du site Omic Tools (<https://omictools.com/>) qui permet de trouver des outils dont le fonctionnement est similaire. Le logiciel Mendeley a également été utilisé pour conserver et annoter l'ensemble des publications des outils.

2.3.2 Langages et commandes utilisées

Durant ce stage différents langages ont été utilisés. le logiciel [R] a servit à la comparaison visuelle des contenus en séquences de deux fichiers fastq (via un script R et des librairies pour les diagrammes de Venn). Pour les manipulations quotidiennes de fichiers j'ai utilisé les langages de script awk et bash ainsi que les commandes sed et grep. J'ai également appris à me servir de la commande "parallel" [10] (celle-ci est très utilisée dans les programmes du Genoscope comme NaS) et "paste" (qui est très utile pour le parsing de fichiers fastq).

La commande parallel permet d'appliquer un même traitement sur un ensemble de fichiers de manière simultanée (c'est de la parallélisation). Elle s'avère utile car elle permet d'éviter, lorsque cela est possible, l'implémentation d'un script avec du multithreading (exécution simultanée de traitements différents à l'intérieur du script) qui s'avère souvent plus complexe à mettre en œuvre. La parallélisation se fait alors en découplant le flux d'entrée et en travaillant par bloc avec des exécutions en parallèle de plusieurs blocs.

2.4 Description et fonctionnement de NaS

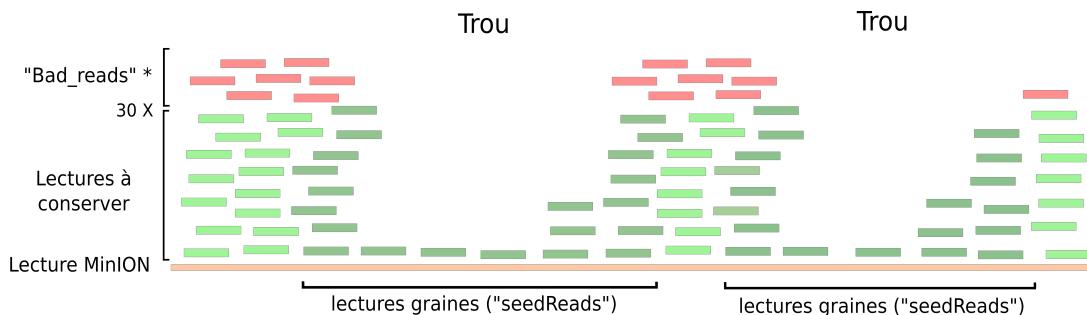
Pour comprendre la nature même de mon travail et les différentes implémentations réalisées au cours de cette alternance, il est important de comprendre l'implémentation même de NaS. Ce dernier est organisé dans un wrapper (un script qui appelle l'exécution d'autres programmes) écrit en bash. Le wrapper permet de gérer l'ensemble des paramètres des différents programmes appelés.

NaS fonctionne de la manière suivante (figure 8) : après avoir créé les différents répertoires pour le fonctionnement du workflow, les lectures Illumina au format fastq sont converties au

format fasta et regroupées dans un unique fichier.

Les séquences Illumina sont ensuite alignées sur chaque lecture Nanopore. Deux programmes d'alignements peuvent être utilisés en fonction du mode sélectionné : fast (BLAT [11]) ou sensitive (LAST [12]). BLAT garde en mémoire l'index du génome (celui-ci est constitué de 11-mers non chevauchants) et a été conçu pour aligner rapidement des séquences similaires à 95 %. LAST quant à lui est un aligneur plus sensible capable de trouver des similarités moindre entre séquences (il peut prendre en compte davantage de mismatchs) et gère mieux les répétitions dans l'alignement. Cependant, il reste 2 à 3 fois moins rapide que BLAT et est donc à utiliser dans le cas de génomes complexes ou de résultats insuffisants obtenus avec BLAT. Pour l'ensemble de mes tests, j'ai utilisé le mode fast (avec BLAT).

Après l'alignement des séquences Illumina sur les lectures Nanopore, les lectures alignées sont souvent insuffisantes pour recouvrir entièrement (couverture horizontale) et de manière profonde (couverture verticale) la lecture Nanopore. Elles sont donc utilisées comme lectures graines (seed reads) pour un nouveau recrutement de lectures Illumina qui servira à combler les gaps (zones au sein de la lecture Nanopore faiblement couvertes) en vue d'un futur micro-assemblage (l'ensemble des lectures Illumina correspondant à une lecture Nanopore seront co-assemblées). Cette étape de recrutement est l'étape limitante du pipeline actuellement (70% du temps de traitement) malgré une parallélisation.



* Lectures n'apportant pas de nouvelle information

FIGURE 5: La sélection des SeedReads

Le script de recrutement des lectures parcourt d'abord le fichier d'alignement (obtenu avec BLAT ou LAST) pour sélectionner les 30 premiers X (30 premières lectures participant à la couverture verticale d'une zone de la lecture Nanopore, figure 5), ceux-ci sont appelés "good_reads". Les lectures dépassant cette limite ne sont pas prises en compte (30X suffisent pour le micro-assemblage qui s'en suit), elles sont appelées "bad_reads" (bien qu'elles pourraient elles aussi par-

ticiper à l’assemblage). Le script de sélection de lectures Illumina similaires aux seedReads, SelectReads, considère ensuite les zones faiblement couvertes (moins de 5X) et les définit comme gap. Les lectures en amont et en aval d’un trou (impliquées dans une fenêtre de 1000 bases des deux cotés du trou) sont considérées comme participant à ce dernier et seront utilisées pour le recrutement (les zones fortement couvertes ne sont pas utilisées car le recrutement n’est pas nécessaire dans ces régions). La taille cumulée de la zone considérée comme gap (impliquant les zones en amont et en aval de ces derniers) sert à calculer le nombre minimal et maximal de lectures à recruter (figure 6) pour considérer le recrutement comme effectif.

$$NbReads_{max} = \frac{(nb_gaps + taille_fenetre) \times Couverture_max}{(2 \times Longueur_lecture)} \quad (1)$$

$$NbReads_{min} = \frac{(nb_gaps + taille_fenetre) \times Couverture_min}{(2 \times Longueur_lecture)} \quad (2)$$

FIGURE 6: Calcul des scores de recrutement

Le recrutement se fait en 5 cycles au maximum et les scores servent de contrôle pour le recrutement (si la couverture minimale est atteinte dans les zones de gap d’une lecture Nanopore alors on arrête le recrutement). Le recrutement est effectué par Compareads [13]. Ce logiciel compare deux ensemble de lecture et recrute celles qui partagent trois 31-mers disjoints (par défaut). Si le recrutement est suffisant (supérieure au score minimal de lectures à recruter) alors un micro assemblage (assemblage des lectures correspondant à une lecture Nanopore) est effectué à l’aide de NEWBLER. Cet assembleur utilise une approche OLC (Overlap Layout Consensus) pour procéder à l’assemblage des lectures. Dans les régions fortement répétées, le micro-assemblage peut amener à des problèmes de graphes complexes (plusieurs contigs correspondent à une même région). Pour pallier à ce soucis, un script, Untangle_complex_regions, est chargé de sélectionner le chemin dans le graphe le plus cohérent avec l’alignement (figure 7).

Des lectures NaS sont ainsi obtenues et une dernière étape permet d’obtenir des métriques de qualité (N50 , taille cumulée ...) des lectures Nanopore et NaS (une observation de la correction des lectures Nanopore est rendue possible).

En sortie de NaS, un alignement avec bwa (paramétré pour prendre en compte les longues lectures) des lectures NaS obtenues a été effectué sur la référence pour obtenir des métriques d’alignement (% d’identité, proportion des lectures alignées ...). Cette dernière permet également d’avoir une idée de la qualité de la correction avec NaS. L’ensemble de ces étapes est résumé sur

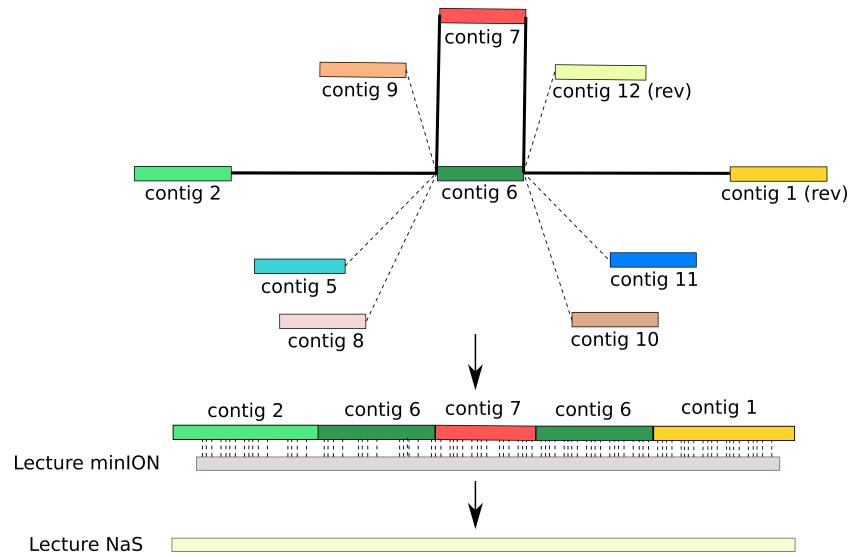


FIGURE 7: Sélection d'un chemin dans le graphe des contigs, à l'aide de l'alignement des contigs sur la lecture MinION

la figure 8. Mon travail d’alternance se situe au niveau du remplacement de l’outil de recrutement des lecture (en bleu sur la figure 8).

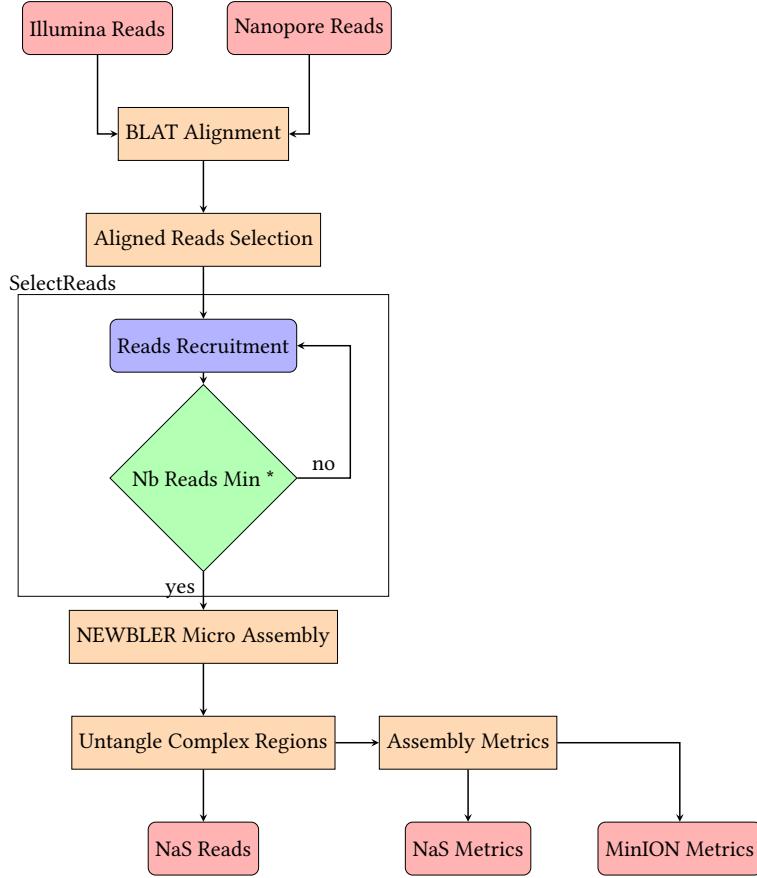


FIGURE 8: Schéma Général du fonctionnement du workflow de NaS

* or > 5 iterations

Les lectures Illumina sont tout d'abord alignées sur les lectures Nanopore avec BLAT puis recrutées.

Si le nombre de lectures est insuffisant, un recrutement est relancé. Ensuite, les lectures sont micro-assemblées et scaffoldées dans le cas de graphes de contigs complexes. Enfin, les métriques sont générées.

2.5 La Stratégie KMC2 - Cookiecutter

Dans l'objectif de remplacer l'outil de recrutement de lectures actuellement utilisé dans NaS, une proposition de remplacement par deux outils associés a été proposée. Ces outils sont KMC2 [14] et Cookiecutter [15]. KMC2 est un compteur de k-mer basé sur le principe des minimiseurs (plus petite séquence capable de distinguer un k-mer d'un autre k-mer, figure 9).

Minimizers	
CGTTGATCAATTG	Read
CGTTGATC	Minimizer: rev_comp(CGTT) = AACG
GTTGATCAAT	Minimizer: rev_comp(TGAT) = ATCA
GATCAATT	Minimizer: AATT
ATCAATTG	Minimizer: rev_comp(ATTT) = AAAT

FIGURE 9: Exemple de construction d'un minimiseur [14]

Dans notre implémentation (figure 10), il a été utilisé pour créer une librairie de k-mer à partir des seedreads. Une fois la librairie créée, Cookiecutter effectue le recrutement sur la base de l'algorithme d'Aho-Corasick (pour la recherche de chaînes de caractères, dans notre cas de séquences). L'algorithme d'Aho-Corasick construit un automate à partir d'un ensemble de mots (ici la librairie de k-mers) ce qui minimise fortement l'espace utilisé (il évite certaines redondances dans l'automate). Il effectue ensuite une recherche à partir de l'automate construit dans l'ensemble des lectures et recrute celles qui partagent un k-mer en commun (une taille de k-mer de 100 a été choisie dans notre cas).

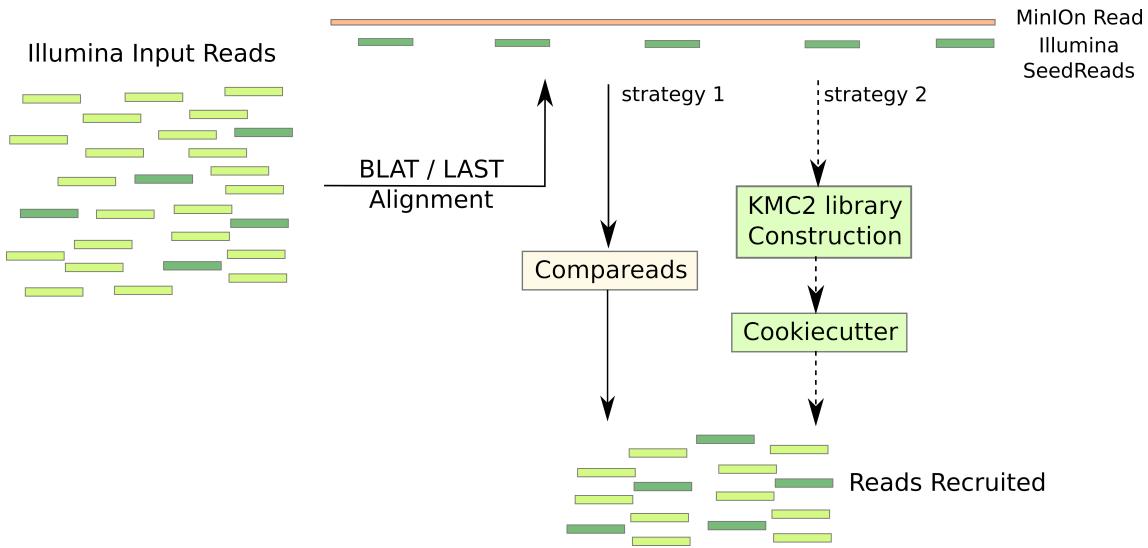


FIGURE 10: Schéma Général du recrutement avec KMC2 et Cookiecutter

2.6 La Stratégie Minimap

Une autre stratégie a également été proposée, elle consiste à aligner au préalable l'ensemble des lectures Illumina entre elles. Ensuite les résultats d'alignement servent au recrutement des lectures impliquées dans une lecture Nanopore. Dans cet objectif, un nouvel outil d'alignement,

Minimap [15] a été utilisé. Ce dernier a normalement été conçu pour aligner des lectures longues et bruitées (PacBio ou Nanopore) mais a été ici utilisé pour retrouver les lectures Illumina similaires. L'ensemble des associations entre lectures a ensuite été regroupé dans un fichier, appelé fichier d'association. Ce fichier possède pour chaque ligne une lecture représentante et un ensemble de lectures associées (le nombre de lectures associées est limité à 100 pour limiter la taille du fichier).

Lors du recrutement (figure 11), les seedreads correspondant à une lecture Nanopore sont stockées en mémoire. Le fichier d'association est parcouru ligne par ligne et si la lecture représentante de la ligne correspond à l'un des seedreads alors l'ensemble de lecture qui lui est associé sera recruté. Cette stratégie procède donc à un recrutement par "bloc".

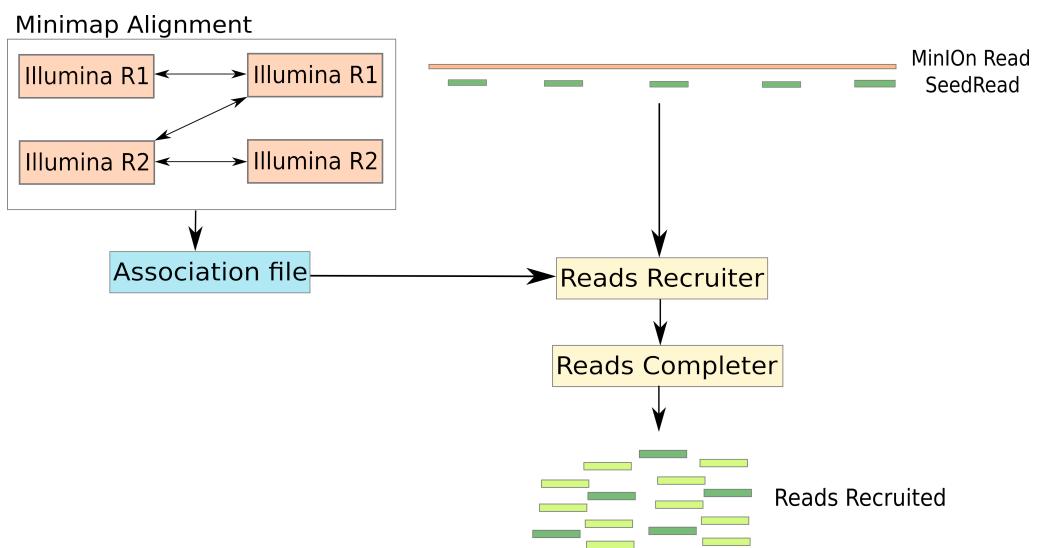


FIGURE 11: Schéma Général du recrutement avec Minimap

Cette stratégie a été implémentée en remplacement de Compareads (pour le recrutement des lectures), elle est donc exécutée en parallèle (un traitement par lecture Nanopore). Cette stratégie nécessite donc de parcourir le fichier d'association pour chaque lecture.

Une autre stratégie parcourant le fichier d'association une seule fois pour limiter le temps de lecture du fichier a elle aussi été implémentée. Cette dernière, globale, prend tout d'abord en mémoire les seedreads correspondant à chaque lecture Nanopore. Ensuite un seul parcours du fichier d'association est effectué. Au fur et à mesure de la lecture du fichier d'association, le recrutement est effectué et les lectures recrutées sont réparties dans les fichiers de sortie correspondant aux lectures Nanopore concernées.

3 Résultats

3.1 Architecture logicielle

Au cours de cette alternance, j'ai suivi l'architecture adoptée précédemment pour le développement du pipeline de correction. J'ai ajouté des options pour le choix de l'outil lors de la correction des données permettant de tester les nouvelles stratégies implémentées. Comme expliqué précédemment, le pipeline est regroupé dans un wrapper qui appelle différents outils et scripts (majoritairement codés en Perl). Le script Perl modifié (`selectReads`) est organisé en fonctions qui sont appelées différemment selon les options choisies par l'utilisateur.

Le wrapper, lors de l'appel des différents scripts, utilise parallel pour exécuter plusieurs traitements de manière simultanée.

3.2 Visualisation pour la comparaison de données

Pour comparer les contenus en lectures communes et celui des librairies de k-mers générées, un script R a été créé. Il utilise la librairie `VennDiagram` et permet de visualiser les différences de contenus en lectures à l'aide de diagrammes de Venn (figure 12). Cette comparaison par visualisation permet d'encourager l'utilisation d'un outil si les résultats obtenus sont similaires à ceux obtenus avec `Compareads` (notre but étant de garder la même qualité pour des performances de calcul supérieures).

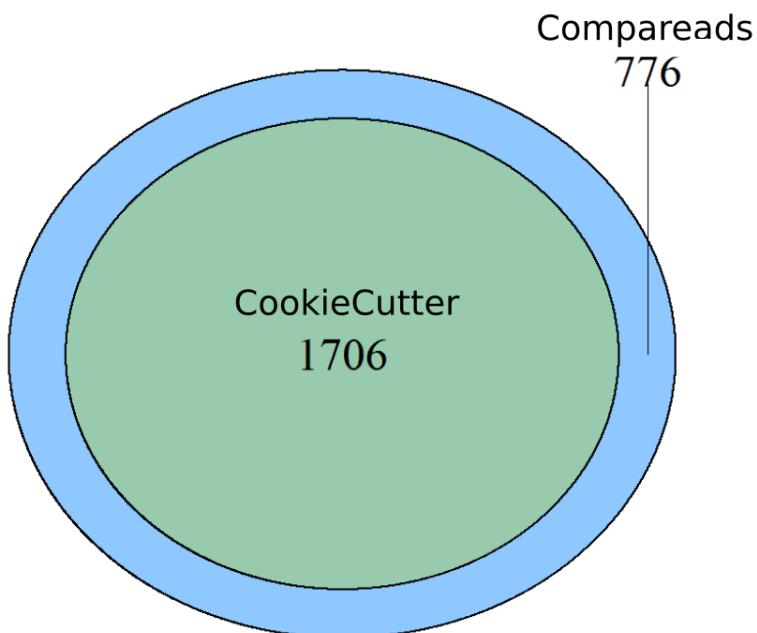


FIGURE 12: Comparaison du recrutement avec *Cookiecutter* (Vert) et *Compareads* (Bleu)

Cette comparaison visuelle a également permis de sélectionner KMC2 comme outil pour la construction de la librairie (c'est l'outil le plus rapide, qui associé à Cookiecutter, nous permettait d'obtenir des résultats similaires à ceux de compareads). Dans l'exemple donné figure 12, le contenu en lectures recrutées a été comparé entre compareads (bleu) et cookiecutter (vert). On remarque dans cet exemple que cookiecutter recrute moins de lectures (1706 lectures) que compareads (2482 lectures) mais les lectures recrutées correspondent à des lectures également recrutées par compareads.

Une autre forme de visualisation a été utilisée pour permettre la comparaison de la qualité des lectures avant et après correction, le mummerplot. Un mummerplot est un graphique représentant l'alignement de deux séquences (figure 13). Si les deux séquences comparées sont fortement similaires alors la ligne rouge (représentant les zones identiques) sera continue (et discontinue dans le cas contraire). Dans le cas d'un alignement inversé les séquences sont représentées en bleu et sont orientées de manière inverse par rapport à la référence.

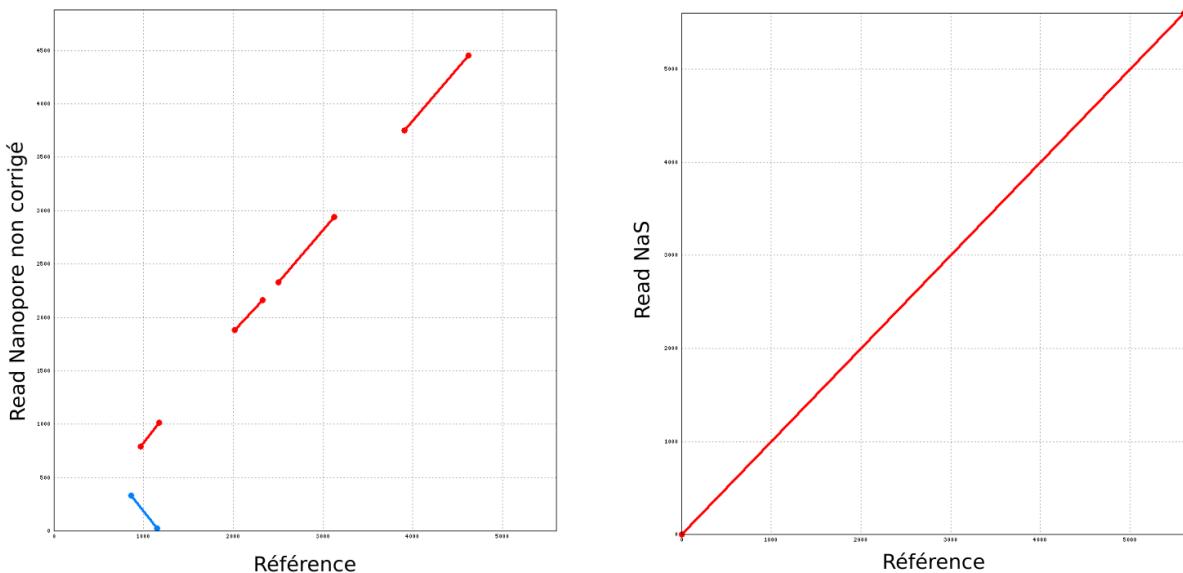


FIGURE 13: Alignement de la séquence ch102_file28 sur la référence avant correction et après correction (NaS)

3.3 Implémentation de KMC2 - Cookiecutter

L'implémentation de KMC2 et de Cookiecutter a été effectuée dans le script de recrutement des lectures, selectReads, par l'ajout d'une fonction spécifique pour CookieCutter. Vu que cette implémentation a repris l'architecture du script, elle n'a nécessité que l'ajout d'un traitement alternatif faisant appel à KMC2 puis CookieCutter ainsi qu'une option de sélection de l'outil pour l'utilisateur.

Les premiers résultats ont montré un gain de 20% dans le temps de recrutement (sur des fichiers très petits, avec moins de 1000 seedreads) et nous ont donc encouragés dans des tests à plus grande échelle. Un test a été effectué sur le jeu de données test créé précédemment et les métriques de qualité de l'alignement sur la référence ont été comparées à celles obtenues sans correction et avec correction de l'ancienne méthode impliquant Compareads. Les résultats sont résumés dans le tableau 1.

Tableau 1: Comparaison de la Stratégie Compareads et Cookiecutter

	Nanopore	NaS Compareads	NaS KMC2-Cookiecutter
Number of Reads	1000	837	836
Cumulative Size (Mb)	6.5	6.06	6.03
N50	8007	8683	8624
Max Size	74921	22992	22992
Average Size	6518	7249	7220
Identity percent (%) vs reference	78	99.84	99.86
Number of Aligned Reads	876 (87.6%)	831 (99.28%)	830 (99.28%)
Elapsed time	-	11 h 26 min	12 h 34 min

Un premier état des lectures avant correction (colonne 1) permet de se rendre compte de la qualité des lectures Nanopore brutes. On remarque que ces dernières peuvent être assez longue (quasiment 75 Kb pour la lecture la plus longue) avec une taille moyenne légèrement supérieure à 6.5 Kb (cette moyenne est supérieure à celle d'une librairie complète car les séquences inférieures à 500 bases sont exclues dans notre jeu de données test). L'alignement sur le génome de référence a aboutit à un pourcentage d'identité faible (78 %).

On constate sur notre jeu de données test que la correction a tendance à diminuer la taille cumulée en passant d'une valeur de 6.5 Mb à 6.03 ou 6.06 Mb (on a donc une perte de séquences lors de la correction). Cependant les autres métriques s'améliorent lors de la correction. Des séquences de plus grande taille sont observées lors de la correction. En effet, le N50 augmente d'environ 600 pb pour Compareads et Cookiecutter. La taille moyenne des séquences augmente elle aussi de 700 pb lors de la correction. Le pourcentage d'identité lors de l'alignement contre la référence augmente très fortement pour être quasi identique à cette dernière (% d'identité supérieur à 99.8 % pour Compareads et Cookiecutter). On constate globalement que l'on a bien corrigé les lectures

Nanopore avec les deux stratégie et les métriques obtenues sont très similaires.

Cependant les performances observées pour les deux stratégies diffèrent. En effet la stratégie avec Compareads qui nous sert de référence a nécessité 11 heures et 26 minutes pour corriger les 1000 lectures Nanopore tandis que celle basée sur KMC2 et Cookiecutter a nécessité plus d'une heure supplémentaire pour un résultat très similaire. La stratégie KMC2-Cookicutter qui avait montré des résultats encourageants pour des petits jeux de données s'avère en fait plus lente que Compareads lors de l'augmentation de la taille du jeux de données testé. Cette stratégie a donc été invalidée par la perte de performance observée lors de l'augmentation de la taille de fichiers.

3.4 Implémentation parallélisée de Minimap

La seconde implémentation a nécessité l'utilisation de Minimap. Les trois alignements réalisés ont été concaténés et triés (seul le nom des lectures associées ont été gardés) pour créer le fichier d'association (d'une taille de 150 Go). Le fichier d'association est formé de la lecture représentante (en première position de la ligne) et de l'ensemble des lectures qui lui sont associées (séparées d'espaces).

Un script a été créé pour permettre le recrutement à partir du fichier d'association. Ce dernier retrouve le nom des lectures associées à celles du fichier de seedreads. Enfin, un second script a été développé pour reconstituer des fichiers fastq à partir des fichiers constitués des noms de lectures en sortie de recrutement et des fichiers Illumina, être utilisé pour la suite du workflow. L'intégration des scripts précédents a ensuite été faite dans le script selectReads (comme celle de KMC2 et Cookiecutter). Le programme est donc exécuté de manière parallèle. Pour chaque lecture Nanopore, le fichier d'association est parcouru et les lectures associées aux seedreads sont recrutées.

Nous constatons pour la correction avec la stratégie minimap parallélisé, que les métriques de qualité des lectures (N50, average size, voir Tableau 2) sont inférieures d'approximativement 100 pb par rapport à celles obtenues avec Compareads. Une petite partie de la perte de taille cumulée (100 Kb) peut être expliquée par les trois lectures manquantes lors de la correction avec Minimap. Les autres métriques, % d'identité et nombre de lectures alignées sur la référence, sont très similaires entre les deux stratégies. Le temps de traitement est cependant inférieur dans le cas de la stratégie Minimap (9 h 21 min au lieu de 11 h 26 min pour Compareads) et ce malgré l'ajout du prétraitement des données. Les deux heures de prétraitements peuvent être retirées dans le cas d'une nouvelle correction sur le même jeu de lectures Illumina.

Tableau 2: Comparaison de la Stratégie Compareads et Minimap Parallélisée

	Nanopore	Nas Compareads	Nas Minimap Parallelized
Number of Reads	1000	837	834
Cumulative Size (Mb)	6.5	6.06	5.96
N50	8007	8683	8575
Max Size	74921	22992	24733
Average Size	6518	7249	7150
Identity percent (%)	78	99.84	99.87
Number of Aligned Reads	876 (87.6%)	831 (99.28%)	829 (99.4%)
Elapsed time	-	11 h 26 min	7 h 21 min (+~ 2h*)

Cependant cette stratégie, du fait de la taille du fichier d’association (150 Go pour une levure), est limitée par le débit de lecture du noeud sur lequel est effectué le traitement (environ 600 Mo/sec dans le cas de notre cluster de calcul). Cela s’avère d’autant plus limitant que cette stratégie est parallélisée, le débit de lecture est donc divisé par le nombre de cœurs alloués au traitement (la parallélisation n’impacte donc que faiblement le gain de temps). Ainsi une stratégie alternative a été proposée pour pallier aux nombreuses lectures du fichier d’association, Minimap global.

3.5 Implémentation globale de Minimap

L’implémentation de la Stratégie Minimap global, a nécessité plusieurs modification au sein du workflow. Premièrement, le script selectReads (chargé de trouver les seedreads à partir d’un alignement et d’effectuer le recrutement) a été amputé de sa partie recrutement. En effet, le recrutement est voulu de manière globale (pour l’ensemble des fichiers à la fois) or ce dernier est exécuté en parallèle dans selectReads. Les seedReads sont récupérés en sortie de selectReads et stockés en mémoire.

Une première version (figure 14 avec pointillés) de cette implémentation mimait le fonctionnement de selectReads, les informations concernant la couverture étaient reprises pour calculer les scores minimum et maximum en vue de limiter le recrutement. Cependant les résultats obtenus ne montrait aucun gain de qualité lors de la correction. Il existe en effet un cas limite dû à la correction par bloc associé à la limite du score. En effet, pour fixer la limite du recrutement des lectures, un score maximum est calculé selon la couverture moyenne souhaité des régions considérées comme trou (trou et régions adjacentes). Or, vu que la limite est fixée sur la couverture moyenne et non

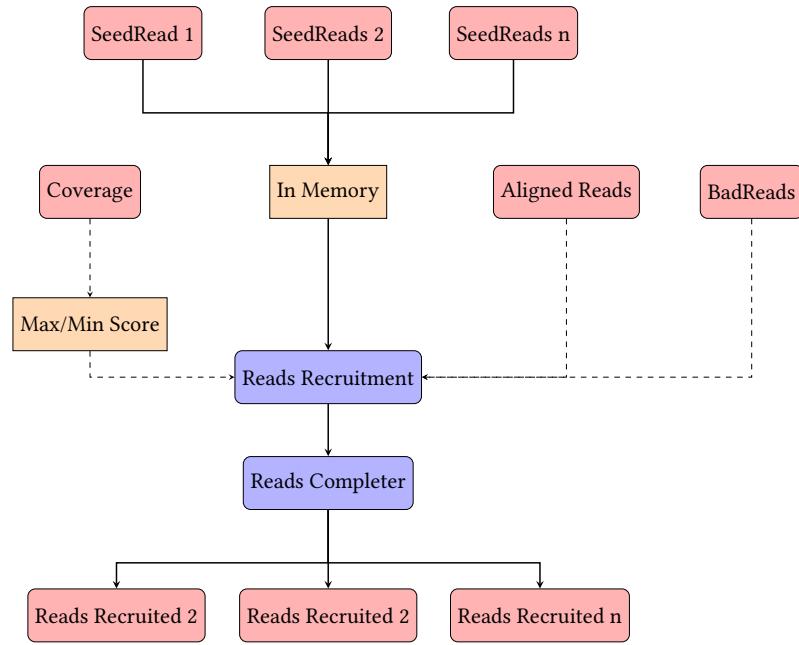


FIGURE 14: Schéma Général du fonctionnement du workflow de recrutement minimap global

par base, il est possible d'atteindre le score max en recrutant de manière hétérogène sur les trous (figure 15).

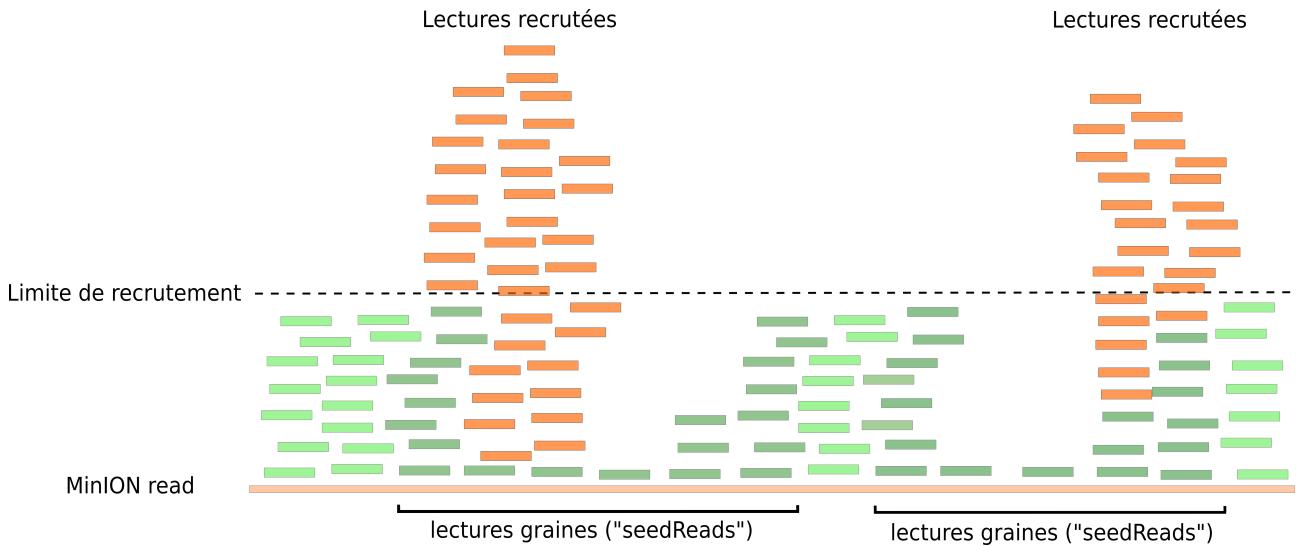


FIGURE 15: Schéma de la configuration limite du recrutement par bloc

On remarque que dans cette configuration, le recrutement atteint sa limite maximum sans pour autant combler les trous (du fait de la répartition très hétérogène des lectures recrutées). Une seconde version du script de recrutement a alors été développée sans limite de score (figure 14 sans pointillés). Cette dernière stratégie a montré des résultats similaires à ceux obtenus avec Compareads (voir Tableau 3).

Tableau 3: Comparaison de la Stratégie Compareads et Minimap Global

	Nanopore	NaS Compareads	NaS Minimap Global
Number of Reads	1000	837	825
Cumulative Size (Mb)	6.5	6.06	6.00
N50	8007	8683	8692
Max Size	74921	22992	25612
Average Size	6518	7249	7273
Identity percent (%) vs reference	78	99.84	99.81
Number of Aligned Reads	87.6%	831 (99.28%)	820 (99.39%)
Elapsed time	-	11 h 26 min	6 h 52 min (+~ 2h*)

* temps lié à la création du fichier d'association

Nous constatons que les résultats obtenus sont très similaires entre la stratégie Compareads et la stratégie Minimap Global. Moins de lectures sont tout de même obtenues lors de la correction avec la stratégie basée sur Minimap (825 au lieu de 837) mais on observe une longueur maximum supérieure. Les autres métriques (Cumulative Size, N50, Average Size et Identity Percent) restent très similaires. Le temps de traitement, quant à lui, passe de 11 h 26 min à 8 h 52 (avec la création du fichier d'association).

Ce temps sur la stratégie Minimap Global a été obtenu avec un fichier d'association constitué d'ensemble de 100 lectures associées par lecture représentante. De façon à améliorer encore le temps de traitement, nous nous sommes intéressé à l'influence de la taille de l'ensemble de lectures associé à la lecture représentante dans le fichier d'association. Plusieurs tests ont ainsi été effectués sur des tailles d'ensemble allant de 10 à 100 dans l'objectif de trouver un plateau de qualité au delà duquel la taille de l'ensemble associé n'impacterait que faiblement le gain de qualité. Les résultats de ces tests sont résumés dans la figure 16.

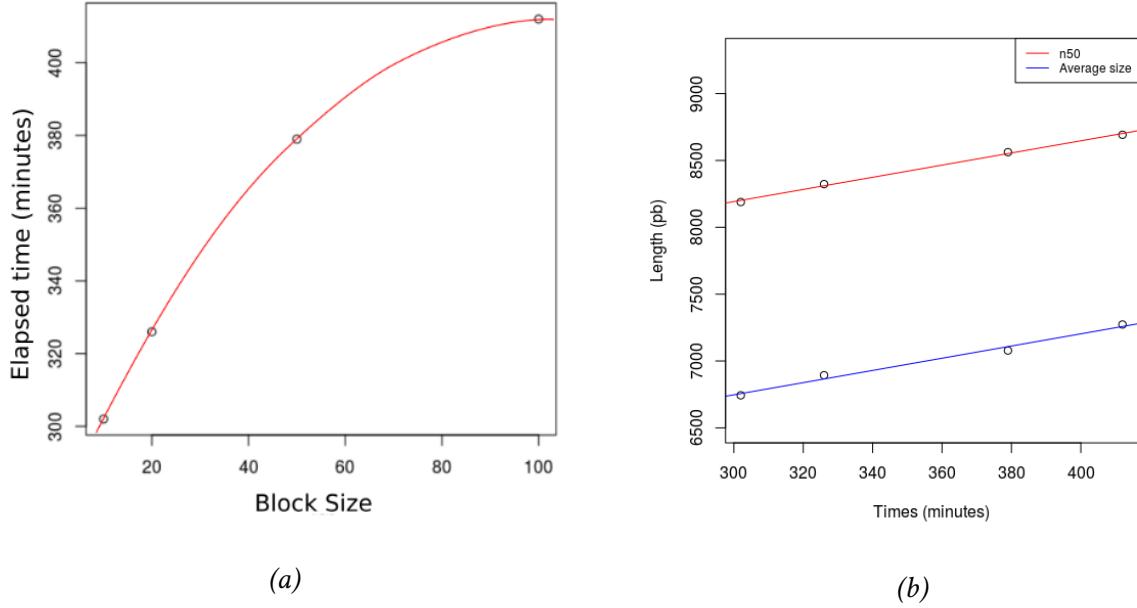


FIGURE 16: Evolution du temps d'exécution (a) et le la taille de lectures corrigées (b) en fonction de la taille de bloc

Nous constatons que le N50 et la taille moyenne de lecture semblent corrélés avec le temps d'exécution et la taille de bloc. Ainsi, le N50, la taille moyenne de lectures et le temps d'exécution augmentent de manière linéaire en fonction de la taille de bloc. Il n'y a donc pas de plateau net qui permet de discriminer des tailles de bloc. Il est important de faire remarquer que le caractère qui nous permet d'observer la correction des lectures sont les mesures de son expansion (avec le N50 et la taille moyenne). Ainsi, les faibles N50 et taille moyenne de lectures ne signifient pas forcément que la correction sur la lecture Nanopore est mauvaise (il faut faire un assemblage des lectures recrutées pour le vérifier).

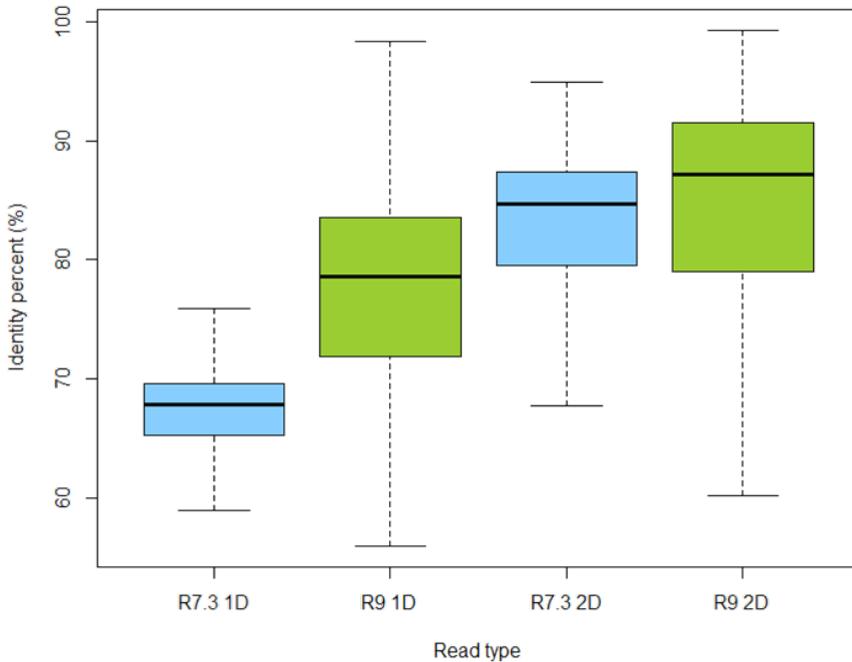
4 Discussions

4.1 Évolution de la technologie Nanopore

L'un des avantages majeurs de la technologie Oxford Nanopore réside dans sa possibilité de séquencer de très longs fragments d'ADN. Au cours de nos analyses, nous avons obtenues des lectures 2D allant jusqu'à des tailles de 75 Kb, indiquant que le système est capable de lire sans interruption au moins 150 000 nucléotides. De plus, les résultats de ces analyses indiquent que le taux d'erreur des lectures MinION actuelles est semblable à celui obtenu avec les autres technologies longues lectures (environ 15% pour les lectures 2D).

Cependant, les erreurs ne sont pas réparties de manière aléatoire et impactent fortement la lecture des homopolymères (cette caractéristique semble intrinsèque à la technologie Oxford Nanopore). Parce que la détection dans le pore se fait par 6 nucléotides, la séparation des événements pose problème dans le cas d'homopolymères dont la longueur dépasse 6 bases [16]. Avec la version actuelle du système, les homopolymères ont tendance à favoriser les délétions (ce qui représente 66% des erreurs observées dans les homopolymères). Cela pourrait être amélioré à l'aide d'une vitesse constante ou d'une augmentation de la vitesse de passage à travers le pore. De la même manière, l'algorithme du basecaller pourrait être optimisé pour augmenter la précision de l'appel de base. ONT a annoncé récemment plusieurs changements dans la prochaine version du système, incluant un mode rapide (250 bp / secondes au lieu des 70 actuellement) et de nouveaux basecaller basés sur les réseaux de neurones. Après plusieurs tests de ce nouveau système, nous avons constaté une diminution du taux d'erreurs pour les lectures 1D et 2D (figure 17).

Nous constatons sur la figure 17 que la diminution du taux d'erreurs est représentée par l'augmentation du pourcentage d'identité (par rapport à la référence chez *Acinetobacter baylyi ADP1*) lors du passage d'un système à l'autre. Avec l'utilisation de ce nouveau système, les homopolymères favorisent l'apparition d'insertions. Ces erreurs posent problème car elles aboutissent à la construction de contigs de moindre qualité. De plus, les indels (insertions délétions) impactent fortement la prédiction de gènes car elles provoquent des décalages du cadre de lecture. Une correction des lectures est donc toujours d'actualité même si nous nous attendons à une augmentation de la qualité des séquences obtenues avec la technologie Oxford Nanopore dans les années à venir.



*FIGURE 17: Évolution du pourcentage d'identité des reads produits par le MinION pour les versions actuelle (R7.3 en bleu) et à venir (R9 en vert) sur *Acinetobacter baylyi* ADP1*

4.2 Veille technologique

Inscrit dans un travail de veille technologique, les travaux d'optimisation n'échappent pas aux difficultés de ce type d'approche. En effet, les choix initiaux d'outils ou d'algorithmes ont parfois évolués au profit d'alternatives plus performantes. Ces choix se basent tout d'abord sur une recherche bibliographique en vue de trouver un outil qui correspond à nos attentes non seulement en terme de fonctionnement mais aussi au point de vue des performances (qui sont fortement recherchées dans notre cas). Par la suite, la pertinence de l'outil est testée à très petite échelle (il peut en effet arriver que les résultats décrits dans un article ne correspondent pas à ceux retrouvés après test). Malgré ces précautions nous avons constaté que ces tests minimaux peuvent être insuffisants pour bien évaluer la pertinence d'un outil. En effet, dans le cas de Cookicutter, les premiers tests se sont avérés encourageants avec un gain de 20% dans le temps d'exécution par rapport à Compareads mais rapidement décevants par la suite ce qui nous a obligé à abandonner cette stratégie. Du fait de l'évolution rapide des algorithmes et des outils, le choix devient de plus en plus important pour investir au mieux le temps de travail. L'arrivée de nouveaux outils nous oblige parfois à réviser notre approche pour toujours viser les meilleures performances (ce fut le cas pour Minimap qui nous a permis d'envisager une approche globale).

Une phase importante d'appropriation des scripts (compréhension des commandes nouvelles et des librairies) a été nécessaire et ce notamment pour l'implémentation de la stratégie globale avec minimap qui s'inspire fortement de la logique du script de recrutement développé au préalable. Quelques difficultés ont en effet été rencontrées et ont nécessité de réétudier les scripts pour bien acquérir le raisonnement. Un autre travail constant d'adaptation / réadaptation des paramètres a lieu pour garder un équilibre entre bonne correction des lectures et augmentation significative des performances de recrutement (ce travail est d'ailleurs toujours en cours).

5 Conclusions et Perspectives

Les objectifs de cette alternance visaient à mettre en place une veille technologique et une optimisation du pipeline NaS permettant de corriger des lectures Nanopore en vue d'application sur des génomes plus larges. En effet, l'une des problématique de l'alternance reste l'application de la correction sur des volumes de données plus importants. A cette fin, l'objectif était de trouver une stratégie ou des outils alternatifs plus performants et permettant d'obtenir la même qualité de correction des lectures.

Après une première année d'alternance, tous les objectifs ne sont pas atteints mais les approches implémentées sont encourageantes. Bien que du temps a été passé sur l'implémentation qui allie KMC2 et CookieCutter et que cela s'est avéré décevant, l'implémentation de la stratégie globale avec Minimap a montré des résultats plus prometteurs. En effet, les premiers résultats non ajustés (au niveau de la taille de bloc) montrent un gain de plus de 20% sur le temps de traitement. Cette dernière s'améliore avec des tailles de bloc inférieures mais l'équilibre entre le gain en temps de traitement et la qualité de correction reste encore à définir. Un gros travail reste également à faire sur la partie prétraitement des données pour automatiser la création du fichier d'association en combinant multithreading et parallélisation. Une autre tâche consiste à trouver un moyen de réduire la taille du fichier d'association (cette modification devrait impacter là aussi le temps de traitement). La taille du fichier d'association étant le paramètre limitant de cette stratégie, la réduction de sa taille pourrait permettre d'élargir les applications sur des données plus volumineuse encore (ce qui est l'objectif de l'alternance).

D'autres alternatives et de nouveaux outils ont vu le jour et restent à tester. C'est le cas notamment de SRC Linker qui indexe un jeu de lectures et requête sur ce jeu de données (tout comme Compareads). Ce dernier utilise des structures pour stocker de manière optimale des données de chaque lecture et effectue un recrutement par comparaison de k-mer. Bien que les quelques tests réalisés montrent des performances exceptionnelles pour la recherche de lectures communes (1 minute sur des fichiers de 3.5 Go), de nombreux tests restent à réaliser pour vérifier si une implémentation peut être faites.

Références

- [1] Sanger F, Nicklen S, Coulson AR. **DNA sequencing with chain-terminating inhibitors.** Proc Natl Acad Sci USA, 1977 Dec.
- [2] Ross MG, Russ C, Costello M, Hollinger A, Lennon NJ, Hegarty R, Nusbaum C, Jaffe DB. **Characterizing and measuring bias in sequence data.** Genome Biol, 2013 May.
- [3] Thomas Hackl, Rainer Hedrich, Jörg Schultz, and Frank Förster. **large-scale high- accuracy PacBio correction through iterative short read consensus.** Bioinformatics (Oxford, England), 30(21) :3004–3011, 2014 Nov.
- [4] Leena Salmela and Eric Rivals. **accurate and efficient long read error correction.** Bioinformatics (Oxford, England), 30(24) :3506–3514, 2014 Dec.
- [5] James Gurtowski and Michael Schatz. **ectools, Long Read Correction and other Correction tools.** <https://github.com/jgurtowski/ectools>, 2014.
- [6] Goodwin S, Gurtowski J, Ethe-Sayers S, Deshpande P, Schatz MC, McCombie WR. **Oxford Nanopore sequencing, hybrid error correction, and de novo assembly of a eukaryotic genome.** Genome Res, 2015 Nov.
- [7] Madoui MA, Engelen S, Cruaud C, Belser C, Bertrand L, Alberti A, Lemainque A, Wincker P, Aury JM. **Genome assembly using Nanopore-guided long and error-free DNA reads.** BMC Genomics, 2015 Apr.
- [8] Nicholas J. Loman, Aaron R. Quinlan. **Poretools : a toolkit for analyzing nanopore sequence data.** Oxford University Press, 2014 Aug.
- [9] Kiełbasa SM, Wan R, Sato K, Horton P, Frith MC. **Adaptive seeds tame genomic sequence comparison.** Genome Ressource, 2011 Mar.
- [10] Ole Tange. **GNU Parallel : The Command-Line Power Tool.** The USENIX Magazine, 2011.
- [11] Kent WJ. **BLAT—the BLAST-like alignment tool.** Genome Ressource, 2002 Apr.
- [12] Maillet N, Lemaitre C, Chikhi R, Lavenier D, Peterlongo P. **Comparereads : comparing huge metagenomic experiments.** BMC bioinformatics 2012, 13 Suppl 19 :S10.
- [13] Deorowicz S, Kokot M, Grabowski S, Debudaj-Grabysz A. **KMC 2 : fast and resource-frugal k-mer counting.** BMC bioinformatics, 2015 Jan.
- [14] Ekaterina Starostina, Gaik Tamazian, Pavel Dobrynin, Stephen O'Brien, Aleksey Komissarov. **Cookiecutter : a tool for kmer-based read filtering and extraction.** BioRXiv, 2015 Aug.

- [15] Heng Li. **Minimap and miniasm : fast mapping and de novo assembly for noisy long sequences.** Bioinformatics, 2016 Jul.
- [16] Matei David, Lewis Jonathan Dursi, Delia Yao, Paul C Boutros, Jared T Simpson. **Nanocall : An Open Source Basecaller for Oxford Nanopore Sequencing Data.** BioRXiv, 2016 Mar.

RÉSUMÉ :

Le séquençage avec des technologies courtes lectures a permis l'étude du génome de nombreuses espèces. Cependant du fait de la longueur des lectures, l'assemblage des régions répétées reste compliqué. Pour pallier à ce problème, une génération plus récente de séquenceurs est apparue et notamment le MinION proposé par Oxford Nanopore Technologies. Celui-ci est caractérisé par sa possibilité de séquencer de longs fragments (8-20 Kb) mais également par les nombreuses erreurs qu'il produit (15%). Pour diminuer cet important taux d'erreur, le Genoscope a développé une stratégie de correction des lectures Nanopore à l'aide de lectures Illumina. Cette stratégie montre une très nette amélioration du taux d'identité mais reste limitée par son temps de traitement inadapté à de gros volumes de données. Pour améliorer cette stratégie, deux alternatives ont été proposées. La première, basée sur la combinaison de KMC2 et Cookiecutter s'est avérée encourageante au début puis rapidement décevante sur des jeux de données volumineux. La seconde, basée sur Minimap, a montré une diminution dans le temps de traitement de l'ordre de 20% sur des jeux de données actuellement étudiés. Ce travail d'amélioration de l'implémentation est toujours en cours afin de continuer de diminuer les temps de calcul.

The DNA sequencing with short reads technologies allowed the genome study of several species. However, due to read length, the assembly of repeated regions is difficult. Recently, a new generation of sequencers able to solve this assembly problem appeared with the Oxford Nanopore Technologies sequencer, the MinION. This MinION is able to sequence long fragments (8 - 20 Kb) but generates several errors (15%). In order to deal with those errors, the Genoscope developed a software able to correct Nanopore reads using Illumina ones. Although the Nanopore reads are corrected, the performance does not allow correction on large dataset. To improve this strategy, two alternatives have been proposed. The first one, based on the combination of KMC2 and Cookiecutter, despite the good results at the beginning turned out to be disappointing. Thus a second strategy, based on Minimap, was proposed. This strategy decreases the treatment time of 20% for currently studied data. This improvement work is still in process to decrease treatment times.

Nanopore, correction, assemblage, optimisation, NGS

MinION, NaS, Overlap Layout Consensus, Benchmarking, Parallélisation