Laboratoire de Bioinformatique pour la Génomique et la Biodiversité

Master de bioinformatique - ingénierie de plate-forme en biologie

Université de Paris

Rapport d'alternance

Gestion informatique des données de séquençage

---

28 janvier 2022

William Amory

sous la resposabilité de Frédérick Gavory

Table des matières

1 Introduction

1.1 LBGB au sein du Genoscope et du CEA

Le Genoscope (CNS[1]) a été créé en 1996 pour participer au projet mondial de séquençage du génome humain (*Human Genome Project*) qui à débuté en 1990 et c'est terminé en 2003. Il a notament participer au séquençage du chromosomes 14 humain. Le Genoscope participe égalment à développer des programmes de génomiques en france dans le cadre du projet France génomique. Aujourd'hui un des projet phare du Genoscope est le projet **Tara**, qui a pour objectifs l'étude des écosystèmes marins.

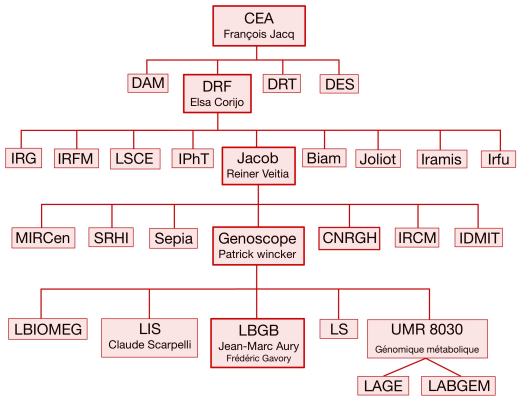

FIGURE 1 – Organigramme situant l'équipe du LBGB au sein du Genoscope et du CEA

Le Laboratoire de Bioinformatique pour la Génomique et la Biodiversité (**LBGB**) dirigé par Jean-Marc Aury, fait partie du Genoscope qui est une composante de l'institut François Jacob (Jacob) de la direction de la recherche fondamental (**DRF**) du Commissariat à l'Énergie Atomique et aux Énergies Alternatives (**CEA**), qui à été fondé le 18 octobre 1945 par Charles de Gaulle. L'intégration du génoscope au CEA a été réalisée en 2007, et en 2017 il devient une direction de l'institut François Jacob.

1.2 Contexte et missions du LBGB

Les missions qui sont confiées au LBGB sont de réaliser le contrôle qualité des données de séquences issues des différents séquenceurs, d'effectuer l'assemblage[2] des séquences et l'annotation[3] des génomes. Tout en permmettant la visualisation de chacune des missions (visualisation des annotations, de la qualité des reads[4], ect.). Il a egalement la mission de faire de la vielle technologique d'outils et méthodes permettant de réaliser ses autres missions. Le laboratoire est divisé en plusieurs groupes de travail. Le groupe *production* (dont je fais parti), le groupe *assemblage*, le groupe *annotation* et le groupe *évaluation des technologies de séquençage*.

Les misssions du groupe de *production* sont de développer, tester et maintenir les scripts dans l'objectif de répondre aux besoins des équipes de recherche et de séquençage en automatisant au maximum les processus. Notament dans la mise en place et au maintient de pipelines automatiques pour la génération des fichiers de séquences, le contrôle qualité et les analyses biologiques de ces derniers. Le groupe à également la mission de faire de la veille technologique et d'évaluer de nouveaux outils et méthodes pour chacune de ses autres missions.

1.3 Présentation du workflow NGS

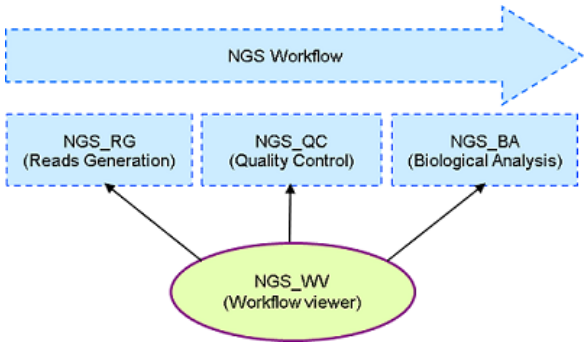

FIGURE 2 – Workflow de génération, de contrôle qualité et d'analyse biologique des fastq

Le workflow ngs est composé de trois pipelines pour la technologie Illumina. Le premier (ngs_rg), permet la génération des reads. Le second (ngs_qc), permet de réaliser le contrôle qualité. Le dernier (ngs_ba), permet de faire les analyses biologiques. Ces trois pipelines sont automatisés dans le workflow et permmettent de réaliser la disribution des données de séquences par projet, de les trier par échantillons, runs et technologies de séquençage. Ils réalisent aussi le nettoyage, l'analyse de ces fichier et mettent à jour la bases de données de référence ngl.

1.4 La technologie MGI

Le genoscope et le CNRGH[5] ont récement fait l'aquisition de séquenceurs MGI[6] (2 DNBSEQ-G400 et 1 DNBSEQ-T7).

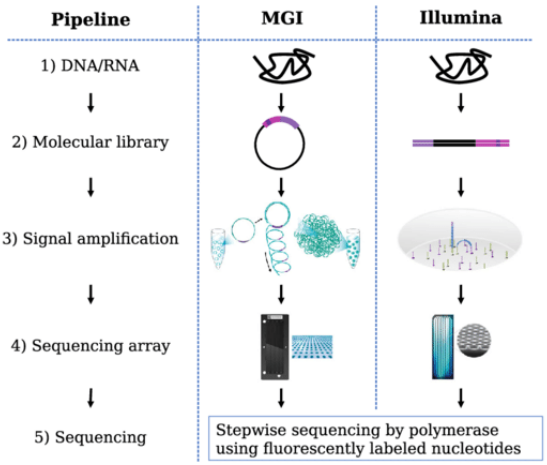

FIGURE 3 – Différences entre Illumina et MGI de technologie NGS

Il s'agit de séquenceurs à haut débit équavalent à un HiSeq 4000 (Illumina) pour le DNBSEQ-G400 et à un NovaSeq 6000 (Illumina) pour le DNBSEQ-T7. Les principales différences entre MGI et Illumina sont dans la création des librairies et dans la méthode d'amplification d'ADN. Les librairies sont double brins circualaire pour MGI, alors que pou Illumina elle est double brins linéaire. L'anplification ADN est réaliser en solution pour MGI puis déposé sur la Flowcell[7], alors que pour Illumina elle esr réaliser après immobilisation sur les Flowcell.

	DNBSEQ-G400	DNBSEQ-T7	HiSeq 4000	NovaSeq 6000
Max Number of Flow Cells	2	4	2	2
Max Lane/Flow Cell	4	1	4	4
Run Time	$\sim$ 14-37 h	$\sim$ 20-30 h	$\sim$ 24-84 h	$\sim$ 13-44 h
Max Reads Per Run	1.8 billion	5 billion	10 billion	20 billion
Max Read Length	2 $\times$ 200 bp	2 $\times$ 150 bp	2 $\times$ 150 bp	2 $\times$ 250 bp

TABLE 1 – Spécification des séquenceurs

2 Ojectifs

L'objectif principale de ma mission est la mise en place d'un workflow pour les séquenceurs de MGI. Plus précissément il s'agira de créer un pipeline de génération de reads (ngs_rg_mgi) et un pour le contrôle qualité (ngs_qc_mgi). Le workflow devra créer et mettre à jour l'états des run et des *lanes* dans ngl, réaliser le demultiplexage et faire un contrôle qualité des fastq à chaque étapes. De plus, il renommera les fichiers fastq et les déplacera dans le projet associé dans ngl. Il devra aussi déplacer les fichier de statistques et les fichiers fastq correspondant aux index non attendus dans leurs répertoire dédiés dans ngl.

Les autres objectifs de ma mission sont de réaliser des évaluations de nouveaux outils pour les différents pipelines mis en place pour les différentes technologies de séquençage. Tel que l'évaluation d'outils de génération de fichier fastq et de démultiplexage[8] en vu du remplacement de bcl2fastq par bcl-convert. Ainsi que de maintenir les pipelines des différentes technologies en ajoutant, remplaçant certains outils suite à une évaluation de ces derniers ; notament pour le workflow d'Illumina et celui de MGI une fois ce dernier créé.

3 Matériels et Méthodes

La Genoscope possède 7 cluster de calucls pour la *production* sur le serveur *inti*, ces derniers dispose de 16 coeurs et de 257 Go d'espace de stockage. l'accès à l'utilisation des clusters est réaliser par le logiciel Slurm[9]. Il dispose également de sa propre bases de données de référence (ngl). La gestion et le suivi du développememnt informatique est réaliser par le système Jira[10]. L'écriture du workflow des pipelines pour les séquenceurs MGI sera réaliser dans le langage de programmation Perl. L'utilisation de ce langage est rendu necessaire pour des raison historique du laboratoire, puisque de nombreuses librairies et modules qui seront à utiliser dans l'écriture des pipelines sont écrits en Perl. Le workflow pour MGI s'appuirera sur le workflow d'Illumina qui est totalement implémenté en Perl.

C'est pour toutes ses raisons qu'il m'a été nécessaire d'apprendre à coder en Perl en réalisant un programme permettant de faire des analyses statistiques élémentaire sur des fichiers fastq. Tel que le taux de GC, la moyenne du score de la qualité, ainsi que plusieurs autres métriques. Le programme est capable de gérer les fichiers fastq issue de séquençage *single end* et *paired end*. Cela m'a permis de prendre en main les modules utilisé pour les différents pipelines déja en place, notament pour le pipelines d'llumina. De plus cela ma permi de prendre en main de l'environement de travail, l'utilisation du lancement de job sur les noeuds de calculs et l'utilisation des modules utilisé pour le workflow d'llumina

Une première évaluation d'outils à également été efféctué en vu du remplacement de bcl2fastq par bcl-convert. Il s'agit de deux logiciels de génération de fichiers fastq et de démultiplexage qui sont developpés et commercialisés par Illumina.

Dans un premier temps, il a été necessaire de déterminer les conditions optimales (temps total rapide, pourcentage d'utilisation cpu[11] maximum) en fonction des ressource disponible

sur les cluster pour la *prodution* (16 coeurs maximum) sur le nouveau serveur (*inti*), pour bcl2fasq dans l'objectif de pouvoir comparer les performances des deux logiciels dans les mêmes condtions. Les conditions optimales sont déterminées en fonction des paramètre suivant de bcl2fatq :

- `r` : nombre de *threads* accordé pour la décompréssion et la lecture des *Bases Calls*
- `p` : conversion des *Bases Calls* en fastq
- `w` : écriture et compréssion des fichier fastq

Tous ces tests ont été réalisés sur le même noeud de calcul, dans l'objectif de minimiser les biais. La comparaison est effectué sur le temps total de génération des fastq et le démultiplexage, ainsi que le temps cpu et le pourcentage d'utilisation des cpu.

4 Résultats des évaluations de bcl2fastq et bcl-convert

4.1 Détermination des meilleurs paramètres pour bcl2fastq

Après avoir effectué différentes combinaisons des paramètres, nous avons mis en évidence que la variation du paramètre `r` et `w` en fixant le paramètre `p`, n'apportait pas de différences significatives. Nous avons donc fait varier les paramètres `p`, `r` et `w` de manière à ce que chacun des paramètre soit égale au nombre de coeurs accordé aux deux logiciels.

4.2 Comparaison entre bcl2fastq et bcl-convert

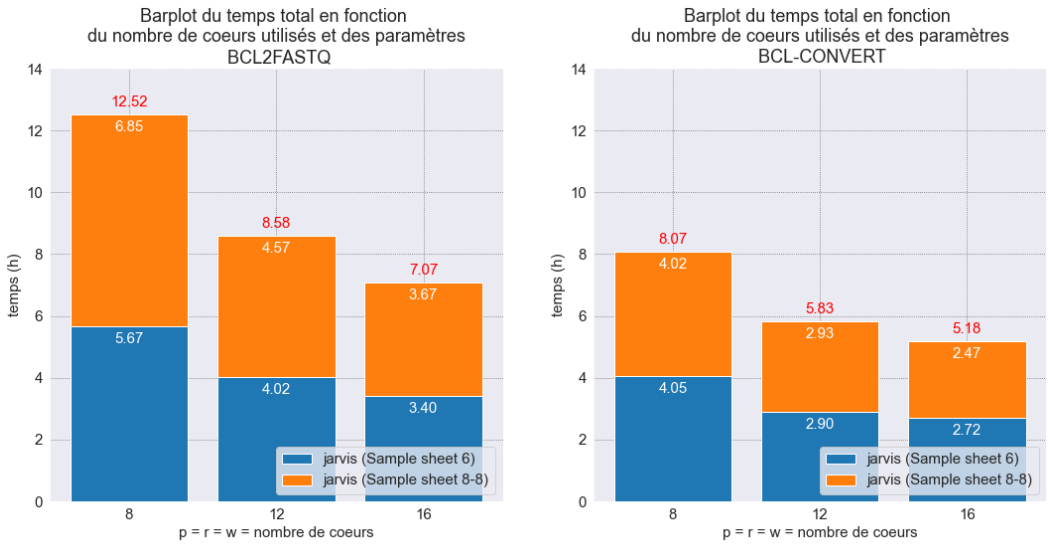

FIGURE 4 – Temps total de génération des fastq pour bcl2fastq et bcl-convert

La figure **??**, montre la différence de temps total des deux logiciels. Il y a deux *sample sheet*, car le nombre de bases considérés des *reads index* entre les *lanes* est différents, obligeant à réaliser deux appelle différents au logiciels pour générer les fastq et le démultiplexage. On oserve bien que plus on augmente le nombre de coeurs pour chacun des logiciels plus la génération des

fastq et le démultiplexage est rapide. De plus on remarque que bcl-convert permet de réduire le temps d'environs 1/3 par rapport à bcl2fatq.

5 Perspectives

Concernant les perspectives de bcl-convert il reste à réaliser un cahier des charges référençant tous les changements à effectuer dans les différents pipeplines pour la mise à jour de bcl2fastq vers bcl-convert. Ce cahier des charges prendera en compte le changement d'arborescence des fichier de sortie entre les deux logiciels, ainsi toutes les modifications à effectuer dans les différents pipelines pour permettre le bon fonctionnement des workflow. Dû à la pression actuelle autour de la technologie MGI, c'est un autre dévelopeur qui ce chargera de suivre ce cahier des charges et de réaliser les modifications.

Concernant le workflow de MGI, il nous faut dans un premier temps déterminer les outils et méthodes necessaires (utilisation de ceux du workflow d'Illumina ou de nouveaux). Une fois ceci déterminer il restera à écrire les deux pipeline,s celui de génération de reads (ngs_rg_mgi) et celui de contrôle qualité (ngs_rg_mgi). L'objectif sur le long terme est d'arriver à un workflow totalment automatisé, comme celui d'Illumina.

Il y a aussi l'évaluation d'autres outils utiles pour les pipelines, comme l'évaluation d'outils de *trimming*, *filtering*, d'assignation taxonomique, ect.

5.1 diagramme de gantt

Notes

[1]Centre National de Séquençage

[2]reconstruction d'un génome à partir fragment de ce dernier

[3]Documenter le plus exhaustivement possible les informations de l'assemblage permmettant de prédire la fonction d'une molécule

[4]Lecture d'une séquence par un séquenceur d'ADN d'un fragments d'ADN.

[5]Centre National de Recherche en Génétique Humaine

[6]membre du groupe BGI dont les missions sont : R&D, production et vente d'instruments de séquençage d'ADN, de réactifs et de produits connexes

[7]Lame d'absorbtion des fragment d'ADN et cuve réacteur du séquençage

[8]Séparation des différents reads d'une *lane* en fonction de l'index d'échantillon

[9]Logiciel open source d'ordonnancement des tâches informatiques

[10]Logiciel de gestion de projet, d'incidents et de suivi de bugs

[11]Central Processing Unit (unité centrale de traitement, en français)