



HAL
open science

Identification and quantification of microbial strains in metagenomic samples using variation graphs

Kévin da Silva

► **To cite this version:**

Kévin da Silva. Identification and quantification of microbial strains in metagenomic samples using variation graphs. Bioinformatics [q-bio.QM]. Université de Rennes 1, 2022. English. NNT: . tel-03896860

HAL Id: tel-03896860

<https://hal.science/tel-03896860>

Submitted on 13 Dec 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE DE DOCTORAT DE

L'UNIVERSITÉ DE RENNES 1

ÉCOLE DOCTORALE N° 601
*Mathématiques et Sciences et Technologies
de l'Information et de la Communication*
Spécialité : *Informatique*

Par

Kévin DA SILVA

Identification and quantification of microbial strains in metagenomic samples using variation graphs

Thèse présentée et soutenue à Jouy-en-Josas, le 8 mars 2022, avec le soutien de la Région Bretagne et du métaprogramme HoloFlux de l'INRAE
Unité de recherche : Institut de Recherche en Informatique et Systèmes Aléatoires, UMR 6074

Rapportrices avant soutenance :

Hélène TOUZET Directrice de recherche, CNRS, CRISAL Lille
Claudine MEDIGUE Directrice de recherche, CNRS, Genoscope, Evry

Composition du Jury :

Président :	Eric PELLETIER	Directeur de recherche, CEA, Evry
Rapportrices :	Hélène TOUZET	Directrice de recherche, CNRS, CRISAL Lille
	Claudine MEDIGUE	Directrice de recherche, CNRS, Genoscope, Evry (absente)
Examineurs :	Eric PELLETIER	Directeur de recherche, CEA, Evry
	Eric RIVALS	Directeur de recherche, CNRS, LIRMM, Montpellier
Directeur de thèse :	Pierre PETERLONGO	Chargé de recherche, Inria, Rennes
Co-encadrants de thèse :	Nicolas PONS	Ingénieur de recherche, INRAE, MetaGenoPolis, Jouy-en-Josas
	Magali BERLAND	Ingénieure de recherche, INRAE, MetaGenoPolis, Jouy-en-Josas

Invité(s) :

Florian PLAZA OÑATE Ingénieur de recherche, INRAE Transfert, MetaGenoPolis, Jouy-en-Josas

ACKNOWLEDGEMENT

À mes parents, qui ont toujours eu la grandeur d'âme et d'esprit de m'encourager à donner le meilleur de moi-même sans jamais tomber dans le piège de la pression à faire des études pour faire des études. Merci d'avoir toujours fait passer mon bonheur et mon bien-être avant tout. Merci pour la confiance et la fierté dont vous faites preuve. À toute ma famille, pour leur soutien et encouragements. À tous mes plus proches amis pour avoir supporté mes doutes et plaintes, mais aussi partagé les petites victoires.

La concrétisation de ce travail de thèse n'est pas seulement née des trois ans et demi qui ont composé le contrat doctoral, c'est aussi le résultat de la convergence de plusieurs événements et rencontres qui ont façonné le chemin parcouru. Ainsi, mes remerciements ne peuvent se faire que de manière chronologique.

Tout d'abord je souhaite remercier mes enseignants du Master de bioinformatique de l'Université Paris-Saclay. Leur bienveillance et leur passion ont été communicatives et ont grandement contribué à mon épanouissement dans cette discipline. Jamais je n'ai autant pris plaisir à étudier dans l'enseignement supérieur que pendant ces deux années. Merci également pour votre confiance et vos encouragements puisque déjà à cette époque vous avez été plusieurs à me rassurer sur mes qualités et compétences pour poursuivre en thèse. J'ai tenté de faire résonner ces paroles lors des moments de doutes et de difficultés. Merci également à la promotion étudiante pour la camaraderie et l'entraide que vous avez fait régner. En particulier Amila Malinovic, merci pour ton amitié et les binômes que nous avons formé pendant ces années. Mention spéciale également pour Audrey Duval, dont la ténacité et la rigueur ont été source d'inspiration et le sont encore aujourd'hui, tu mettais la barre haute ce qui a indéniablement façonné mes standards d'exigence.

Merci à Ariane Bassignani, ma camarade INFJ, non seulement pour les années Master auxquelles tu as apporté une richesse sociale, mais aussi pour avoir probablement été le déclencheur principal de tout ce qui a suivi. Entre autres pour avoir diffusé l'annonce de recrutement à MetaGenoPolis et vanter mes mérites. Et suite à cela, ma reconnaissance va à Nicolas Pons, Magali Berland et tous les autres permanents de MetaGenoPolis qui ont accepté de me faire confiance et m'ont offert ma première expérience professionnelle en

tant qu'ingénieur d'études. Votre bienveillance et vos qualités professionnelles ont rendu ce premier pas dans la vie active inoubliable et je pense que je n'aurais pu souhaiter meilleure atmosphère professionnelle. C'est aussi dans les meilleurs moments qu'on se rend compte de l'importance de notre environnement même dans un cadre purement professionnel. J'ai joui d'une immense et précieuse liberté à échanger des idées avec les autres collègues, à travailler sur plusieurs projets ou encore à suivre des formations comme je le souhaitais. Sans pouvoir tous les citer, Sébastien Fromentin, Susie Guilly, Victoria Meslier, Florence Thirion, Mathieu Almeida, Manolo Laiola et Laurie Alla, vous avez été formidables.

Merci à Samar Berreira Ibrahim. Puis-je encore trouver d'autres mots que ce que j'ai déjà dit pour souligner ton importance ? Probablement pas, alors je me contenterai de te faire honneur dans un paragraphe à part. De la nouvelle stagiaire que je trouvais bien extravertie et bien familière envers moi au bout de seulement deux jours de stage, si on m'avait dit qu'elle serait en réalité un binôme de trajet du matin et du soir, un binôme de discussions personnelles et professionnelles, ou encore un binôme de formations. Merci d'être aussi solaire, merci d'être aussi sociable, et merci de m'avoir accordé une place. Ton amitié a été et est précieuse, notamment car tu sais que tu fais partie de ces personnes (si rares) capables de parler « avec toi » et pas simplement « à toi ». Je te souhaite le meilleur pour la suite.

Je renouvelle ma gratitude envers Nicolas Pons et Magali Berland qui, après m'avoir accepté en tant qu'ingénieur, m'ont proposé de poursuivre en thèse. J'en garderai toujours une profonde reconnaissance, tant cela a été valorisant pour moi et mes compétences, et tant ça m'approchait un peu plus de mes objectifs professionnels. Avec Florian Plaza Oñate, je vous remercie de m'avoir encadré ces quelques années, votre bienveillance et votre soutien ont été indispensables. Merci de n'avoir rien lâché même dans mes moments où j'étais le plus borné ou pessimiste.

À Pierre Peterlongo, mon directeur de thèse. Je mesure grandement la chance que j'ai eu de partager cette expérience professionnelle avec lui. Je pense que jamais encore je n'avais rencontré une personne aussi intègre, à la fois bienveillant et exigeant, sérieux et drôle, débordant d'énergie, d'idées et d'optimisme. Merci de m'avoir montré la voie, je ne doute pas que pas mal de choses que tu m'as apportées continueront de résonner en moi et me serviront professionnellement et personnellement. Malgré les difficultés, la thèse restera une expérience positive, en grande partie grâce à toi. Tes qualités de leader, tel que je le conçois, sont une véritable source d'inspiration et de respect.

Cette expérience de thèse a également été l'occasion de découvrir une nouvelle ville et une nouvelle équipe. Merci à GenScale et plus globalement Symbiose d'avoir créé un cadre si propice aux échanges et au bien-être professionnel des gens, notamment des doctorants. Merci à tous les doctorants de Symbiose, le partage d'expérience et le soutien entre nous ont été des plus réconfortants. Parmi les « anciens », je tiens particulièrement à remercier Méline Wery. Que tu m'as manqué après la fin de ta thèse. Tu as sans doute été la personne dont les mots ont été les plus rassurants. Et je n'oublierai pas non plus nos conversations plus légères et notre brillant duo au karaoké.

Merci à Nicolas Guillaudeau, mon camarade de promotion de doctorants. Tu as été un coup de foudre amical quasi-instantané, mon premier ami à Rennes, et d'une gentillesse et d'un soutien indéfectible pendant cette thèse. Merci pour les parties de cartes (quand l'innocence de début de thèse nous en laissait encore le temps !), les escape games et tout le reste qui a rendu la vie au laboratoire et à l'extérieur si riche. Merci aussi à Camille Juigné, quel bonheur que tu sois restée à Symbiose, d'abord en stage, puis en ingénieure et maintenant en thèse. J'aurais alors raté tellement de choses, tellement de discussions sur le féminisme, la bienveillance, la communication non-violente, et aussi tellement de jolies sorties. Merci d'avoir autant égayé ma vie rennaise, de m'avoir autant enrichi personnellement par nos discussions. Et puis surtout, sans toi, il n'y aurait sans doute pas eu non plus le taekwondo. Merci aussi à Ludovic Fourteau, ton humour et ta sociabilité ont également beaucoup comptés. Et merci à Diane Dewez, qui n'était pas de Symbiose mais que j'ai eu le plaisir de rencontrer via ses expérimentations en réalité virtuelle et au badminton. Tu as également été fort enrichissante, merci pour nos thérapie-café, et les discussions autour de l'introversion et la confiance en soi.

En plus d'être soutenue par ses encadrants, la thèse profite de l'appui de personnes extérieures. Je tiens donc à également remercier tous les membres de mon comité de suivi de thèse pour leur écoute et leurs conseils avisés : Rayan Chikhi, David Vallenet, Olivier Jaillon et Mahendra Mariadassou.

Enfin, un immense merci à l'ensemble des membres du jury de thèse de nous faire l'honneur de s'intéresser à notre travail et d'avoir accepté de prendre de leur temps pour cela : les rapportrices Claudine Médigue et Hélène Touzet, ainsi que les autres membres en plus de mes encadrants, Eric Rivals et Eric Pelletier.

Bonne lecture !

RÉSUMÉ DE LA THÈSE

Introduction

Les microorganismes, et en particulier les bactéries, sont prépondérants dans de nombreux écosystèmes, dont le corps humain. Les technologies de séquençage haut-débit ont joué un rôle majeur dans leur caractérisation, notamment en métagénomique, en donnant accès à la totalité de leur information génétique et à leur potentiel fonctionnel. Les études qui s'intéressent aux communautés microbiennes dans le domaine de la santé humaine par exemple, ont le plus souvent pour objectif d'identifier des biomarqueurs signatures en associant des caractères phénotypiques à l'abondance de gènes ou d'espèces bactériennes dans une population. Cependant, un niveau de résolution plus fin est désormais nécessaire, ces signatures pouvant être indétectables en restant à l'échelle de l'espèce car en réalité dépendantes de souches microbiennes spécifiques. L'étude des communautés microbiennes à l'échelle de la souche présente donc un grand intérêt et soulève de nombreux défis bioinformatiques.

Ce travail de thèse se propose d'explorer ce domaine d'étude grandissant et ses thématiques associées telles que la représentation de multiples séquences génomiques par des graphes, ainsi que de proposer une nouvelle solution bioinformatique aux défis actuels.

Concepts biologiques et approches méthodologiques

Le développement des technologies de séquençage nouvelle génération, qui permettent de lire les séquences ADN, est à l'origine de l'essor des domaines tels que l'étude des maladies génétiques ou la médecine de précision, grâce au haut-débit des séquences que ces machines peuvent traiter. Parmi les technologies les plus utilisées, le séquençage Illumina qui produit des lectures de séquençage courtes, environ 150 paires de bases (pb), reste dominant (Goodwin et al., 2016). Le séquençage Illumina permet également de produire des lectures appariées, offrant une plus grande couverture horizontale du fragment séquencé. Les lectures de séquençage sont par la suite assemblées pour reconstruire tout

(génomome complet) ou partie (contigs) d'un génomome de référence. Ces génomomes de référence servent de base à de nombreuses études et développements en bioinformatique, dont le travail de thèse présenté.

Cette thèse se focalise sur l'étude des bactéries. Leur génomome est généralement composé d'un unique chromosome, mais de l'information génétique peut également être portée par des plasmides dont le nombre est variable. Qu'il s'agisse du chromosome ou des plasmides, la structure ADN est généralement circulaire. Plusieurs méthodologies, notamment sur des caractères morphologiques, ont été utilisées pour classifier les bactéries ainsi qu'établir une taxonomie. Les technologies de séquençage ont justement apporté de nouvelles informations pour raffiner ces classifications, basées cette fois sur des caractéristiques génétiques. Le nombre croissant de séquences bactériennes disponible a permis par des études de génomique comparative de mettre en évidence la diversité qui existe au sein d'une même espèce bactérienne (Fraser-Liggett, 2005; Tettelin et al., 2005; Lefébure and Stanhope, 2007), conduisant à une nouvelle échelle taxonomique que sont les souches. Cependant, encore aujourd'hui, il n'existe pas de définition consensus pour une souche. En 1995, Tenover *et al.* propose de définir une souche comme « un isolat ou groupe d'isolats pouvant être distinguer d'autres isolats du même genre ou de la même espèce par des caractères phénotypiques, des caractères génétiques, ou les deux » (Tenover et al., 1995).

En parallèle de la notion de souche, l'observation des variations génomiques au sein d'une même espèce a introduit le concept de *pangenome*. Le pangenome est l'ensemble des gènes retrouvés chez une espèce, et est généralement divisé en deux grandes catégories. Tout d'abord le *génomome cœur*, qui correspond à l'ensemble des gènes retrouvés chez toutes les souches de l'espèce étudiée. Et le *génomome accessoire*, qui correspond aux gènes présents chez une seule souche ou un nombre limité de souches (Tettelin et al., 2005). Cependant, d'autres travaux ont raffiné cette catégorisation en s'intéressant au pangénomome à des échelles taxonomiques plus élevées (Makarova et al., 2007; Koonin and Wolf, 2008). Les études en pangénomique s'intéressent essentiellement à caractériser la composition en gènes d'une souche ou encore l'impact des transferts horizontaux de gènes d'un point de vue évolutif. Bien que ce travail de thèse n'emploie pas à proprement parler une approche pangénomique, il existe un lien clair avec les questions que soulève une résolution à l'échelle de la souche. Cet aspect sera notamment re-détaillé dans les perspectives.

Si la génomique peut être élargie au concept de pangénomique à l'échelle d'une espèce, à l'échelle d'une communauté d'espèces elle est élargie au concept de *métagénomique*. En métagénomique, un échantillon environnemental est séquencé, révélant les génomes de tous les individus présents dans l'échantillon. Dans un écosystème microbien, l'ensemble des microbes retrouvés est appelé *microbiome* et l'ensemble des génomes de ces microbes est le *métagénome*. En particulier, le séquençage métagénomique *shotgun*, qui consiste à décomposer aléatoirement l'ADN en plusieurs fragments, permet (en théorie et avec une profondeur de séquençage suffisante) une couverture quasi-exhaustive de toutes les espèces présentes ainsi que leurs variations génétiques, des informations indispensables pour permettre des analyses allant jusqu'à une résolution au niveau des souches.

Les analyses en métagénomique s'intéressent à différentes questions reflétant les diverses caractéristiques d'une communauté microbienne, en particulier sa composition, sa dynamique, ou encore les réseaux métaboliques impliqués. Dans cette thèse, on s'intéressera à la question de la composition de la communauté, c'est-à-dire à mettre en évidence les souches présentes dans un échantillon, faisant ainsi le lien avec les intérêts à long-terme présentés en introduction que sont la découverte de biomarqueurs. La section suivante présente le contexte qui existe déjà, au niveau espèce, dans cet objectif d'analyse de la composition, puis les outils développés pour des analyses au niveau souche seront abordés dans l'état de l'art.

Contexte scientifique et objectifs

Dans le but de répondre à la question de la composition d'un échantillon métagénomique au niveau espèce, deux principales approches sont utilisées : les approches basées sur des références, qui sont particulièrement adaptées pour identifier des espèces présentes dans une base de données de référence, et les approches sans référence, qui sont utiles lorsqu'aucune espèce proche n'est disponible (Comin et al., 2021).

Une méthode d'analyse classique, et qui a contribué à poser le contexte de ce projet de thèse du fait de ses limitations, repose sur l'utilisation d'un catalogue de gènes représentatif de l'écosystème étudié. Cette approche utilise à la fois des éléments des méthodes basées sur des références et des méthodes sans référence. Brièvement, et pour prendre l'exemple de la construction du catalogue de 3.3 millions de gènes du microbiote intestinal (Qin et al., 2010), des échantillons fécaux humains ont été séquencés puis les lectures de

séquençages assemblées en contigs. À partir des contigs, les gènes ont été extraits à l'aide d'outils de prédiction de gènes puis regroupés par similarité en familles de gènes. Afin de réduire la redondance, un seul gène au sein d'une famille de gènes est sélectionné comme représentant. Cette étape de réduction de la redondance élimine alors les variations qui auraient été indispensables à l'identification de souches bactériennes. Ces catalogues de gènes peuvent ensuite être utilisés pour reconstruire des *espèces métagénomiques* (Nielsen et al., 2014), et plus récemment des *pangénomomes d'espèces métagénomiques* (Oñate Plaza et al., 2019), distinguant les gènes cœur des gènes accessoires, un premier pont vers la pangénomique et vers l'échelle des souches.

Étant donné l'intérêt grandissant et la nécessité de procéder à des analyses métagénomiques au niveau souche, ce projet de thèse propose un nouveau cadre d'exploration permettant un profilage individuel d'échantillons métagénomiques, c'est-à-dire identifier et quantifier les souches dans une communauté bactérienne d'un écosystème donné ainsi qu'inférer de nouvelles souches.

Les catalogues de gènes sont construits à partir de sous-ensembles d'échantillons tandis que de nouveaux sont sans cesse disponibles. La prise en compte de ces nouveaux échantillons implique de mettre à jour ces catalogues. Les catalogues de gènes sont donc des bases de données figées qui nécessitent de repasser par toutes les étapes de construction pour constituer de nouveaux catalogues mis à jour. De cette limite naît alors une première nécessité d'avoir une structure de données pouvant être mise à jour de manière dynamique. C'est pourquoi nous nous sommes tournés vers les structures de graphe, qui permettent cette fonctionnalité. L'état de l'art ci-dessous détaille les approches par graphe.

De plus, une autre limite déjà abordée est la suppression de variations lors de la sélection du gène représentant de la famille de gènes. À nouveau, les graphes semblent être une structure adéquate pour représenter un ensemble de séquences similaires, et ainsi conserver toutes les variations apportées par toutes les séquences.

Les objectifs principaux de la thèse sont donc (i) d'utiliser une structure plus globale, pouvant être requêtée et dynamique, appliquée à la métagénomique (ii) identifier et indexer les gènes de souches bactériennes (iii) calculer les abondances des souches présentes dans l'échantillon et prédire la présence de nouvelles souches.

Bien qu'ils ne répondent pas entièrement aux objectifs de la thèse, d'autres outils ont déjà été développés dans le but de s'intéresser aux souches bactériennes et sont détaillés dans l'état de l'art.

Etat de l’art

Les graphes ne sont pas une nouvelle structure de données en bioinformatique, ils sont notamment utilisés par les outils d’assemblage. Un graphe est une structure composée de nœuds reliés par des arêtes. Une succession de nœuds liés par des arêtes est communément appelée un chemin. Dans le cas de l’assemblage, ce sont généralement les graphes de de Bruijn qui sont utilisés. Dans cette structure, les nœuds contiennent des k -mers des séquences d’origine, et les arêtes relient les k -mers se chevauchant sur $k - 1$ bases. Cependant, les graphes sont désormais également utilisés pour représenter les génomes de multiples individus, permettant alors de capturer toutes les variations au sein d’une espèce tout en fusionnant les régions identiques entre ces différents génomes. Ce sont les graphes de génomes, une sous-catégorie de graphe de séquences dans lesquels les nœuds contiennent des sous-séquences des séquences d’origine et les arêtes décrivent la succession non-chevauchante de ces différents segments. Étant donné les limites détaillées dans la section précédente et les graphes ayant gagné en intérêt ces dernières années pour représenter des génomes, ils semblent être un meilleur cadre d’analyse pour répondre à ces défis.

Plusieurs outils utilisant des représentations sous forme de graphe ont déjà été développés. Cependant, ils sont généralement utilisés dans des buts très spécifiques qui ne sont pas adaptés aux objectifs de cette thèse. **Cortex** (Iqbal et al., 2012) est utilisé pour de l’assemblage *de novo* ainsi que la découverte et le génotypage de variations génétiques. **BayesTyper** (Sibbesen et al., 2018) et **GraphTyper2** (Eggertsson et al., 2019) sont des outils de génotypage pour petites variations et les variants de structure. **Pandora** (Colquhoun et al., 2021) est spécifiquement dédié au génotypage des variants de structure. Bien que ces outils démontrent l’activité de la recherche dans le domaine et l’utilité des graphes pour représenter des séquences génomiques, dans le cadre de cette thèse, il est nécessaire de se tourner vers des développements de *frameworks* complets.

HISAT2 (Kim, Paggi, et al., 2019) crée un graphe linéaire à partir d’un seul génome de référence puis incorpore les variations en tant que chemins alternatifs. Cependant, **HISAT2** n’a pas été pensé pour représenter de manière exhaustive les variations retrouvées chez une espèce, ce qui est limitant pour une analyse au niveau souche telle que recherchée dans cette thèse. **Minigraph** (H. Li, Feng, et al., 2020) s’intéresse essentiellement à conserver un système de coordonnées stable dans le graphe comme cela est possible avec une représentation linéaire. Cependant, **minigraph** est plutôt adapté pour des analyses à l’échelle d’une population d’une part, et pour représenter des variants de structure

d'autre part, ce qui là encore est limitant pour les objectifs de la thèse. Au final, c'est `vg toolkit` (Garrison, Sirén, et al., 2018) qui a semblé l'outil le plus adéquate. Il permet de construire des graphes de variations, c'est-à-dire un graphe de génome bidirectionnel (les nœuds peuvent être lus dans les deux sens, correspondant ainsi aux sens de chacun des brins d'ADN) dont les génomes utilisés en entrée sont des chemins colorés dans le graphe. Contrairement aux autres outils mentionnés, les graphes de variations permettent de représenter sans perte toutes les séquences et toutes les variations. En plus de la construction de graphe, d'autres outils font partie de `vg toolkit` ou ont été développés pour fonctionner avec. `Seqwish` (Garrison, 2022) utilise les alignements de séquence deux à deux obtenus via `minimap2` pour construire un graphe de variations. Aussi, `vg toolkit` intègre son propre outil de mapping de courtes lectures de séquençage sur graphe.

En parallèle, des outils ont également déjà été développés afin de s'intéresser à la question de l'identification de souches bactériennes dans des échantillons métagénomiques.

`DESMAN` (Quince et al., 2017) utilise comme référence des gènes cœur d'espèces microbiennes connus et présents en une seule copie. À partir de ces références et d'un ensemble d'échantillons métagénomiques, `DESMAN` reconstitue des haplotypes de souches et en prédit les abondances. Cette approche ne s'applique pas au projet de thèse puisque l'un des objectifs est de pouvoir analyser les échantillons de manière individuelle. `StrainPhlan` (Truong et al., 2017) utilise des génomes de référence ou des marqueurs spécifiques dans les références, ainsi qu'un ensemble d'échantillons. Cependant, `StrainPhlan` n'est capable d'identifier que les souches dominantes et ne propose pas d'abondance. `StrainEST` (Albanese and Donati, 2017) et `DiTASiC` (Fischer et al., 2017) utilisent un ensemble de génomes de référence et permettent l'identification et la quantification de souches parmi les références. Enfin, `mixtureS` (X. Li et al., 2020) utilise un seul génome de référence à partir duquel il infère et quantifie le nombre de souches différentes (sans les identifier) présentes dans un échantillon donné.

En intégrant les deux aspects principaux de cet état de l'art, il apparaît, à notre connaissance, qu'aucun outil n'utilise les graphes pour représenter de multiples séquences dans le but d'être ensuite utilisés pour réaliser un profilage complet d'échantillons métagénomiques, c'est-à-dire identifier les souches présentes parmi les références disponibles, inférer de nouvelles souches et estimer leurs abondances. Afin de répondre à ces objectifs, j'ai développé `StrainFLAIR`.

StrainFLAIR: Profilage d'échantillons métagenomiques au niveau souche en utilisant des graphes de variations

Notre outil **StrainFLAIR** (STRAIN-level proFiLing using vArIation gRaph) a fait l'objet d'une publication scientifique (Da Silva et al., 2021). Il exploite plusieurs outils existants et propose de nouvelles solutions algorithmiques dans le but d'indexer des génomes bactériens au niveau souche ainsi que de les requêter.

Pour la partie indexation, **StrainFLAIR** utilise *Prodigal* (Hyatt et al., 2010) afin de prédire les gènes présents dans un ensemble choisi de génomes de référence. Puis, les gènes sont regroupés en familles de gènes à l'aide de CD-HIT (W. Li and Godzik, 2006). Pour chaque famille de gènes, les séquences des gènes sont alignées deux à deux via `minimap2`, dont le résultat est converti en graphe de variations grâce à `seqwish`. Les chemins colorés du graphe correspondent ainsi à des gènes.

Pour la partie requête, les lectures de séquençage d'un échantillon métagenomique sont alignées sur le graphe en utilisant un des outils de mapping de `vg toolkit`. Le résultat de ce mapping produit des alignements eux-mêmes sous forme de graphe qui a nécessité que nous développons un algorithme pour (i) récupérer les meilleurs alignements sous forme linéaire et (ii) attribuer les lectures aux chemins colorés du graphe en fonction de ces alignements. À partir des lectures couvrant les chemins colorés du graphe, une abondance est calculée pour chacun des chemins colorés. Enfin, l'abondance de la souche est estimée à partir de la moyenne des abondances des gènes spécifiques de la référence. Néanmoins, cette abondance est automatiquement nulle si la proportion de gènes spécifiques d'une souche ne dépasse pas un certain seuil défini par l'utilisateur.

La stratégie de **StrainFLAIR** a été validée sur des données simulées et sur un mock (jeu de données réel mais contrôlé). J'ai ainsi démontré la capacité de notre méthode à identifier et de correctement estimer l'abondance de souches bactériennes dans des échantillons métagenomiques. De plus, dans une des expériences incluant dans le mélange métagenomique une souche absente des références utilisées, j'ai pu mettre en évidence la présence de cette nouvelle souche et estimer son abondance relative. À l'inverse, une des expériences menait à l'indexation de plusieurs souches absentes de l'échantillon requêté, ainsi j'ai démontré que **StrainFLAIR** n'identifiait pas de faux positifs. Les résultats obtenus par **StrainFLAIR** ont été principalement comparés aux résultats de **Kraken2**. **StrainFLAIR**, dans le cas d'un

mélange de plusieurs souches, a montré une meilleure estimation des abondances relatives.

Vers la résolution des chemins ambigus et l'inférence de souches

Suite à la première version de **StrainFLAIR** et la publication de son article associé, j'ai considéré deux nouveaux développements immédiats.

Le premier nouveau développement fut la prise en compte des lectures de séquençage à extrémité appariée. En effet, dans la première version de **StrainFLAIR**, les lectures, même à extrémité appariée, étaient considérées comme des singletons. Or, ce type de lectures permettent une plus grande couverture horizontale des séquences et peuvent aider à lever des ambiguïtés. Dans le cas de notre approche, des ambiguïtés étaient observées durant l'étape d'attribution des lectures à des chemins colorés. Les lectures peuvent alors correspondre avec plusieurs chemins colorés, entraînant une distribution de la lecture entre ces différents chemins colorés pour le calcul des abondances. Or, il est probable que certains de ces chemins colorés associés à la lecture ne soient des faux positifs, introduisant alors du bruit dans le calcul des abondances. J'ai donc mis en place une méthode prenant en compte les attributions de chemins colorés de chaque lecture d'une paire pour ne conserver que les informations cohérentes entre les deux lectures.

Cette nouvelle méthodologie a été validée sur des jeux de données simulés et a démontré une nette amélioration des proportions de gènes détectés et des estimations des abondances relatives dans le cas d'un mélange ne contenant que des souches indexées dans le graphe de référence. Les résultats restaient corrects pour un mélange contenant une souche inconnue, avec à la fois une amélioration des résultats d'abondance pour certaines souches et une légère détérioration pour d'autres. Cette nouvelle méthodologie a également été validée sur un mock, sans amélioration ni détérioration des résultats, sauf pour la souche *Thermotoga* sp. RQ2 au niveau de la proportion de gènes spécifiques détectés. *Kraken2* proposait des résultats laissant suggérer la présence de cette souche dans l'échantillon et la première version de **StrainFLAIR** détectait presque 50% (le seuil choisi au-delà duquel une souche est considérée présente) de gènes spécifiques détectés, alors même que *Thermotoga* sp. RQ2 était censée être absente de l'échantillon d'après les données théoriques. Avec cette nouvelle version, la proportion de ces gènes détectés tombe à environ 20%, confirmant l'absence de cette souche et levant une certaine ambiguïté.

Le second nouveau développement fut la prise en compte des lectures de séquençage qui ne s'alignaient avec aucun chemin coloré du graphe de référence mais qui s'alignaient néanmoins sur un chemin (non coloré) du graphe. Ces lectures apportent des informations sur les nouvelles souches présentes dans un échantillon et non indexées dans le graphe. J'ai donc mis en place un algorithme qui définit des lectures de séquençage compatibles ou incompatibles entre elles selon leur chevauchement et qui estime le nombre minimum attendu de nouvelles souches compte tenu de la structure d'un graphe décrivant les incompatibilités entre lectures, et ce, à l'échelle de chaque famille de gènes. Cet algorithme inclut également une étape de filtrage des lectures, car en plus des lectures émanant de nouvelles souches, peuvent également se retrouver des lectures de régions intergéniques ou des lectures avec des erreurs de séquençage s'alignant par hasard à des chemins non colorés du graphe de variations.

Cette nouvelle méthodologie a été validée sur des jeux de données simulés comprenant un nombre croissant de souches nouvelles à détecter. Bien qu'il subsiste encore quelques familles de gènes pour lesquelles le nombre de nouvelles souches est surestimé, elles constituent un nombre négligeable parmi le nombre de familles analysées. Les résultats sont donc prometteurs mais l'algorithme nécessitera encore des ajustements afin d'améliorer les prédictions.

Conclusion

Les limites des approches actuelles au niveau espèce pour décrire la composition d'un échantillon métagénomique reposent principalement sur la réduction de la redondance lors de la construction de catalogues de gènes, rendant impossible des analyses au niveau souche. D'autre part, même si des outils ont déjà été développés pour s'intéresser aux souches dans un échantillon métagénomique, aucun ne semble proposer une exploration complète comprenant l'identification de souches connues (via des références), la détection de nouvelles souches, et l'estimation de l'abondance de toutes ces souches. A fortiori, aucun outil ne semble non plus proposer d'utiliser des graphes pour répondre à cette problématique, alors même qu'ils constituent une structure de donnée efficace pour représenter des génomes similaires tels que les génomes des souches d'une même espèce.

C'est pourquoi j'ai développé **StrainFLAIR**, un outil de profilage d'échantillons mé-

tagénomiques au niveau souche. Il permet d'indexer des gènes sous forme de graphes de variations puis de les requêter à partir de lectures de séquençage. Notre approche permet de détecter les souches présentes dans l'échantillon parmi les références, d'identifier de nouvelles souches proches de ces mêmes références et d'estimer l'abondance relative de ces souches.

Les principales perspectives pour **StrainFLAIR** seront d'élargir les validations à des jeux de données plus complexes ainsi que de proposer une application avec un jeu de données réel. D'autre part, certains des nouveaux développements nécessitent d'être encore explorés et améliorés, notamment pour la détection de nouvelles souches et l'estimation de leur abondance. Les perspectives plus à long terme viseront principalement à exploiter la dynamicité des graphes de variations, en les mettant à jour avec les nouvelles variations trouvées dans un échantillon, ainsi que d'ajouter une composante pangénomique. En effet, la représentation sans perte des graphes de variations est particulièrement adaptée pour étudier les acquisitions/pertes de gènes au sein d'une espèce et pour analyser plus finement les variations, permettant alors de mettre en évidence l'organisation des génomes et des régions de plasticité au sein d'une espèce.

TABLE OF CONTENTS

Introduction	21
1 Biology concepts and methodological approaches	23
1.1 Genomics	24
1.1.1 The genome	24
1.1.2 Sequencing DNA	27
1.1.3 Assembly and reference genomes	33
1.1.4 Prokaryote genomics	34
1.1.5 Bacterial strains	37
1.2 Pangenomics	39
1.2.1 History and definition	39
1.2.2 Pangenome analysis	42
1.3 Metagenomics	43
1.3.1 History and definition	43
1.3.2 16S and shotgun metagenomics	43
1.3.3 Metagenomics analysis	45
2 Scientific context and objectives	49
2.1 Overview on species detection and quantification in metagenomics	50
2.1.1 Reference-based methods	50
2.1.2 Reference-free methods	53
2.2 Reference gene catalog	54
2.2.1 The 3.3 million genes catalog	55
2.2.2 Updated gene catalogs	56
2.2.3 Reconstructing species	58
2.3 Thesis objectives	61
3 State of the art	63
3.1 Graph representations and alignments	64
3.1.1 Graph data structures	64

TABLE OF CONTENTS

3.1.2	Sequence-to-graph alignments	70
3.2	Tools for strain-level profiling	71
3.3	Thesis rational	77
4	StrainFLAIR: strain-level profiling of metagenomics sample using variation graphs	79
4.1	Pipeline	79
4.1.1	Overview	79
4.1.2	Indexing strains	81
4.1.3	Querying variation graphs	83
4.2	Validation	88
4.2.1	Validation on a simulated dataset	89
4.2.2	Validation on a real dataset	97
4.2.3	Abundance metrics validation	100
4.2.4	Performances	101
4.3	Conclusion	102
5	Towards paths disambiguation and strains inference	103
5.1	Path attributions disambiguation	103
5.1.1	Rational	103
5.1.2	Algorithm	104
5.1.3	Validation	107
5.1.4	Conclusion	113
5.2	Inference of new strains	114
5.2.1	Rational	114
5.2.2	Algorithm	115
5.2.3	Validation	117
5.2.4	Conclusion	124
	Conclusion	127
	Appendix	135
	Bibliography	139
	Scientific contributions	157

List of figures	160
List of tables	162

INTRODUCTION

Microorganisms, and especially bacteria, are preponderant in many ecosystems, including the human body. The high-throughput sequencing technologies played a major role in their characterization, notably in metagenomics, by giving access to their whole genetic information and their functional potential. Studies of bacterial communities in the field of human health for example often aim at identifying biomarker signatures by associating phenotypic characteristics to gene abundances or bacterial species abundances in a cohort of individuals. However, a more refined resolution is now needed, as those signatures might be invisible at the species level and be in fact strain-dependant. Studying bacterial communities at the strain level is of great interest and arise various bioinformatic challenges.

This thesis work proposes to explore this growing field and associated thematic like multiple genome sequence representations as graphs, and offers a new bioinformatic solution to the current challenges.

This thesis manuscript is decomposed into five chapters. The first chapter lays the foundation of all biological concepts and related methodological approaches required to fully understand every implication of this work. The second chapter details the scientific context with the current and most popular approaches to study metagenomic samples, at the species level, drawing the challenges in the field and highlighting the main objectives of the thesis. The third chapter describes the state of the art precisely on the concern of graph representation for a set of genome and on the strain-level analysis of metagenomic samples, placing this work relatively to the field and emphasizing my contribution. The fourth chapter presents the main work realized for this thesis and published, the proposed solution to the challenges raised and considering the already existing approaches. Called **StrainFLAIR**, its pipeline is detailed and its results on various datasets are commented. Finally, the fifth chapter reports the developments realized on **StrainFLAIR** after its publication and release. The manuscript ends with a conclusion summarizing the challenges, contributions, limitations, and perspectives of this project.

BIOLOGY CONCEPTS AND METHODOLOGICAL APPROACHES

The first chapter of this thesis focuses on defining all the pre-required concepts and presents their history in order to fully understand the following context, challenges, and state of the art for this work.

This chapter is divided into three parts, starting with all **genomics**-related definitions and notions. Then, as illustrated in Figure 1.1, at a single species level on one hand, genomics is extended to **pangenomics**, and, at a community level on the other hand, genomics is extended to **metagenomics**.

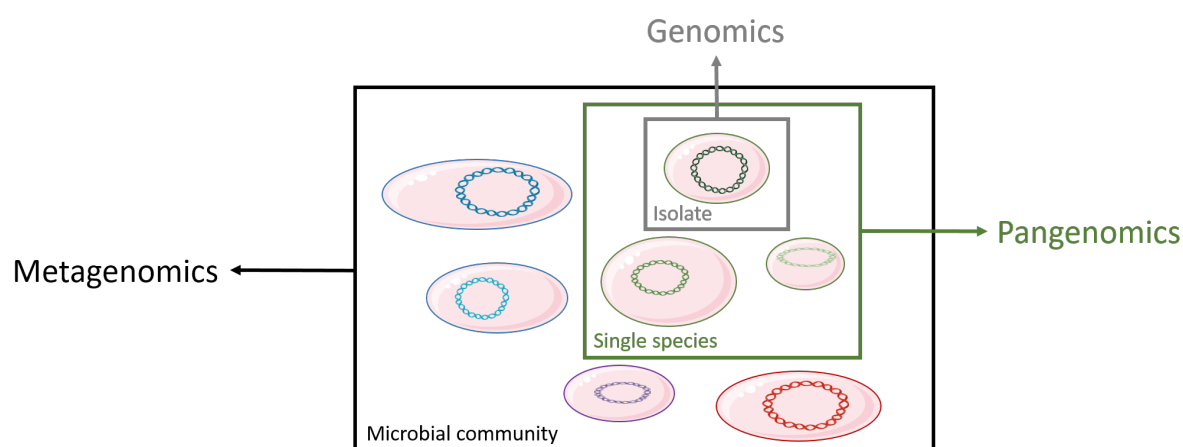


Figure 1.1 – **Illustration of the link between Genomics, Pangenomics, and Metagenomics.** Made partially from smart.servier.com images.

1.1 Genomics

Genomics is an interdisciplinary field of biology that studies the structure and function of the genome, as well as related sub-fields such as evolution. This section gets back to the definition of genome and the main methodological and bioinformatic approaches that have been developed to study it, and describes one of the main characteristics for this thesis that are the variations in the genome, leading to the focus on strains.

1.1.1 The genome

Definition and structure

The **genome** is the whole set of genetic information carried by every living organism. The study of the genome is what is called *genomics*. Depending on the species, the genome is organized into one or several **chromosomes** (Figure 1.2a). Chromosomes are themselves organized into **genes** (Figure 1.2b), the basic unit coding for proteins, key actors of all chemical reactions essential to life. A chromosome is a condensed molecule of deoxyribonucleic acid (**DNA**) which holds the genetic material for protein synthesis (Figure 1.2c).

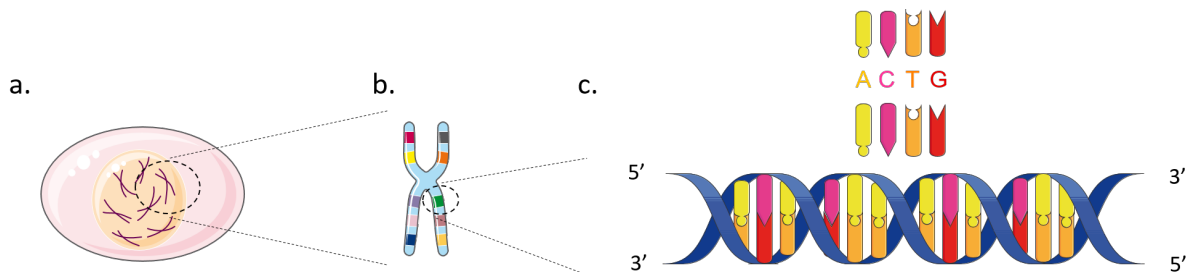


Figure 1.2 – **From a chromosome to a DNA molecule.** **a.** Example of a cell in which the nucleus contains several chromosomes. **b.** A compacted chromosome in which several genes are localized, illustrated by different colors. **c.** A double-stranded DNA molecule composed of the four nucleotides, illustrated by different colors. The forward and reverse strands are paired from their 5' to their 3' ends. Made partially from smart.servier.com images.

DNA has two strands organized as a double helix structure. Each strand is composed of four elementary bricks called **nucleotides** (or bases) and symbolized by single letters: A for Adenine, T for Thymine, C for Cytosine, and G for Guanine. Because the nucleotides from one strand form bonds with particular other nucleotides from the other strand, the two strands are described as complementary. Indeed, A and T are paired together, as are

C and G. When mentioning length measures in the DNA, it is then usual to address them in base pairs (bp).

Moreover, the DNA strands are oriented according to their two distinct ends. The 5' end is a phosphate group and the 3' end is a ribose group. During DNA synthesis, nucleotides are incorporated from the 5' end to the 3' end. Often, one of the strand is called the *forward strand* while the other is called the *complementary strand*.

The genome size is highly variable across species and can represent a large volume of data. Among the most studied model organisms, can be mentioned: *Escherichia coli* with a genome of 4.6 Megabases (Mb), *Caenorhabditis elegans* with a genome of 100 Mb, or *Homo sapiens* with a genome of 3.1 Gigabases (Gb).

Studying genomics aims to identify the information contained in genomes and thus better understand the biological functions involved. This is of great interest in many research fields such as in phylogenetic studies to understand the evolutionary relationships among species, in ecology studies to understand the interactions between organisms, or in human health to understand the causes and mechanisms of diseases for example. Part of this identification of information also involves looking into variations in the genome.

Variations

The genome is unique to each individual and can undergo several mutations (changes in the DNA) caused intrinsically by errors during biological events (e.g. DNA replication) or by external factors (mutagen agents). Identifying and characterizing those **variations** are also a major concern when studying genomics, particularly in *comparative genomics* where the genomic characteristics of different organisms are compared.

When variations among the same species occur on the same gene, each possible version is called an **allele**. The co-existence of different alleles in a population is what is called *genetic polymorphism*. Different alleles can also co-exist within the same individual. This is the case for diploid or polyploid organisms, that is to say organisms that have paired sets of two chromosomes or more, respectively. For instance, the human genome is diploid, i.e. each chromosome is present in two copies (except for the sexual chromosomes) and called *homologous chromosomes*. The composition of the set of alleles constitutes the *genotype*. If the alleles are identical for a given locus (a fixed position on a chromosome), the genotype

is said *homozygous*, as opposed to *heterozygous*. The specific combination of alleles of the whole set of genes located on a single chromosome corresponds to an **haplotype** (see Figure 1.3). Several types of variations exist and affect the genome in different ways.

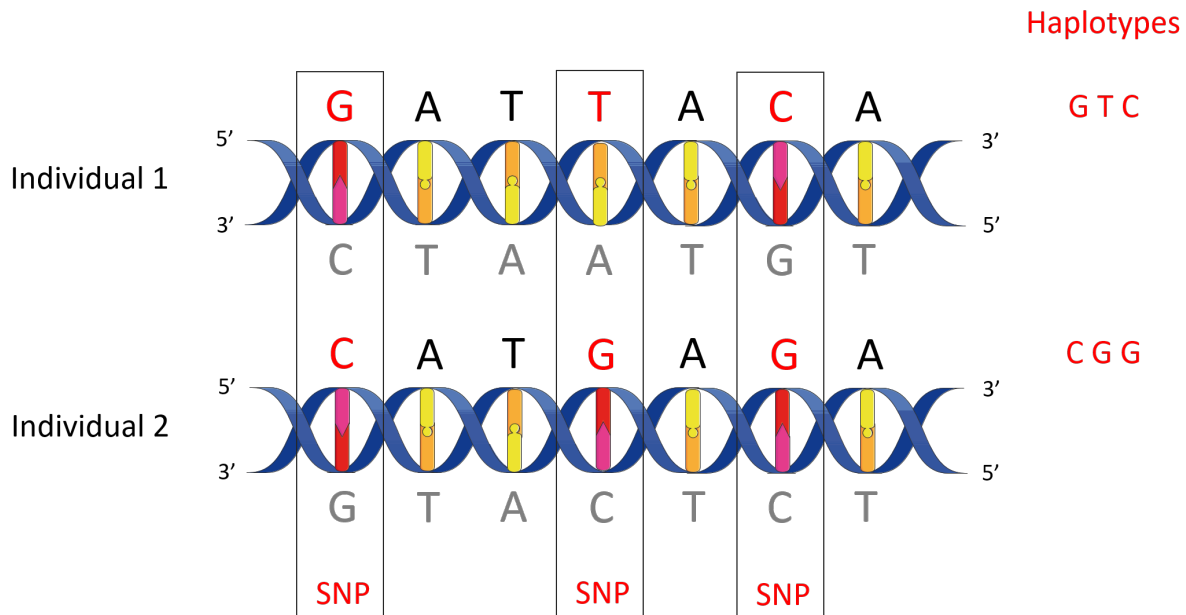


Figure 1.3 – **Illustration of multiple SNPs defining haplotypes between two individuals.** Considering two haploid individuals, the locus represented shows three SNPs at the first, fourth and sixth position of the forward strand. The set of those particular SNPs, GTC for the first individual and CGG for the second, defines the two possible haplotypes.

Made partially from smart.servier.com images.

As previously mentioned, the DNA molecule codes for proteins. Briefly, the DNA is read during a *transcription* process, resulting in a messenger ribonucleic acid (mRNA) that is itself read during a *translation* process by sets of three successive (ribo)nucleotides called *codons*. The way of dividing the DNA sequence into triplets of consecutive and non-overlapping nucleotides defines the *reading frame*. The codons are the ones coding for amino acids, the basic units of the proteins. Because various different codons can code for the same amino acid, the genetic code is said to be redundant. As a result, the types of variations affect differently the genome.

Variations limited to a single base pair are called **single nucleotide polymorphism** (SNP). However, this terminology usually implies specifically substitutions, that is to say the replacement of one base by another one (see Figure 1.3). Because of the redundancy of the genetic code, substitutions are more likely to induce silent mutations that does not

affect the resulting protein neither the resulting function. On the other hand, insertions or deletions of nucleotides, conjointly referred as **indels**, are more likely to induce a shift in the reading frame of the DNA, producing truncated proteins. This can result in malfunctions or even no functions at all. Finally, indels of more than 50 bp long are called *structural variants* (SVs). Due to their length, SVs are likely to have more important consequences on biological functions (Mills et al., 2011). Similarly to smaller variations, SVs can be insertions or deletions. They can also involve more complex rearrangements like inversions (the sequence segment orientation is inverted) or translocations (the sequence segment is found in another region in the genome), and even combinations of multiple different rearrangements.

Accessing DNA sequences and exploring their variations have been made possible thanks to the development of sequencing technologies and bioinformatic tools. The next sections detail those advances before discussing more focused aspects of genomics in bacteria.

Highlights

Across species, variations in the genome are observed and are highly valuable information. They are used for classification and as markers of biological functions or dysfunctions for example.

1.1.2 Sequencing DNA

DNA sequencing consists in determining the base pair sequence of a DNA fragment. To access this information, several sequencing technologies have been developed over the years.

DNA fragmentation

Despite the progress in sequencing technologies, they still cannot read the genome sequence in a single piece. Hence, the DNA needs to be fragmented beforehand. The size of the fragments is one of the main characteristics distinguishing the sequencing technologies. Those fragments are then sequenced, resulting in multiple copies of the molecule read called **sequencing reads**. Most often, DNA fragmentation uses a **whole-genome**

shotgun approach that consists in fragmenting randomly and several times the genome (see Figure 1.4).

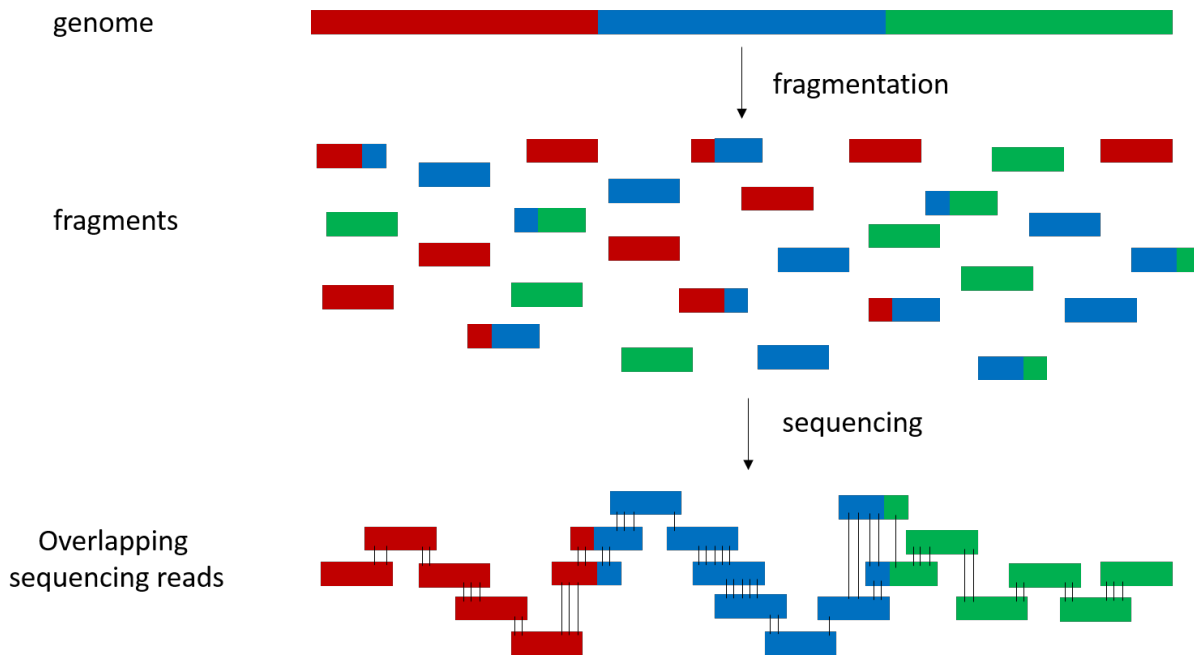


Figure 1.4 – **Whole-genome shotgun sequencing.** The genome is fragmented into random fragments several times. This allows for overlapping fragments that reconstruct the initial sequence.

Following the sequencing, the set of reads corresponds to the whole molecule of DNA sequenced. Since multiple copies have been generated, each position of the genome is *covered* by a certain number n of reads. Averaged, it defines the *sequencing depth*. Additionally, the percentage of sequence covered by at least one read defines the *breadth of coverage* of the genome.

First generation sequencing

Developed in 1977, Sanger sequencing has been the gold standard for DNA sequencing for almost 40 years. After a culture of identical cells as a source of DNA, the principle consists in using a DNA polymerase to generate a complementary copy to a single stranded DNA template. Four separate sequencing reactions are realized, each one adding only one of the four dideoxynucleotides (ddNTPs). Dideoxynucleotides resemble the DNA monomers enough to allow incorporation into the growing strand. However, they lack a

3' hydroxyl group which is required for DNA extension, leading to the synthesis reaction termination. Moreover, each dideoxynucleotide bears a specific fluorescent label which will serve for automatic detection of the DNA base. As a result of both of these characteristics, many DNA fragments of different length are generated and are terminated at all base positions of the template by one of the fluorescent dideoxynucleotide. The DNA fragments are then separated by size using electrophoresis techniques.

Thus, Sanger sequencing provided genomic information as fragments called **sequencing reads** of few hundred bases long. In parallel, the ribosomal ribonucleic acid (rRNA) was described as a marker for classification of bacterial species (often the 16S rRNA gene). Here, the rRNA is directly sequenced from the environment, without going through a bacterial cultivation step, a considerable advantage since cultivation is not always possible. This approach revealed the bias of the cultivation-based methods, as they led to an understanding of microbial biodiversity shifted towards the easiest cultivable species (J. T. Staley and Konopka, 2003). The vast majority of microbial biodiversity had then been missed (Hugenholtz et al., 1998; Rappé and Giovannoni, 2003).

Most recent Sanger sequencing instruments use capillary-based electrophoresis. It allows up to 96 sequencing reactions to be analyzed simultaneously. This limits the number of generated sequencing reads and therefore increases the cost for a large sequencing project. However, in the past decades, next-generation sequencing systems have been introduced and, this time, allow millions of sequencing reactions to be analyzed at the same time. Nevertheless, Sanger sequencing is still extensively used today for small-scale projects or to validate next-generation sequencing data.

Next-generation sequencing

Next-generation sequencing (NGS) has enabled the sequencing of thousands of DNA molecules simultaneously, given them also the denomination of high-throughput technologies. Those technologies have been revolutionizing the fields of genetic diseases, clinical diagnostics, and personalized medicine from their ability to sequence multiple individuals at the same time.

Although different machines have been developed, they share common features. NGS platforms require a library preparation step. This library is obtained by amplification or ligation with custom adapter sequences. Then, each library fragment is amplified on a solid surface with covalently attached DNA linkers that hybridize the library adapters. The amplification creates clusters of DNA, each originating from a single library fragment,

and each cluster then acts as an individual sequencing reaction. The output is a collection of DNA fragments (sequencing reads) generated at each cluster. Those fragments length is between 75 bp and 400 bp, hence their “short-reads” denomination. The differences between the various NGS machines lie mainly in the technology for the sequencing reaction and can be categorized into four groups: pyrosequencing, sequencing by synthesis, sequencing by ligation, and ion semiconductor sequencing. Additionally to sequencing reaction differences, those high-throughput sequencing technologies outputs differ in three essential aspects: the length of the reads obtained, their quantity, and their error rate (see Table 1.1; Goodwin et al., 2016).

The first pyrosequencing approach was the 454 pyrosequencing (Roche; Margulies et al., 2005). Like Sanger sequencing, a single nucleotide is incorporated at a time. The incorporation of a dNTP into a strand results in the release of the pyrophosphate which induces an enzymatic cascade leading to a bioluminescence signal. A camera detects each burst of light, that can be attributed to the incorporation of one or more identical dNTPs. The 454 sequencing offers a greater read length compared to other short-read sequencing technologies, with reads up to around 700 bp, providing advantages for repetitive or complex DNA. However, a major drawback is the dominance of indel errors rate, despite the overall error rate being equal compared to the other technologies. Eventually, the 454 platform has been unable to compete in terms of yield or cost.

The Ion Torrent (Rothberg et al., 2011) is similar to the 454 sequencing, as it relies on adding a single nucleotide at a time. For this reason, they also share the same advantages and limits. The difference between the two technologies lies in the way of detecting the dNTP incorporation. The Ion Torrent uses ion semiconductor sequencing. As each dNTP is incorporated, the platform detects the H⁺ ions that are released through the pH change induced. This sequencing technology produces sequencing reads of around 400 bp.

The best known sequencing by ligation technology is the SOLiD platform (Thermo Fisher; Valouev et al., 2008). It uses two-base-encoded probes with fluorometric signals ligated to anchors. Four fluorescent signals are used, each covering a subset of four combinations of dinucleotides over all sixteen. Several cycles of probe–anchor binding and ligation, with different offsets to ensure every base is sequenced, are realized to elongate the complementary strand. SOLiD sequencing can produce reads up to 75 bp.

Finally, among the sequencing by synthesis technologies, the Illumina platform is the best known (Bentley et al., 2008). Similarly to Sanger sequencing, it uses fluorescent ter-

minator molecules. The OH group on the 3' end is blocked, preventing further elongation of the complementary strand. However, once the colors corresponding to the bases have been detected, the fluorophores are cleaved and washed, and the OH group regenerated, leading to the continuation of the hybridization of untransformed nucleotides and the elongation of the complementary strand. Because Illumina does not rely on single nucleotide incorporation, it is less susceptible to homopolymer (same consecutive bases) errors, as opposed to 454 or Ion Torrent sequencing. Illumina produces reads of around 150 bp.

Considering the cost, the fewer limitations, and the high level of cross-platform compatibility, Illumina dominates the short-read sequencing industry. The works presented in this thesis use or are inspired by Illumina short-read sequencing, as it is still one of the most used platform in the field. Nevertheless, new strategies and new sequencing technologies have emerged to produce longer reads.

From longer reads to third generation sequencing technologies

Complex and long repetitive elements, copy number alterations, and SVs are particularly relevant to adaptation and diseases (McCarroll and Altshuler, 2007; Mirkin, 2007; Stankiewicz and Lupski, 2010). However, those elements are so long that short-read technologies cannot resolve them (Mahmoud et al., 2019). Hence the need for longer reads.

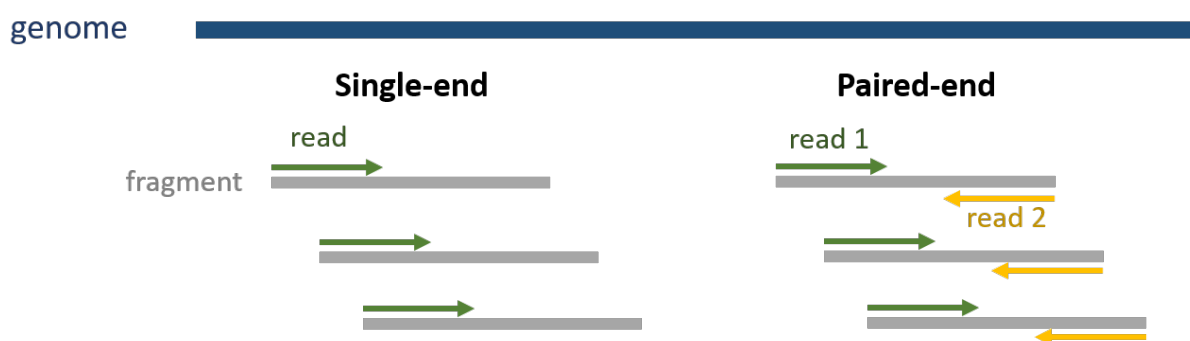


Figure 1.5 – **Single-end and Paired-end sequencing.** In single-end sequencing, the sequencing starts on only one end (represented by the green read). On the contrary, in paired-end sequencing, the fragment is sequenced from both ends on opposing strands (represented by the green and yellow reads).

One of the first attempt to overcome such limitations is the use of **paired-end reads** (see Figure 1.5). For instance, Illumina offers such technology. As opposed to single-end reads, paired-end reads correspond to the sequencing of two ends from opposing strands. The distance between the two ends can be controlled, from overlapping to being a given size apart. Other technologies, based on NGS but looking for short reads coming from the same long DNA molecule, have also been developed. For example, Moleculo technology developed what they called *synthetic long reads* (Voskoboynik et al., 2013), and 10X Genomics technology produced *linked-reads* (Zheng et al., 2016).

Going further, in the last decade, third generation sequencing technologies emerged. Instead of using amplification processes, the DNA molecule is directly sequenced and in real-time, producing reads of several kb long. There are two prominent long-read sequencing: single-molecule real-time sequencing by Pacific Biosciences (PacBio) released in 2011 and Nanopore sequencing by Oxford Nanopore Technologies released in 2014. Both sequence a single molecule, abolishing amplification bias (Heather and Chain, 2016). However, they suffer from a high error rate averaging about 13% (Dohm et al., 2020), at least in first pass. While several methodologies and error correction tools have allowed to lower to 5% of error rate (Goodwin et al., 2016) or even 1% with PacBio HiFi reads (Wenger et al., 2019), this is still higher than the current NGS technologies. Moreover, since third generation sequencing is still emerging, the technologies are continuously improved and require new methodological developments that need to evolve with those advances. Finally, the cost per Gb is still higher than NGS sequencing as shown Table 1.1, slowing its global use.

While the work presented do not use long reads, they are of particular interest for the perspectives they will bring in the field.

Highlights

Whole-genome shotgun and Illumina short-read sequencing are the most popular strategies for sequencing DNA molecules.

Sequencing reads length are usually around 150 bp, but paired-end reads allow for a longer breadth of coverage.

	Technology	Read length (bp)	Error rate	Cost per Gb (US\$)
First gen.	Sanger	400-900	<0.1%	NA
Second gen.	ILLUMINA	150 (single-end) 300 (paired-end)	<0.1%	\$7
	Roche 454	700	1%	\$9,500
	SOLiD	75	<0.1%	\$70
Third gen.	Pacific Bioscience	10,000	5%	\$1,000
	Oxford Nanopore	10,000	5%	\$750

Table 1.1 – **Sequencing technologies characteristics.** Read length, error rate and cost for each main technologies in 2016. Third generation sequencing is now capable to generate reads with even lower error rates (around 1%).

Adapted from Goodwin et al., 2016.

1.1.3 Assembly and reference genomes

As seen, sequencing DNA produces overlapping fragments. In order to resolve the complete sequence of the initial genome sequenced, those fragments have to be ordered and correctly overlapped. This process is called *assembly*. Since each position is covered by n reads depending of the sequencing depth and since genomes are constituted of repetitive sequences that can exceed the size of the reads, assembly is confronted to this complexity and often leaves gaps between longer overlapped fragments. Those assembled fragments are called *contigs*. Methodologies, like the use of paired-end reads, also allow to order those contigs, leading to *scaffolds*.

Most of the assemblers (tools for assembly) use graph representations to reconstruct the contigs. Those graph structures are of particular interest in this thesis and are further presented in Chapter 3.

The development of high-throughput technologies and assembly approaches have led to an increasing availability of **reference genomes**. A reference genome is the result of the assembled sequencing reads from a number of individual donors and therefore used as a representative of one idealized organism of a species. Reference genomes have been and still are core bases of many studies and bioinformatic developments, in particular as the input for aligning (or mapping) sequencing reads, as presented in the following sections and chapters.

1.1.4 Prokaryote genomics

In the three-domain system represented in the tree of life, Archaea and Bacteria are called *prokaryotes*, while Eukarya are called *eukaryotes*. The main difference lies in the existence of a enveloped nucleus containing the DNA in the case of *eukaryotes*. This thesis focuses on prokaryotes, especially bacteria, and this section presents their specific characteristics.

Genome structure specificities

One of the main characteristics to note in prokaryotes is the existence of only one chromosome. Although they do not have a nucleus delimited by a membrane, the chromosome is still located in a specific region of the cell called *nucleoid*. However, the nucleoid does not contain all the genetic material. Additionally to the chromosome, smaller DNA molecules called *plasmids* can be found in the cell.

While plasmids are not exclusive to prokaryotes, they are most commonly found in bacteria. Plasmids can replicate independently and carry genes that, despite not being essential for the cell functions, benefit their survival. Indeed, those genes usually confer selective advantages like antibiotic resistance or virulence factors that can be disseminated in the bacterial population through a process called *horizontal gene transfer*.

In both chromosome and plasmids, the DNA molecule is usually circular (linear plasmids also exists). As opposed to eukaryotes that have linear chromosomes.

Another main characteristic of prokaryotes is the organization of genes into operons. While this is not exclusive to prokaryotes, operons are still most commonly found in those cells. An operon is a set of co-localised and co-expressed genes. The distance between the genes of an operon is then close to zero or sometimes even negative when overlapping each other (Koonin and Wolf, 2008). And because they are controlled by the same promoter (DNA sequence where proteins bind to start the transcription), they are expressed (or inhibited) together.

Finally, the size of the genome is not only very variable across species of the different domains as mentioned in Section 1.1.1, but also within the prokaryotes. Approximately, it scales from 160 Kb (*Carsonella ruddii*; Katsir et al., 2018) to 15 Mb (*Sorangium cellulosum*; Schneiker et al., 2007). Additionally, the distribution of the prokaryotic genomes

size reveals two peaks. One around 2 Mb and a smaller one around 5 Mb (Koonin and Wolf, 2008).

Highlights

The prokaryotic genome is usually composed of a single chromosome, and none, one or multiple plasmids. Both structures are usually circular.

Classification

Initially, prokaryotes were considered as a single species capable of expressing various shapes. The first attempts for classification were then based on morphological observations.

With the development of the cultivation techniques, new tests to distinguish bacteria were developed, and permitted the phenotypic description of these organisms (Woese, 1992; Logan, 2009; Brenner et al., 2015). Morphology can relate to cellular characteristics such as the shape of the bacteria, the presence or absence of an endospore (a structure found during dormant states), the presence or absence of flagella (a structure used for mobility) or, one of the best known, Gram staining. It is used to classify bacterial species into gram-positive bacteria and gram-negative bacteria, based on the chemical and physical properties of their cell walls. Gram-positive cells have a cell wall composed of a thick layer of peptidoglycan that retains the violet-colored stain. Gram-negative cells have a cell wall composed of a thinner peptidoglycan layer that less retains the stain and appears pink under microscope (Coico, 2006). Some prokaryotes do not have cell walls and cannot be colored at all. Morphology can also refer to characters relative to the colony such as color, dimensions or form. In addition to the morphological observations, phenotypic description includes physiological and biochemical features. Physiological features include data related to the culture medium: liquid or solid nature of the medium in order to grow, growth temperature, pH values, salt concentrations, or atmospheric conditions (e.g. aerobic/anaerobic). Biochemical features include data on sources of energy in order to grow or products of the metabolic activity (Rosselló -Mora and Amann, 2001).

Close to morphological characteristics, bacteria can be classified into *serotypes* that define *subspecies*. A serotype is a group of organisms that have the same surface structures.

Two bacteria may look the same under microscope but present different surface antigens (molecule capable of initiating an immune response). For instance, for the *Salmonella* bacteria, more than 2,600 serotypes have been described based on two antigenic structures (Grimont and Weill, 2007): the O antigen on the lipopolysaccharidic cell surface and the H antigen part of the flagella. Determining the serotype, or serotyping, consists in observing through a microscope the presence or absence of aggregates. In contact with the serum of a patient, if the antibodies present in the sample are specific to the serotype of the cells studied, the bond between them will form aggregates. Otherwise, in the absence of aggregates, antibodies are not specific and the cells are from a different serotype. Those surface structures are particularly involved in the virulence or antibiotic resistance of the bacteria.

Prokaryotes are also classified according to clinical considerations. Therefore, they are classified as *pathogens* when associated with diseases, and non-pathogens, themselves divided into *commensals* when sharing the same nutrients of other organisms, or *saprophytes* when processing decayed organic matter. Classification according to pathogenicity is, in fact, of limited value for two main reasons. Firstly, many species considered as commensals may cause diseases depending on circumstances. This is the case with *Escherichia coli*, *Staphylococcus saprophyticus* or *Streptococcus viridans* for example. The pathogenicity depends on the host (e.g. age or genetic factors) as well as on microbial factors, and the interaction between both of them (Shanson, 1989). Secondly, because of clinical criteria for classification, some prokaryotes are considered as different species while being extremely close according to other criteria. For example, this is the case for *Bacillus cereus* and *Bacillus anthracis*. While being genetically close, *B. anthracis* has two plasmids (pX01 and pX02) absent in *B. cereus*. Those two plasmids are responsible for the pathogenicity of *B. anthracis* that lead to the anthrax disease (Helgason et al., 2000).

While classification has long been based on phenotypic characteristics, the sequencing of the first bacterial genome in 1995 (Fleischmann et al., 1995) has been a game changer and opened the way for sequence comparisons. The main disadvantage of phenotype-based classifications is that the whole information potential of a prokaryotic genome is not used, since gene expression is directly related to the environmental conditions. Sequencing gave access to the whole genetic repertoire of microorganisms and DNA sequences from different microorganisms could then be compared leading to similarity measures. Historically

at specific loci level and now at the complete genome level. The Needleman–Wunsch algorithm is one of the best known and still used algorithm in bioinformatics to align protein or nucleotide sequences (Needleman and Wunsch, 1970). From the comparison of several genomes, phylogenetic trees can then be inferred, giving new insights about the species evolution.

Classifying groups of biological organisms based on shared characteristics is called **taxonomy**. Each group, or taxon, corresponds to a taxonomic rank. The highest rank, mentioned previously, is the domain rank (Archaea, Bacteria and Eukarya) which is then decomposed into: kingdom, phylum, class, order, family, genus, and species, which has been the main focus of the discussions until now. Serotypes allow for an even lower rank, the subspecies. However, only classification using genetic characteristics can fully access the whole potential of the information contained in the genome. In that respect, the following section details a rank under subspecies and using genetic characteristics: **strains**, that will be applied more specifically to bacteria in this thesis.

1.1.5 Bacterial strains

Thanks to the expansion of sequencing techniques and the resulting increased number of bacterial genome sequences available, comparative genomics highlighted the genetic diversity within bacterial species (Fraser-Liggett, 2005; Tettelin et al., 2005; Lefébure and Stanhope, 2007). Those genomic variations defined the strain rank. They typically consist of SNPs and acquisition or loss of genomic elements (genes, operons, or plasmids; Tettelin et al., 2005).

In fact, the definition of a bacterial strain is not trivial. Theoretically, a strain lineage is composed of genetically identical genomes. According to the *Bergey's Manual of Systematic Bacteriology* “A strain is made up of the descendants of a single isolation in pure culture and usually is made up of a succession of cultures ultimately derived from an initial single colony” (J. Staley and Krieg, 1984). However, in practice, very close genomes are also considered as the same strain. While mutations and acquisition/loss of genes are expected within the same strain, the increasing number of these genomic events can lead the strain to evolve into what will be considered to be a different strain. Hence the need to also add specific phenotypic or genotypic traits, including serotyping or functional traits, to the definition. Tenover *et al.* defined a strain as “an isolate or group of isolates that

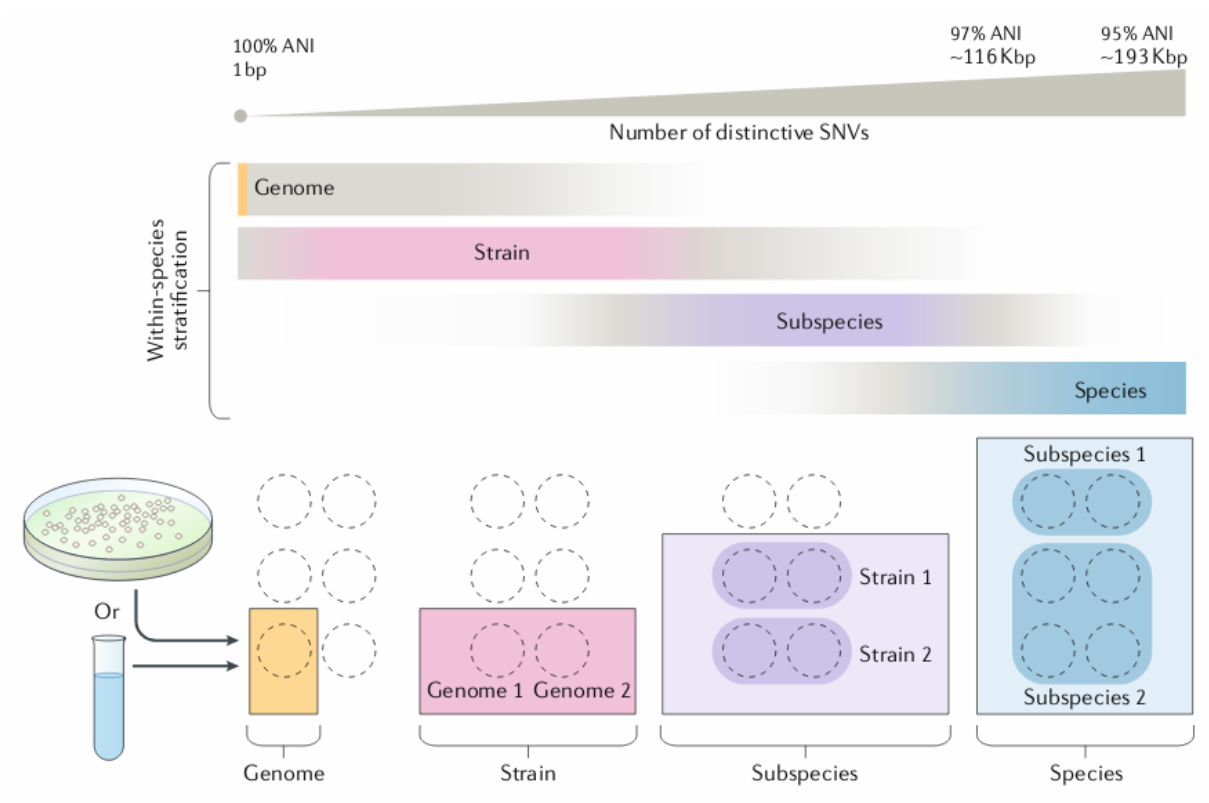


Figure 1.6 – **Within-species stratification.** Bacterial species may be stratified according to the number of single nucleotide variations (SNV) in the whole genome. This is also related to sequence similarity measures like the ANI score presented on the top of the figure. The colored portions of the bars represent the recommended scope of use of the term, while the grey parts their common but unspecific use. Taken from Van Rossum et al., 2020.

can be distinguished from other isolates of the same genus and species by phenotypic characteristics or genotypic characteristics or both” (Tenover et al., 1995).

To conclude, a universal definition of strain based on strong biological characteristics has not been established and may not exist. While a genome correspond to an individual, the distinction between strain, subspecies, and species might be seen as a spectrum rather than discrete categories as illustrated in Figure 1.6.

In parallel to the concept of strains, those genomic variations found at the single species level introduced the concept of *pangenome* that is detailed in the following section.

Highlights

While there is no universal definition, a strain lineage usually refers to genetically identical or very close genomes.

1.2 Pangenomics

1.2.1 History and definition

The pangenome of a species is the whole set of genes found within this species. The pangenome concept emerged from the first comparative studies. In 1998, when studying *Helicobacter pylori*, several DNA fragments from various strains were found while absent from the known reference (Akopyants et al., 1998). Another study compared a pathogenic strain of *Escherishia coli* O157 to a commensal strain K12 and showed a substantial gain in the O157 genome called *pathogenicity islands* (Perna et al., 2001). The term of *microbial pangenome* was actually first used in 2005 (Medini et al., 2005) to describe the union of the shared genes found in the genome set of interest.

As a result, the pangenome was first divided into *core genome* (genes present in all strains of the species) and *accessory genome*, also known as dispensable genome or flexible genome (genes present solely in one or some strains; Tettelin et al., 2005). A schematic representation of the pangenome is provided in Figure 1.7. The core genome typically includes housekeeping genes for the cell envelope or regulatory functions while the accessory genome includes genes for specific adaptations (e.g. antibiotic resistance).

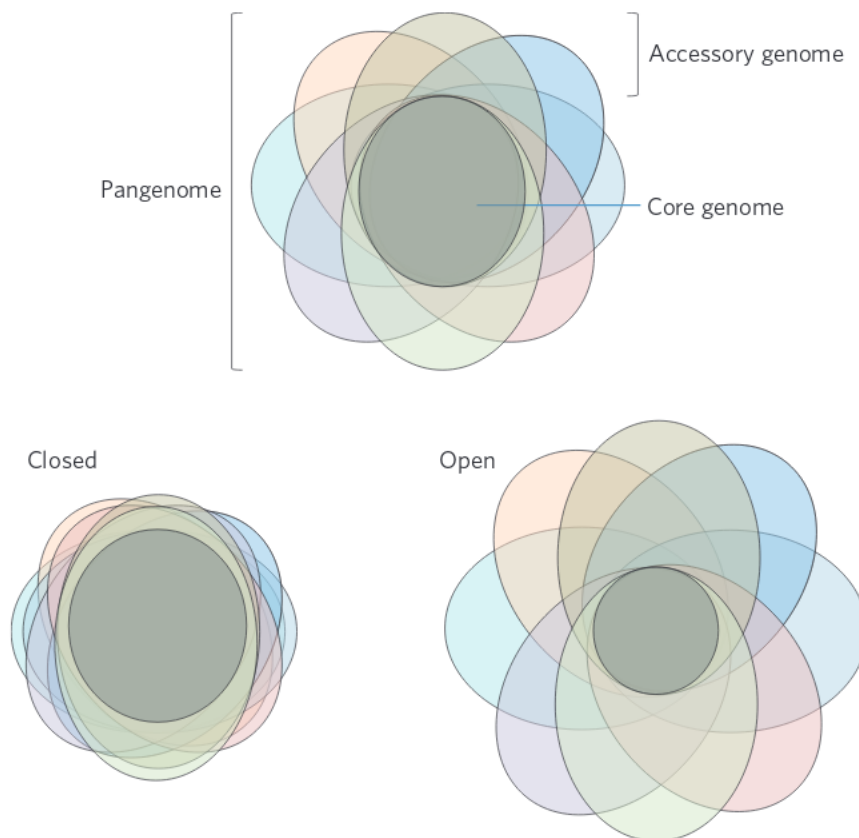


Figure 1.7 – **Schematic representation of pangenomes as Venn diagrams.** Each colored circle represents the set of genes of a single strain. The size of the pangenome differs among the species. Closed pangenomes have a larger proportion of core genes, as opposed to open pangenomes. Taken from McInerney et al., 2017.

However, other works (Makarova et al., 2007; Koonin and Wolf, 2008) have extended the study of pangenome not only at the level of a single species, but at the kingdom taxonomic level. The distribution of the number of organisms sharing genes in orthologous gene families (genes found in different species but inherited from the same last common ancestor) formed a U-shape higher on the left (see Figure 1.8). This distribution could be decomposed into three parts:

- The *core genome*: corresponding to the right part of the distribution, lower than the left part. As with the previous classification into core/accessory genomes, the core genome here still refers to the genes common to all strains/species considered;
- The *shell genome*: corresponding to the middle part of the distribution. It refers to the moderately shared gene families;
- The *cloud genome*: corresponding to the left part of the distribution. It refers to rare genes families.

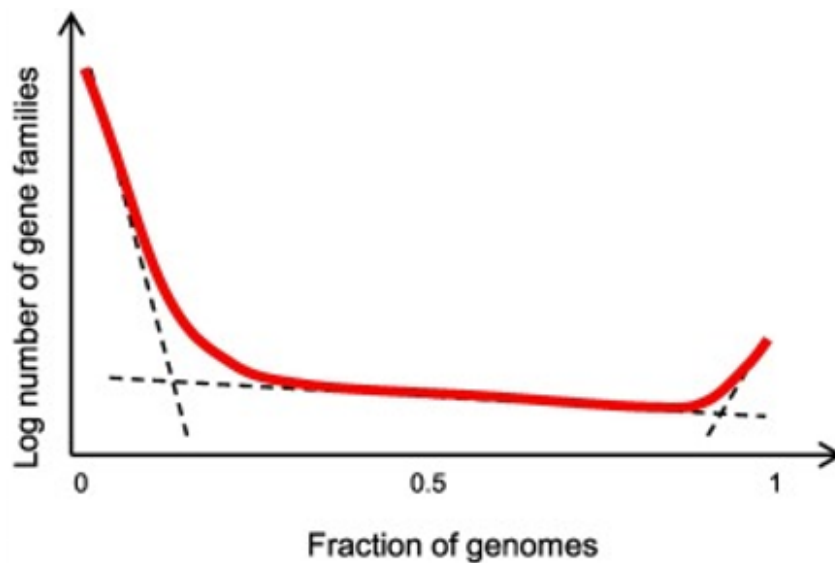


Figure 1.8 – **Gene commonality in prokaryote genomes.** A plot of sharing of orthologs (gene commonality) between prokaryote genomes shows an asymmetric U-shape. It also shows three distinct tangents (dashed lines) corresponding to the three sub-divisions of the genome into core, shell and cloud. The number of organisms has been converted into a scale from 0 to 1 representing the fraction of genomes while the number of gene families has been log-transformed.

Taken from Koonin, Makarova, et al., 2021.

A negative correlation is usually found between the size of the pangenome (total number of genes in the pangenome) and the proportion of core genes. This led to the definition of “*open*” and “*closed*” pangenomes. An open pangenome is a large pangenome with a low proportion of core genes (see Figure 1.7). Species with an open pangenome are associated with higher rates of horizontal gene transfers allowing them to extend their set of genes. A closed pangenome, on the contrary, is a small pangenome with a high proportion of core genes. Hence species with a closed pangenome are less subject to horizontal gene transfer, usually because they are species living in isolated niches (McInerney et al., 2017).

The pangenome is dominated by gain/loss of adaptative genes and, as such, reflects the species response to selective pressure (Brockhurst et al., 2019). Thus, the analysis of a species pangenome has many applications in functional, evolutionary or epidemiological studies (Tettelin et al., 2005; Medini et al., 2005).

1.2.2 Pangenome analysis

Microbial pangenome analyses serve several purposes. As previously seen, pangenome is closely related to the strain taxonomic rank. Hence, one of the first important aim is to characterize bacterial strains by their set of genes. This is of particular interest to uncover strain-specific antibiotic resistance or virulence factors for example. This can also be applied to the identification of new strains in whole metagenomics samples, which will be further discussed in the following sections and chapters. Moreover, pangenome analyses are used to study the impact of horizontal gene transfer on evolution.

Pangenome profiling (determining the pangenome of a species) usually uses genomic homology-based strategies, an approach that is also used in this thesis and that will be further described in Chapter 4. Briefly, genes are clustered into gene families by sequence similarity and, for each strain, a profile of presence/absence of families is often generated.

For example, *Roary* (Page et al., 2015), *PanOCT* (Fouts et al., 2012) or *PGAP* (Zhao et al., 2012) have been developed for orthologs clustering. Another recent example is *PPanGGOLiN* (Gautreau et al., 2020), that builds pangenomes through a graph representation and a statistical method to partition gene families in persistent (a less strict definition of core genome), shell and cloud genomes.

While this thesis work cannot be labeled as being a pangenomic approach, there is a clear and close link with strain-level resolution concerns that will be raised here. The

pangenomic dimension is especially of great interest for the perspectives point of view.

Most of the current tools provide good pangenome profiling for isolates, but cannot resolve species relationships at the community level. The next section presents this other aspect of the extended field of genomics.

1.3 Metagenomics

1.3.1 History and definition

In 1998, Handelsman *et al.* suggested the term *metagenomics* to refer to the direct sequencing of DNA in an environment, potentially revealing the genomes of all individuals present in the sample (Handelsman *et al.*, 1998). More precisely, metagenomics is the study of genetic material recovered directly from environmental samples in an untargeted way. The complete set of microbes found in a particular microbial ecosystem is called *microbiome*. While the collection of genomes from those microbes is called *metagenome*.

Before the emergence of metagenomics, the first studies of microorganisms in environments such as water, soil, or human tissues at a resolution beyond the human eye were made using direct observation through microscopes. It allowed to view single-cell organisms and observe morphological characteristics. With the further development of more powerful microscopes and staining techniques, imagery-based methods were widely used for bacteria classification (Madigan *et al.*, 2014). The microbiology field then relied on microbial isolates since isolation and cultivation became the most common approach for microbial characterization (Ben-David and Davidson, 2014). Even today, cultivation-dependent methods are still used to determine characteristics of microbial species. Associated with this, the field of molecular biology also experienced an expansion with DNA sequencing techniques allowing for a larger-scale screening of the existing microbes.

1.3.2 16S and shotgun metagenomics

As previously mentioned in Section 1.1.2, 16S rRNA gene sequencing (shortened here as “16S sequencing” for readability) has been used for taxonomic profiling. In the NGS era, 16S sequencing is still used and constitutes what is called *metataxonomics* or *metabarcoding*. It is important to note that in fact, metataxonomics and metagenomics are now often distinguished (Esposito and Kirschberg, 2014). Indeed, the strict definition emphasizes the untargeted nature of metagenomics, which conflicts with the sequencing of a

particular gene in 16S sequencing. Yet, both relate to studies of genetic content in an environmental sample. The 16S sequencing uses PCR to target and amplify portions of the gene encoding for the 16S rRNA. This gene contains nine hypervariable regions. Their sequences can be unique and thus can be used to separate species and to identify bacteria present in a complex community. After sequencing, a data processing step is realized by assigning sequences to Operational Taxonomic Units (OTUs). OTU assignment can be reference-based or *de novo*. For reference-based methods, reads are mapped to known 16S genes from a reference database and assigned to the closest match. *De novo* methods are usually applied to the reads that did not map using a reference-based approach. Those previously unmapped reads are clustered by their similarity and the taxonomy can be inferred by various strategies that will not be detailed here (Westcott and Schloss, 2015; Rideout et al., 2014).

While 16S sequencing is a targeted approach, shotgun metagenomics is, however, dedicated for whole-genome sequencing. Similarly to whole-genome shotgun presented in Section 1.1.2, shotgun metagenomics consists in fragmenting DNA from a sample in a random manner. Instead of fragmenting a single genome, all genomes present in the sample are fragmented. Those fragments are then sequenced using NGS technologies. Shotgun metagenomics allows a detailed characterization of entire microbial communities as it targets all DNA material from all the species present in the sample. Similarly to 16S sequencing, reference-based or *de novo* methods can be used to produce a species-level profile of the sample. This will be further discussed in Chapter 2.

One sequencing technique has no absolute advantage over the other one and depends solely on the objectives of the study. The following reviews the pros and cons of both sequencing techniques.

Firstly, bacterial coverage depends on the species coverage of available reference databases. Those references are constructed from assembly of sequencing data. In the case of 16S sequencing, assembly requires few sequencing reads as only the 16S gene needs to be assembled for characterization of the species present in the sample. On the contrary, shotgun metagenomics depends on assembling entire genomes thus requiring more sequencing and more computation. As a result, 16S databases contain references for more species than shotgun one's. Therefore, 16S sequencing identifies more diverse bacterial phyla and families than shotgun sequencing (Shah et al., 2011; Tessler et al., 2017).

Secondly, in terms of taxonomic identification, because 16S sequencing targets a bacterial marker gene, it is limited to bacterial identification only. Shotgun sequencing on the other hand is able to identify all taxonomic domains (bacteria, eukaryotes, and archaea).

Finally, and one of the main key point for this thesis, 16S and shotgun sequencing do not reach the same taxonomic resolution. Using 16S sequencing, usually only genus-level identification is reached. However, combination of errors correction tools, optimized primers and curated reference database allows for up to species-level identifications. For shotgun sequencing, since it covers all genetic variations, it should in theory allow for strain-level resolution. In practice, bioinformatic tools for strain-level profiling are under active development and this aspect will be further discussed in Chapter 3. Overall, shotgun sequencing provides more information but harder to interpret.

Unless stated otherwise, the rest of the manuscript will only consider whole-genome shotgun sequencing, its applications and outputs, as it is the required methodology for the strain-level resolution aimed in this thesis. Hence, mentions of metagenomics or metagenomics sequencing will refer to the strict definition previously given.

Highlights

Metagenomics sequencing allow the study of complete microbial communities directly from environmental samples.

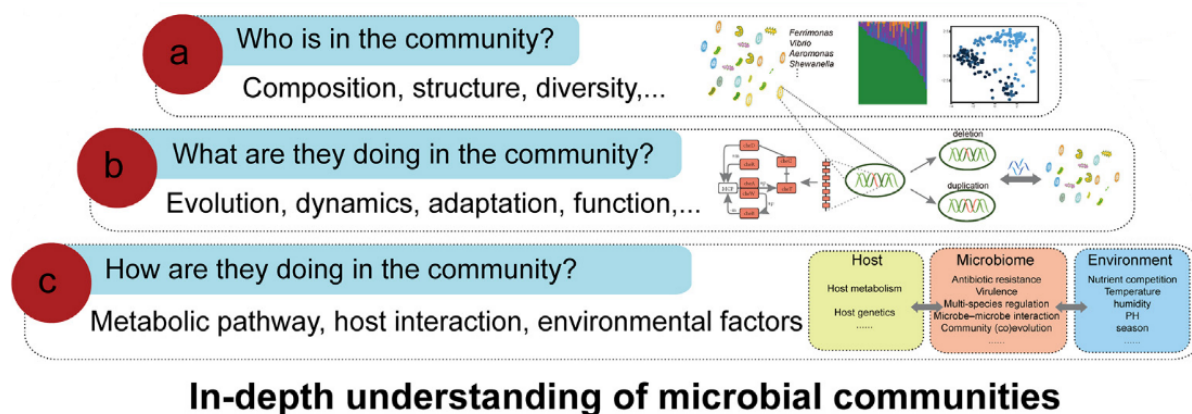
Additionally to the close-to-exhaustive coverage of all the present species, all genetic variations can also be covered, opening the possibilities for strain-level resolution analysis.

More bioinformatic developments are needed in this direction.

1.3.3 Metagenomics analysis

Metagenomics can be used for taxonomic and functional analysis. For example, it has been used to study the microbiome composition of the human body, such as the gut (Solé et al., 2021). The imbalance of the gut microbiome, also known as dysbiosis, has been implicated in several diseases including metabolic disorders, obesity, diabetes or autoimmune diseases (Carding et al., 2015). This imbalance may be the result of a gain/loss of community members or changes in their relative abundance. The role of the microbiome is then of great interest for human health and requires to develop downstream analysis to

both explore and query those microbiomes. When performing microbiome studies, three main questions arise (see Figure 1.9; Zhong et al., 2021).



In-depth understanding of microbial communities

Figure 1.9 – **In-depth understanding of microbial communities.** Metagenomics analysis can be summarized into three concerns to address: Who is in the community? What are they doing in it? and how they do it?

Taken from Zhong et al., 2021.

First is the question of “who” composes the sample, implying a focus on the composition, structure, and diversity of the microbial community. This can be addressed as a general overview using richness (“How many microbes?”) or using diversity metrics (“How are microbes balanced to each other?”). Richness of the gut microbiota, for instance, is strongly correlated to metabolic markers and a high richness is often associated to an healthy gut microbiome (Le Chatelier et al., 2013). At a more detailed level, present microbes in the sample can be described at different taxonomic levels by the relative proportion (or abundance) of the community members.

The second question that can be addressed is what are they doing in the ecosystem of interest. This implies to understand how microbes interact with the environment or the host on one hand, and how they interact between each other in the other hand. Those interactions can be described as the functional chemistry carried out by the microbes, by characterizing the products consumed from and excreted to the environment.

The third question is to understand how they do what they have been found doing in the previous question. This relates to determining the enzymatic pathways activated or

overexpressed, and how they are distributed among the present microorganisms.

For the purpose of this thesis, only the analysis of the composition will be addressed, especially in terms of relative abundance of the community members. Hence, the next chapter focuses on the existing context for detection and quantification of species in metagenomics.

SCIENTIFIC CONTEXT AND OBJECTIVES

Microorganisms are predominant in most of the ecosystems. They are found in the ocean (Sunagawa et al., 2015) or, of particular interest in this thesis, the human body (Clemente et al., 2012). They play major functioning roles in those ecosystems (New and Brito, 2020) that high-throughput sequencing, as seen in the previous chapter, allowed to study as a whole, especially in terms of composition at the species level (Jovel et al., 2016).

Metagenomics analyses and resolving the species present in a sample with their relative abundances have highlighted associations with phenotypes, particularly in human health (Ehrlich, 2011; Vieira-Silva et al., 2020; Solé et al., 2021). Notably, some diseases are characterized by the presence of potentially pathogenic bacteria, whereas others result from the depletion of health-associated species.

However, while existing analyses mainly focus on a species-level resolution, characterizing samples at the strain level has a growing interest. For example, *Escherichia coli* has a highly variable genome and is well known for presenting commensal strains (thus harmless) whereas others are pathogens (Rasko et al., 2008; Loman et al., 2013). Strain-level analyses are therefore crucial to highlight new associations with phenotypes and will provide a better understanding of their functional impact in host-microbe interactions and advances towards personalized medicine.

This chapter presents the current methodologies in metagenomics to detect and quantify bacterial species in an environmental sample, as well as their limits that have led to the strain-oriented purpose of this thesis.

2.1 Overview on species detection and quantification in metagenomics

As previously mentioned, one of the major objective when studying metagenomics samples is to be able to describe the species they contain. This description usually consists in identifying the species present on one hand, and to quantify them on the other hand. The methods used for this purpose can be divided into two main approaches: *reference-based* and *reference-free* approaches.

2.1.1 Reference-based methods

Reference-based methods are also known as taxonomic classification or taxonomic binning. With reference-based methods, the input sequences are clustered into bins that correspond to a taxonomic identifier. Since they rely on reference genomes, these approaches are particularly useful to identify exact genomes or close relatives present in the sample compared to the reference database. Among the reference-based methods, several approaches can also be distinguished: alignment-based methods, marker-based methods and sequence-composition-based methods.

Alignment-based methods

Alignment-based methods use sequence alignment approaches. Sequence alignments consist in identifying regions of similarity between two sequences. At a position, if the bases of the two sequences are identical, it is a *match*, otherwise, it is a *mismatch*. If one of the two sequences has no bases at the considered position, it is a *gap*. Considering a score function, all those matches, mismatches, or gaps define an alignment score that represents the similarity between the two sequences. However, aligning correctly and finding the best alignment score for two sequences are not trivial tasks. In 1970, the Needleman-Wunsch algorithm was proposed for the *global alignment* problem (Needleman and Wunsch, 1970) which aims at finding the best alignment of two sequences over their whole lengths. The Needleman-Wunsch algorithm uses dynamic programming (a mathematical optimization method that simplifies a problem into sub-problems in a recursive manner) to find the best solution (best alignment score). In 1981, the Smith-Waterman algorithm was proposed for the *local alignment* problem (Smith and Waterman, 1981) which aims at finding the best alignment over only regions of the two sequences. The Smith-Waterman algorithm

also relies on dynamic programming to find the best solution. Those algorithms have set the foundations for the current state-of-the-art alignment tools.

BLAST (Altschul et al., 1990) has developed a *seed-and-extend* strategy in which highly similar regions (called *seeds*) between the two sequences are first searched. Then, using dynamic programming, the alignment is extended from these seeds. Such strategy has been since widely adopted by other tools. While BLAST is particularly adapted to find local similarities, it is also important to mention, for the following sections, that when working with sequencing reads, the goal is often to map them on reference genomes. That is to say, to find their unique location in the genome with a high similarity. The current state-of-the-art tools for short read mapping are `Bowtie 2` (Langmead and Salzberg, 2012) and `BWA` (H. Li and Durbin, 2009).

Initially, in order to identify species present in a sample, BLAST was used to align the sequencing reads against all sequences in GenBank (a public DNA sequences database). In the mean time, new methods have been developed to allow for faster computations.

For instance, `MegaBlast` (Zhang et al., 2000) uses a greedy algorithm for the alignment that, instead of using the similarity between two sequences with a dynamic programming grid, uses the difference between them (usually a smaller metric compared to the number of identities) and the alignment is defined as the minimum number of differences. Another well known tool is `Megan` (Huson et al., 2007). Alignments are still realized using BLAST as a pre-processing step, then Megan is used to explore the taxonomical content of the dataset by assigning reads to the lowest common ancestor of the set of taxa found through the BLAST results.

One of the main limitations of these approaches is due to the expansion of sequencing techniques, that allowed for an increase of the available sequences. The reference databases are growing and it is more and more computationally demanding, if not infeasible, to align to all possible sequences. Hence, other methods presented below have been developed to provide faster results.

Marker-based methods

Marker-based methods rely on specific sequences (marker sequences), usually genes, to identify the species. Those markers act as taxonomic references and can then be used to detect the taxa present in a sample. Compared to the previous approach that uses

sequences databases, partially or completely composed of full genomes, markers allow for a smaller database and therefore faster assignments.

For instance, **MetaPhlan** (Segata et al., 2012) uses **Bowtie 2** to align reads on a previously constructed catalog of markers. From a set of almost 3,000 genomes, the authors identified 2 million potential markers, that were filtered to get a subset of around 400,000 genes most representative of each taxonomic unit. More than thousands of species were covered with around 200 markers per species. Another example is **PhyloSift** (Darling et al., 2014). It uses **LAST** (Kielbasa et al., 2011) to align reads in order to search for sequence similarity against a database of known reference gene families, adds the sequences to a multiple alignment with the reference genes, and places them onto a phylogenetic tree of the reference genes.

One of the main limitations of these approaches is that they do not exploit the full potential of the sequencing data. Because reads are aligned solely on marker genes, most of them are not classified. An even more promising approach resides in sequence-composition-based methods (Lindgreen et al., 2016).

Sequence-composition-based methods

Sequence-composition-based methods are based on the nucleotide composition. The reference genome, additionally to its taxonomic label, is represented by k-mers. K-mers are substrings of length k contained within the genome sequence. Successive k-mers overlap over $k - 1$ nucleotides. The sequencing reads are then searched and classified using a k-mers database.

For instance, **Kraken** (Wood and Salzberg, 2014) builds a taxonomic tree from reference genomes in which k-mers are associated to each nodes and leafs of the tree. This association relies on the representativeness of the k-mer towards the considered taxonomic level. Reads are themselves decomposed into k-mers that are searched in the taxonomic tree. Finally, classification is determined by finding the highest-weighted path in the tree. Another popular tool is **Clark** (Ounit et al., 2015). Specific k-mers at the species or genus level are used to build the reference database. However, as opposed to **Kraken**, k-mers associated to higher taxonomic levels are not used. Then, **Clark** uses a similar approach compared to **Kraken** by searching the reads k-mers in the created database.

Other tools have been developed with various optimization objectives. In order to reduce the size of the k-mers database, **Centrifuge** (Kim, Song, et al., 2016) uses an indexing scheme based on a FM-index, an index based itself on the Burrows-Wheeler transform, to compress the k-mers database. **Kraken2** (Wood, Lu, et al., 2019), on the other hand, uses minimizers (a method to sample k-mers from a long string). In order to increase the sensitivity of the classification, **SKraken** (Qian et al., 2017), which is inspired from **Kraken**, filter out less representative k-mers for each species. Other strategies propose to use spaced k-mers (Břinda et al., 2015), a strategy inspired from spaced seeds used in the seed-and-extend approach for sequence comparison. Instead of using contiguous k-mers, k-mers interleaved with spaces are used and such strategy showed classification improvements.

Despite those efforts for optimization, the construction of the databases is still a very demanding step, both for RAM and disk space. Nevertheless, the main limitation of reference-based methods is the necessity itself of existing references. Most bacteria found in environmental samples cannot be cultured and remain unknown (Eisen, 2007), without an available reference. For this reason, reference-free methods have been developed.

2.1.2 Reference-free methods

Reference-free methods are usually based on binning strategies. The data used (reads or contigs) are clustered into groups (or bins) that originate from the same species.

Reads clustering

The clustering can be realized directly from the sample sequencing reads. The tools employing this approach, like **MetaCluster** (Wang et al., 2012) or **BiMeta** (VanVinh et al., 2015), are usually based on k-mer distributions, since those distributions are expected to be more similar for reads belonging to the same genome than from different ones. However, those tools are limited to situations with even proportions of species in the sample (their *relative abundance*), which is a requirement usually not met with real samples. On the contrary, other tools, like **AbundanceBin** (Wu and Ye, 2011), work better when there is no species with the same abundance.

Contigs clustering

Instead of clustering directly the sequencing reads, other methods focus on adding an assembly step and cluster contigs. Similarly to the reads clustering approaches, bins of contigs are expected to come from the same species. For instance, **MetaBAT 2** (Kang et al., 2019) uses a graph structure where the contigs are the nodes and the edges between them represent their similarity. A graph partitioning algorithm is then used to cluster the contigs. **GroopM** (Imelfort et al., 2014) map reads onto an assembly realized from the same reads. A coverage profile is then obtained for each contig. The binning is based on the co-varying coverage profiles across multiple samples. **CONCOCT** (Alneberg et al., 2014) clusters contigs by using a combination of sequence composition and coverage, also across multiple samples.

The main limitation of those methods is related to the assembly step. Because of the sequencing errors and/or repetitive regions, reconstructing accurate and long contigs is more challenging, and, by extension, such strategies cannot reconstruct full genomes associated to identified species. Moreover, similar species and strains of the same species are hardly distinguishable.

In conclusion, detection and quantification of species in a sample can be carried out with or without reference genomes. Reference-based methods are best use for identification of species with close relatives in the reference database, while reference-free approaches are particularly useful in the absence of close relatives (Comin et al., 2021). The next section details an approach for the detection and quantification of species in a metagenomic sample that uses similar methods as described up to this point, and will highlight even more the limits and concerns around the identification of strains.

2.2 Reference gene catalog

One of the classic analyses, and the one that laid the foundation of this thesis project, is to use a gene catalog representative of the studied ecosystem. Considering the different strategies defined in the previous sections, this is not actually a reference-based method since it does not rely, at least not only, on known reference genomes. However, a gene catalog is still a reference database in the same way genomes or k-mers are used as databases for these approaches.

2.2.1 The 3.3 million genes catalog

To illustrate the methodology behind the gene catalogs, the field of human gut microbiota is taken as an example. The first gene catalog contained 3.3 million microbial genes (Qin et al., 2010). Human faecal samples from 124 individuals were collected and sequenced using Illumina sequencing (shotgun sequencing with paired-end short reads). In order to construct the most comprehensive catalog, samples were collected from healthy individuals and patients (with various phenotypes expressed or diseases).

The high-quality reads were assembled using the **SOAPdenovo** assembler (R. Li et al., 2010). The assembler uses a popular data structure, the de Bruijn graph, to represent the overlapping reads. The de Bruijn graph will be detailed in Chapter 3. Briefly, it is an oriented graph that represents the overlaps of length $k - 1$ between all the words of length k . One of the pipeline step consists in removing erroneous connections (e.g. removing short tips, merging bubbles) in this de Bruijn graph. However, it is important to note that those short tips or bubbles in the graph are caused by the sequencing errors and/or the genetic variations that exist between strains. Thus, this is a first drawback for exploring the strain-level composition of a sample. Ultimately, the graph was broken at the repeat boundaries and the resulting parts of the graph were linear sequences, the contigs.

The reads were realigned onto the assembled contigs. Since the sequencing technology used output paired-end reads, this paired information was used to order the contigs into scaffolds. The alignment and paired information were also used to fill the gaps between the contigs when possible.

From the contigs obtained, genes were predicted using **MetaGene** (Noguchi et al., 2006). Open Reading Frames (ORFs) are extracted from the contig sequences and associated to a score based on base composition and length. ORFs are defined, at least in this context by the authors of **MetaGene**, as a sequence starting by a start codon and ending by a stop codon. The set of extracted ORFs also included partial ORFs (located at the extremities of the sequence or the entire sequence itself). During a second step, among all ORFs or partial ORFs extracted, an optimal combination of them was computed according to the previously mentioned scores. This approach is particularly useful to predict overlapping genes.

Finally, redundant ORFs (or genes) were removed by pairwise comparison using BLAST with stringent thresholds of identity score of 95% and overlap of 90%. Through this process, only one sequence among the multiple similar ones is selected as a representative gene. It is important to note that those thresholds get rid of the potential sequencing errors, but also interesting variations that again could have been crucial information for strains identification.

2.2.2 Updated gene catalogs

While the previous catalog allowed for a consequent set of gut microbiome genes, it was still based on a limited cohort which restricted the coverage of the global diversity. To overcome this, an integrated gene catalog (IGC) was established (J. Li et al., 2014). Several additional cohorts, alongside the samples from the previous catalog, were sequenced and used to build this catalog. Notably cohorts from different countries that highlighted country-specific gut microbial signatures.

The pipeline to generate this integrated catalog was similar to the pipeline used for the previous catalog (see Figure 2.1). It was first applied to each cohort independently, resulting in intermediate country-specific gene catalogs. Then, the three catalogs were merged together and also with genes found in sequenced genomes and draft genomes (set of scaffolds) from human gut-related prokaryotes available in well known databases (NCBI and EMBL Bank here). This updated gene catalog contains almost 9.9 million genes from the gut microbiota.

The IGC has itself been updated into the IGC2 (Wen et al., 2017). Genes from ankylosing spondylitis patients and healthy controls have been assembled. The genes not present in the IGC were added, leading to a new catalog of almost 10.4 million genes.

Recently, and untied to a unique ecosystem, a new non-redundant gene catalog of 303 million species-level genes (again clustered at 95% of nucleotide identity) has been constructed from 13,174 publicly available metagenomes (Coelho et al., 2021). This highlights how catalogs continue to grow with the increasing number of samples sequenced.

Such catalogs can be used to identify and quantify (in terms of relative abundance) genes in a metagenomic sample. However, another relevant use is to reconstruct species to properly describe the community composition of the sample.

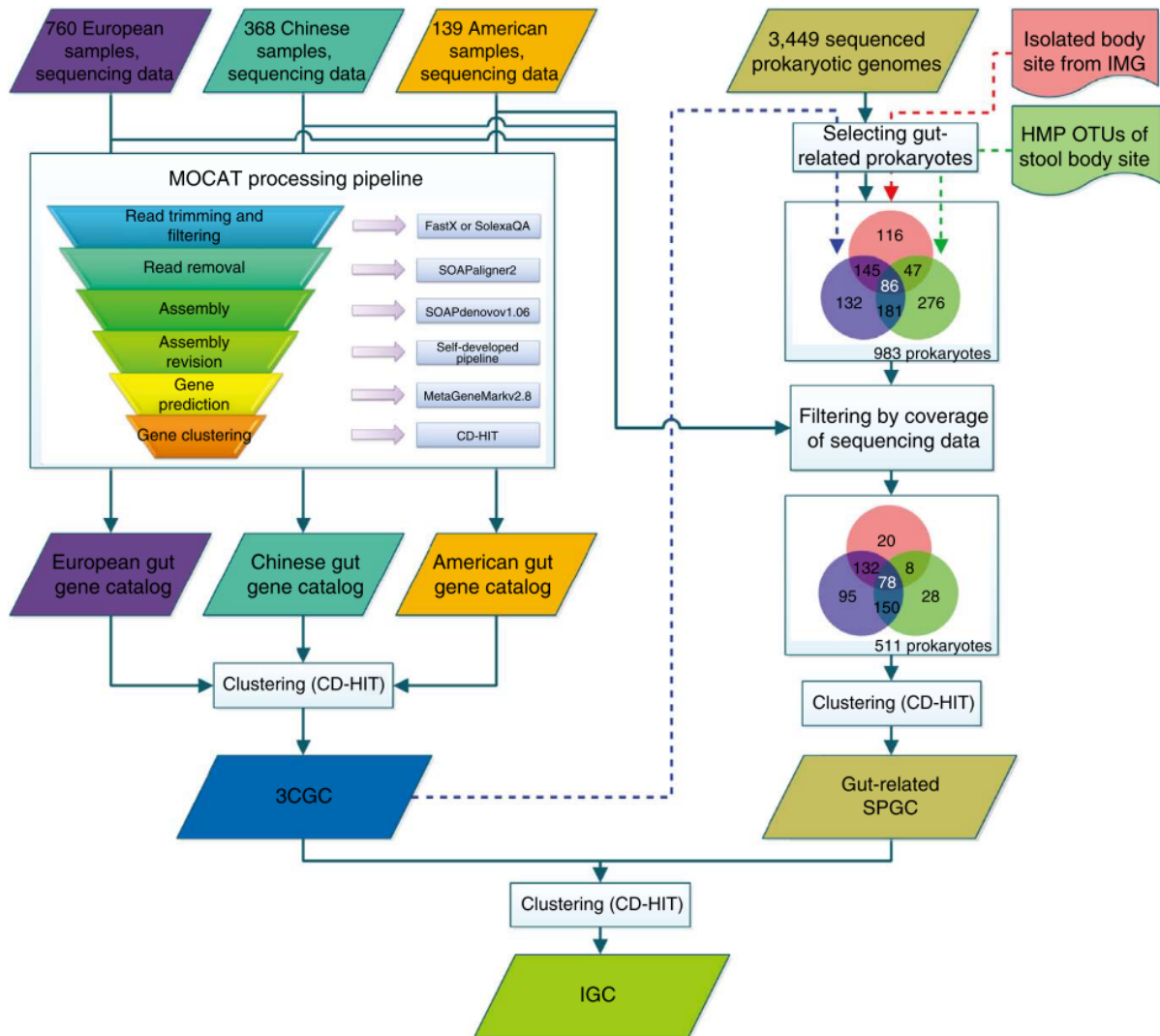


Figure 2.1 – **Construction of the integrated gene catalog.** The pipeline includes data processing and integration. The approach is similar to the construction of the 3.3 million genes catalog but applied to different cohorts independently. Those intermediate catalogs are then integrated into a single gene catalog.

Taken from J. Li et al., 2014.

2.2.3 Reconstructing species

Despite the popularity of Metagenome-Assembled Genomes (MAGs; Eren et al., 2015) reconstruction, reference gene catalogs are still an essential and well-established resource. A MAG is a bin of contigs assumed to represent a single species, therefore they can also be used as reference databases for species identification. However, MAGs present several drawbacks addressed by the current gene catalogs. For instance, short assembled sequences are excluded, while they potentially correspond to genes or partial genes, or biases have been observed, like low genome coverage, preventing the reconstruction of low-abundance organisms (Borderes et al., 2021).

The IGC became the most used publicly available catalog to identify and quantify species in the gut microbiota. While there is no consensus for a catalog of species and thus a consensus on the binning method to use to construct it, this section details both approaches that served as the foundations to initiate the thesis objectives and that proved to be among the best performing binning strategy (by comparison with a gold standard; Borderes et al., 2021).

From a set of samples, the sequencing reads are mapped onto the genes of the gene catalog, resulting in an abundance profile across the samples for each gene.

The first approach uses the canopy clustering algorithm (Nielsen et al., 2014), an unsupervised pre-clustering algorithm, on the gene catalog and the genes abundance profile (see Figure 2.2). Iteratively, a seed gene is chosen among the not yet clustered genes, and genes that share similar abundance profiles (according to distance thresholds) are clustered with it. Once all genes have been clustered, clusters showing close median abundance profiles are merged. Clusters with no sufficient support evidence (e.g. with two or less genes, for which the abundance is in majority driven by a small amount of samples, or for which the abundance profile is not detected in enough samples) are discarded. Clusters with more than 700 genes are called *Metagenomic species* (MGS). Afterwards, for a single metagenomic sample study, the abundance of an MGS can be computed based on the mean abundance of the genes contained in the MGS. This methodology has been successfully used to highlight that the gut microbiota composition is influenced by antibiotics and tyrosine kinase inhibitors, and impacts the success of cancer immunotherapy (Derosa et al., 2020), or to describe the relationship between the gut microbiome and cirrhosis, as

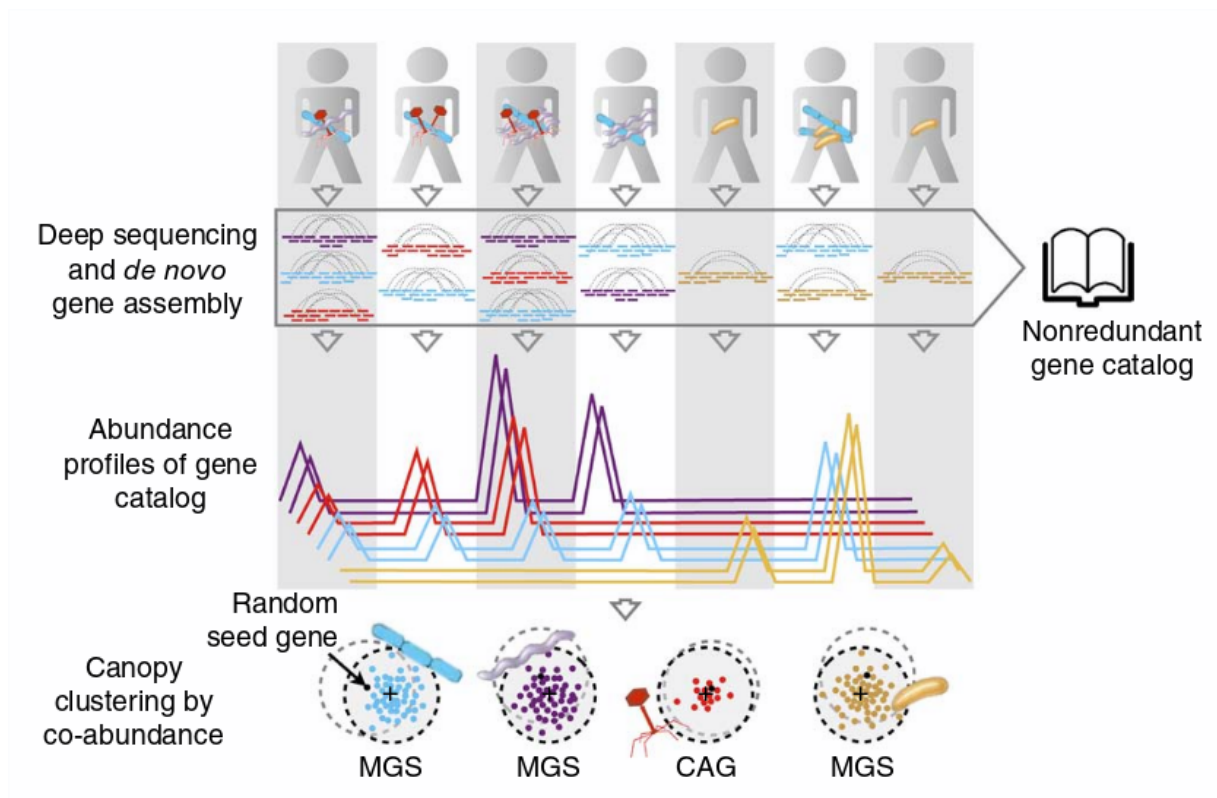


Figure 2.2 – **Overview of co-abundance clustering.** Sequencing reads are mapped onto the gene catalog, providing an abundance profile for multiple samples that can be used to infer metagenomic species.

Taken from Nielsen et al., 2014.

well as its prognosis (Solé et al., 2021), for example.

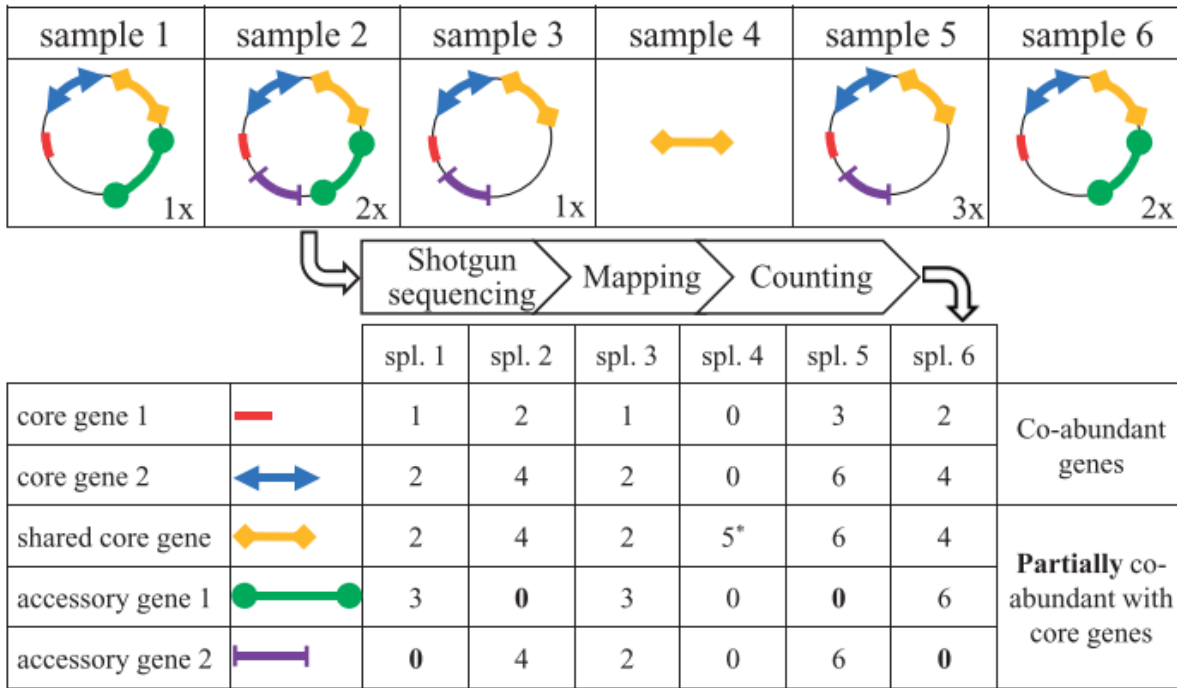


Figure 2.3 – **Simplified model behind MSPminer**. Six samples are represented with five carrying a different strain of the same species. On the bottom right of the strains represented as circles are their absolute abundances. The colored parts correspond to genes. Core genes in red, blue, and yellow. Accessory genes in green and purple. Core genes read count are proportional across samples, while this proportionality for accessory genes is only observed across the subset of samples sharing those accessory genes.

Taken from Oñate Plaza et al., 2019.

The second approach, implemented as the tool **MSPminer** (Oñate Plaza et al., 2019), is also based on clustering methods from the genes profile abundance across samples (see Figure 2.3). However, it adds a pangenomic component. The core genes of a microbial species is expected to be consistently detected in samples where it is present, and accessory genes are expected to have proportional counts only in the samples where a certain strain is present. Briefly, **MSPminer** identifies sets of co-occurring genes, as well as proportional in terms of read counts. According to their presence across the samples, seeds corresponding to core genes are identified. Finally, genes associated to a core seed are grouped, with the seed, into a *Metagenomic Species Pangenome* (MSP). As for an MGS, an MSP is thus a set of genes. However, those genes are labeled as core or accessory. This methodology

has been successfully used to show how interactions between diet and gut microbiome are associated with irritable bowel syndrome (Tap et al., 2021), or how a Mediterranean diet positively impacts the gut microbiota (Meslier et al., 2020), for example.

Despite those efforts in order to get closer to a strain-level resolution by distinguishing core and accessory genomes, as previously seen, the limitations emerged from the gene catalog construction itself. All this scientific background and challenges raised in the field of ecosystems analysis have led to this thesis project.

Highlights

Both reference-based and reference-free methods have proved to be effective for species identification or inference of putative species. However, problems arise when the sequenced sample is composed of closely related species or, even more challenging, composed of a mix of different strains of a species.

Current approaches using gene catalogs get rid of the redundancy when selecting a representative sequence for each genes clusters. Concomitantly, variations, crucial to identify strains of a species, are also lost in the process.

2.3 Thesis objectives

Considering the growing interest and necessity of a strain-level resolution in metagenomic analyses, this thesis work aims at developing a new framework allowing to profile single metagenomic samples, that is to say identify and quantify the strains in the ecosystem community, as well as inferring new strains.

As seen by switching from the 3.3 million genes catalog to the 9.9 million genes catalog (IGC) and the following ones, a major characteristic to note is that each catalog required a set of metagenomic samples. In fact, it is usually a subset of samples, since new samples are often available and will only be used for a following potential update of the catalog. Thus, one of the main limitation of gene catalogs is their frozen nature, highlighting the need for another structure allowing for a more dynamic updating process. In this direction,

we foresaw the graph structures and their ability to be dynamically updated with new information to be a major key to address this concern. The state of the art in Chapter 3 details and reviews those graph representations.

Additionally, the clustering methods used and their thresholds on identity and overlap between sequences imply to select a unique representative sequence among the multiple similar genomes from the same cluster. This results in losing sequences with variations compared to the selected representative and, thus, losing essential information to identify strains. Again, we foresaw the graph structures to be adequate to represent multiple similar sequences. Indeed, as presented in the beginning of the chapter, aligning sequencing reads towards large databases of references is computationally consuming, hence the development of alternatives. Applied to the gene catalogs, it was therefore not conceivable to ignore the redundancy removing step by simply keeping all possible sequences. The graph structure offers a way to compact and represent multiple similar sequences.

Finally, despite the limitations raised from the current approaches presented and how we suggest to address them through graph representations, other tools have been developed specifically for strain-level identification and quantification in metagenomic samples. The state of the art in Chapter 3 also details and reviews those existing tools and their own limitations.

Highlights

The main objectives of this thesis are as follow:

- Use a more global data structure, that can be queried, and dynamically applied to metagenomics;
- Identify and represent all genes from bacterial strains;
- Compute the abundance of strains present in a sample and predict the presence of novel strains.

STATE OF THE ART

The development of high-throughput sequencing technologies has shaped the way genomics studies are conducted. As seen in the first chapter, many methodologies are based on a reference genome that serves as a guide for reconstructing new genomes or for variation identification. A reference genome is never an accurate representation of any single organism genome. Yet, it still represents an approximation of the full genome of any individual and can be used for guided genomic assembly, variant calling or mapping sequencing reads. Moreover, reference genomes provide a unique coordinate system that facilitates sharing of information on variations localization. However, reference genomes represented as flat sequences have demonstrated their limits (Ballouz et al., 2019), especially for the keystone of this thesis, that is to capture the whole genomic variability between multiple similar genomes in order to reach a strain-level resolution in metagenomic analyses. Firstly, read mapping is biased towards the reference. Reads from non-reference alleles may be mis-mapped or not mapped at all. Secondly, even by using multiple reference genomes for the same species, due to the high similarity between them, the mapping would also result in mis-mapped reads or ambiguous alignments generating noise in the downstream analysis (Na et al., 2016).

In this chapter, the state of the art to address the challenges and limitations emphasized here is detailed. The first part is dedicated to the graph representations that have become a popular structure to replace linear reference sequences. The second part is dedicated to the existing tools for strain-level profiling of metagenomic samples, that is to say identifying and/or estimating abundances of strains in a sample. Finally, the third part puts the thesis work relatively to this state of the art.

3.1 Graph representations and alignments

3.1.1 Graph data structures

The graph model

A graph is a structure made up of *vertices* (or *nodes*) connected by *edges* (or *links*). A set of successive connected nodes is usually called a *path*. Using graphs to represent DNA sequences is not a new concept. Most of the assembly softwares use graph representations and related algorithms to assemble reads into contigs. The most popular structures are *string graphs* (Myers, 2005) and *de Bruijn graphs* (see Table 3.1).

Term	Description
Overlap graph	Nodes are sequences, and edges are overlap between the sequences
String graph	An overlap graph for which the redundancy has been removed
de Bruijn graph	Nodes are sequence k -mers, and directed edges connect k -mers whose $k - 1$ suffix overlaps with other k -mers $k - 1$ prefix
Sequence graph	Edges or nodes are labelled with sequences. Used to compress sequence representation and express contiguity between segments with directed or bidirected edges
Bidirected graph	Each edge has a discrete endpoint on either the left or right of a node
Genome graph	A sequence graph relating a genome's sequence information to itself or other genomes
Pangenome graph	A genome graph explicitly involving more than one genome
Variation graph	A pangenome bidirected graph which embed linear sequences as paths

Table 3.1 – **Overview of graphs terminology.** Adapted from Outten and Warren, 2021.

A string graph is an overlap graph without the redundancy, the nodes are sequences of variable size and the edges describe the overlap between the nodes. In the case of an assembly, it corresponds to the overlap between reads, and the length of the overlap needs to be tuned.

In a de Bruijn graph, the nodes are sequences of the same size k and the edges describe the overlap between the nodes, also of the same size $k - 1$. In the case of an assembly, the reads are decomposed into k -mers that will serve as the nodes.

However, graphs can also be used to represent genomes of multiple individuals in order to capture all the variations found in a species while collapsing identical sequences between genomes.

Genome graphs

Considering the limitations mentioned in the introduction of the chapter, a better framework for the analysis of multiple genomes seems required rather than using individual reference genomes. This is why a shift between linear representations to graph representations has been observed during the recent years. Most of those representations are called *genome graphs*, *variation graphs* or *pangenome graphs* (see Table 3.1).

A common way to construct a genome graph is to use a compacted de Bruijn graph (cDBG) from a set of genomes (Beller and Ohlebusch, 2016; Chikhi et al., 2016; Minkin, Pham, et al., 2017). A cDBG is a de Bruijn graph where all unitigs (paths with all but the first node having in-degree 1 and all but the last node having out-degree 1) are compacted into a single node. This structure is particularly relevant for representing and indexing repetitive sequences. Repetitive sequences of length k or more are represented only once. This allows to considerably speed up alignments compared to classic aligners like BWA, especially for genomes composed of many repetitive regions (Liu et al., 2016).

However, cDBG does not keep sample information. For this reason, colored cDBG has been proposed (Iqbal et al., 2012), where each color corresponds to a sample or a population. Eventually, DBG are directed graphs with labeled nodes such that a DNA sequence is defined by the node labels when walking along the graph. This configuration does not allow to distinguish between the forward and reverse complement orientation of the DNA. Hence, directed graphs can be generalized to bidirected graphs so each node can be traversed in both orientations. As opposed to directed graphs, complex DNA rearrangements, like inversions, can be represented in a bidirected configuration without creating separate nodes for the forward and reverse complement orientations (Paten, A. M. Novak, et al., 2017).

For this thesis, proposed graph models as new frameworks for analyses are the main interest. Nevertheless, it is worth mentioning that tools have been developed for specific purposes by relying on graph representations.

Cortex (Iqbal et al., 2012) introduced the cDBG previously mentioned and aims at both realizing *de novo* assembly and discovering/genotyping genetic variants in an individual or population. Taking into account the graph structure locally and genome-wide, the likelihood of each possible genotype is calculated.

BayesTyper (Sibbesen et al., 2018) is a genotyping software. In order to genotype SNPs, indels, and SVs, **BayesTyper** uses exact alignment of read k-mers on a graph constructed from a reference and variants. In practice, close variants are clustered together and a graph is constructed for each cluster. The nodes represent the reference or allelic sequences, and edges represent possible haplotypic links between the sequences. The graph construction uses the reference sequence and all variants in the cluster without collapsing, leading to redundancies of some sequences. The genotyping step is based on a probabilistic model that uses the k-mer profiles generated by traversing the graph.

GraphTyper2 (Eggertsson et al., 2019) is a genotyping software for small variations and structural variants. It is also able to discover small variations, but relies on external resources for SV discovery. **GraphTyper2** constructs a graph from a reference genome and a set of sequence variants in variant-call format (VCF). The graph is a directed acyclic graph (DAG), and a path in the graph represents a possible haplotype. Sequencing reads are aligned to the reference genome and then locally realigned to the graph. Genotyping is done by selecting the two most likely haplotypes in the graph based on the read data. The genotype called is the one that has the highest relative likelihood for each sample.

Paragraph (Chen et al., 2019) is a genotyping software for SVs. It uses a sequence graph in which the nodes represent a sequence and edges represent haplotypic connections. Each path represents an allele (either the reference allele or the alternative allele). **Paragraph** constructs the graph from a reference genome and a VCF file that specifies the SVs breakpoints and alternative allele sequences. Sequencing reads are aligned to the reference genome and realigned to the graph near the breakpoints. Therefore, **Paragraph** uses a very similar approach compared to **GraphTyper2**, even for the genotyping step that also relies on a likelihood maximization based on read counts. The main difference is that **Paragraph** builds the graph and realigns reads on limited specific regions of the genome and not the whole genome.

Pandora (Colquhoun et al., 2021) is a tool for detecting the presence/absence of genes, and genotyping SNPs and indels in bacterial genomes. **Pandora** relies on a pangenome reference graph, as a collection of local graphs. Indeed, **Pandora** focuses on representing only particular loci by constructing the graph from the multiple sequence alignment of

the known alleles for this locus. Sequencing reads are quasi-mapped to the graph, that is to say *Pandora* only compares the minimizers of the read and the local graph, instead of the whole sequence. Thus, each locus has a coverage profile, where regions of low coverage are detected. A *de novo* assembly is used locally in those regions to generate new alleles added to the graph. Reads are quasi-mapped again, generating a presence/absence matrix of the loci.

Gramtools (Letcher et al., 2021) implements a genome graph-based model to discover and genotype SNPs and SVs in a sample. From a set of references (multiple sequence alignment of several reference genomes or one reference genome with a VCF file) **Gramtools** builds a genome graph. Sequencing reads from a sample are mapped onto the graph. Genotyping is realized by calling alleles at each variant site which allows to infer the closest path representative of the sample. This path is then used as a new and personalized haploid reference genome. Discovery of new variants is realized by using standard reference-based variant callers on this personalized reference. Those new variants can afterwards be used to augment (update) the graph.

In conclusion, while those tools have demonstrated the relevance and advantages to use graph structures in genomics, they have been developed for specific purposes, mostly for genotyping, which is not suitable for the thesis objectives. Therefore, on the other hand, other tools have developed a complete framework.

HISAT2 (Kim, Paggi, et al., 2019) is a method that implements a graph-based FM-index (GFM). **HISAT2** starts by creating a linear graph from a single reference genome. Mutations, deletions, and insertions are incorporated in the graph as alternative paths. However, only insertions of up to 20 bp can be incorporated. Any path of the graph defines a string of bases corresponding to the reference genome or its variants. The GFM is used to index the reference genome, one large GFM for the whole genome, and many smaller GFMs to index overlapping portions of the genome. Another significant characteristic is that repetitive sequences in the reference genome are combined into one. The **HISAT2** approach works particularly well for model organisms with available variants. However, **HISAT2** is not intended to represent the exhaustive variations found in a species, and focuses on the most common SNPs, which would be a drawback in the context of this thesis for strain-level profiling.

Minigraph (H. Li, Feng, et al., 2020) is a graph model aiming at representing multiple genomes while preserving the coordinate of the linear reference genome. As previously mentioned, linear representations have the advantage of offering a unique coordinate system. Graph models use the Graphical Fragment Assembly (GFA) format in which each base can be indexed by a segment ID and an offset on the segment. Therefore, the bases have coordinates at the segment level. However, those coordinates are unstable, if the segment is split during the iterative graph construction, the coordinates change. This is why one of the main contribution of **minigraph** is a new reference GFA format. For its graph model, **minigraph** uses a bidirected sequence graph. First from a reference genome, **minigraph** iteratively constructs the graph by mapping each assembly to the reference or the existing graph and augments it with long poorly mapped sequences in the assembly. Hence, **minigraph** mainly captures only long variations between samples. For those reasons, **minigraph** is more adapted to population-scale analyses and/or to represent SVs, and even more specifically for human genome. This is also not suitable for the purpose of this thesis.

Eventually, the **vg toolkit** (Garrison, Sirén, et al., 2018) and associated tools seemed like the most promising softwares and the most adapted for the objectives of the thesis. The next section details the characteristics of **vg**.

Variation graphs

The **variation graph** model developed by Garrison, Sirén, et al., 2018 aims at combining the three main elements of a pangenomic data structure. Thus, a variation graph is a graph $G = (N, E, P)$ composed of a set of nodes $N = n_1 \dots n_M$, a set of edges $E = e_1 \dots e_L$, and a set of paths $P = p_1 \dots p_Q$. The genomes (paths) embedded into the graph are walks through the graph, linking (edges) the DNA sequences (nodes built from an alphabet $A = \{A, C, G, T\}$) between them (see Figure 3.1).

Nodes have numeric identifiers and paths have text string names. The variation graph is a bidirected graph, better suited as mentioned above to represent both strands of DNA, therefore there are four kinds of edges depending on how each of the linked nodes are traversed (forward or reverse complement).

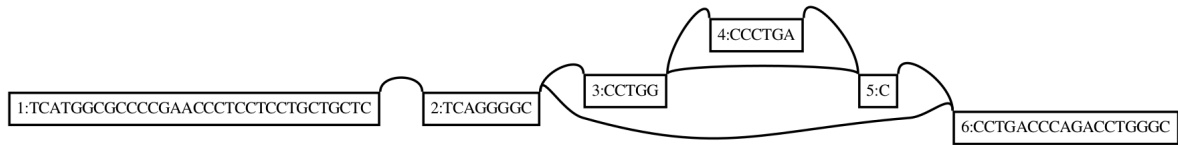


Figure 3.1 – **Example of a variation graph.** A variation graph is composed of nodes and edges. Paths can be embedded into the graph. Taken from Garrison, 2019.

The variation graph model is intentionally simple, as there are neither assertions about the graph structure nor coordinates (as opposed to `minigraph` for which coordinates are its main focus for example). The `vg toolkit` builds a variation graph from VCF files and linear references. Actually, a genome graph can be viewed as an alignment of linear sequences. As such, while `vg toolkit` does not implement coordinates, linear sequences are embedded into the graph and results obtain at the graph level can be projected on the linear sequences. Graphs are implemented into a `.vg` format, an equivalent of GFA. The `vg toolkit` has also its own alignment implementation that is further discussed in Section 3.1.2.

Although graphs built from VCF files and linear sequences were the primary focus of the `vg toolkit`, other tools from the same authors have been developed to expand its use.

Particularly relevant for this thesis, `seqwish` (Garrison, 2022) uses pairwise sequence alignments from `minimap2`, available in GFA format, to losslessly generate the variation graph implied by the initial collection of sequences. Here, losslessly means that the paths embedded in the graph completely and accurately reconstruct the input sequences. The topology of the graph, on the other hand, represents the variations found in the alignment. This method has demonstrated to be faster than other similar approaches like `Cactus` (Armstrong et al., 2020), and to be more flexible than approaches like `SibeliaZ` (Minkin and Medvedev, 2020) that uses DBGs (Garrison, 2019).

In conclusion, the `vg toolkit` provides a graph structure allowing for the reduction of data redundancy without loss of significant information, namely all variations found among the input reference sequences, which is particularly relevant for this thesis.

3.1.2 Sequence-to-graph alignments

As seen in Section 2.1.1, comparing and aligning sequences are at the core of many genomic and bioinformatic analyses. And while consensual state-of-the-art tools exist for linear sequences, the classic algorithms behind them, like the Smith–Waterman algorithm, cannot be applied directly to genome graphs. Hence, mapping sequencing reads to a graph requires specific softwares.

Among the previously cited tools, or related to them, graph alignment heuristics have been developed.

The **de Bruijn Graph-based Aligner** (**deBGA**; Liu et al., 2016) is based on a graph-based seed-and-extension algorithm to align reads onto a DBG. The DBG may have been constructed from one or more genomes. The k-mers from the sequencing reads are used as “seeds” to match to the nodes of the DBG. One of the specificity of **deBGA** is that it merges the seeds that correspond to similar putative read positions if they are localized on the same non-branched path. If the sub-sequences hit by each group of merged seeds also follow a non-branched path, the read is aligned to this path.

HISAT2 implemented a graph FM-index, considered less memory-intensive than k-mer-based indexes (used by the **vg toolkit** for example). The whole-genome FM-index is used to anchor each alignment while the smaller indexes are used for its rapid extension. Otherwise, **HISAT2** is based on the state-of-the-art aligner **Bowtie2**.

Minigraph is based on the same algorithm used by **minimap2**. Ignoring its topology, **minigraph** finds local hits to segments in the graph. If they are connected by edges in the graph, those hits are chained together, giving an approximate mapping location. Due to the strategy chosen to construct the graph stated previously (capturing only long variations), the alignment is not realized at the base level, which is again a drawback for the use of **minigraph** in this thesis work.

The **vg toolkit** has also developed its own sequence-to-graph mapper. Read alignments are represented into a Graph Alignment/Map (GAM) format which is a generalization of the Sequence Alignment/Map (SAM) and Binary Alignment/Map (BAM) classic formats used by the popular sequence-to-sequence mappers like **Bowtie2**.

Similarly to classic read mappers, and thus similarly to the already described **minigraph** method, the approach finds matching seeds and clusters them if they are close. Exact match queries for the seeds are performed in linear time independently of the graph size

by using a GCSA2 index, a k-mer index of a variation graph represented as a DBG (Sirén, 2016). The alignment is realized around the region of each seed cluster.

As opposed to `minigraph`, the `vg toolkit` aligns read at the base level, and also uses the base qualities in alignment score to compute adjusted mapping quality scores. The mapping qualities are computed by comparing the scores of optimal and suboptimal alignments under a probabilistic alignment model, similarly to another state-of-the-art mapper `BWA` (H. Li and Durbin, 2009).

Actually, the `vg toolkit` offers two sequence-to-graph mappers. The first one, `vg map`, outputs one or several paths for each alignment. However, in case of several alignments with equal mapping scores and in the same region, only one is randomly chosen. This thesis work uses the other mapper, `vg mpmap`, to map reads on the variation graph and get more complete results as it is further presented in Chapter 4.

All the presented mappers are mainly dedicated to NGS reads, that is to say short sequencing reads. While this thesis focuses on short reads, it is worth mentioning that other sequence-to-graph mappers have already been developed for long reads. The most popular ones have been developed around the `vg` framework or can be used with a variation graph built from the `vg toolkit`. This is the case of `GraphAligner` (Rautiainen and Marschall, 2020) and `PaSGAL` (Jain, A. Dilthey, et al., 2019).

Highlights

The `vg toolkit` provides a complete environment from a graph data structure (the variation graphs) to sequence-to-graph mappers that suits especially well the challenges around the use of multiple similar genomes.

3.2 Tools for strain-level profiling

In the recent years, several tools have already been developed with various inputs and methodologies to address the issue of a strain-level resolution in a metagenomic sample or several metagenomic samples. The following presents the state of the art on strain-oriented tools, their principle, and the key points in their strategies that are particularly relevant for the thesis rational further detailed in the next section. A summary table is

presented in Table 3.2.

Tool	Inputs		Strain-level outputs		
	References	Samples	Identification	Abundance	
DESMAN	Single core gene	copies species	Multiple	Haplotype inference	Yes
StrainPhlan	Markers reference genomes	from	Multiple	Dominant strain	No
StrainEST	Set of reference genomes	One		Yes (from references)	Yes
DiTASiC	Set of reference genomes	One or Two		Yes (from references)	Yes
mixtureS	One reference genome	One		Number of different strains	Yes

Table 3.2 – **Summary of the input and output characteristics of the existing strain-level profiling tools.**

DESMAN (Quince et al., 2017) is a tool for *de novo* extraction of strains from metagenomes. The rationale behind DESMAN is that no method was developed to resolve strain-level variations in MAGs from assembled contigs and thus without the need for reference genomes. The main challenge being that those MAGs are aggregates of multiple strains.

Before using DESMAN, it is assumed that multiple samples have been sequenced, the resulting reads have been co-assembled (assembly using the pooled set of reads from all samples), the generated contigs binned into MAGs, and the reads from each sample individually have been mapped back onto the contigs.

Firstly, DESMAN identifies core genes present in a MAG in a single copy, based on genes known to be core for all prokaryotes. Those are called single-copy core genes (SCGs). Core genes can also be identified from reference genomes from the same species or related taxa if available. In this case, they are called single-copy core species genes (SCSGs). This identification step on reference genomes is optional, hence the denomination as a *de novo* method, since DESMAN can operate solely on the SCGs, based only on the samples reads.

From the results of read mapping, at each position on SCGs or SCSGs, the base frequency is computed. A likelihood ratio test is applied to those frequencies, summed across samples, in order to determine the positions of the variations. The null hypothesis for the test is that the observed bases have been generated from a single true base under a

multinomial distribution, with an error matrix position-independent. The alternative hypothesis is that two true bases are present. To each variation position, a p-value (corrected for multiple testing) is assigned. Through a probabilistic model applied to the detected variations (selected according to their p-value) across multiple samples, haplotypes are inferred as well as their relative abundance.

Since this first step focuses on core genes, the second component of **DESMAN** consists in resolving the accessory genome. In a highly similar way, a probabilistic model is applied to the variation frequencies. Although, instead of working only on SCGs or SCSGs, the approach is applied to all genes with now also the information of the number of strains present.

In summary, the key characteristics are that **DESMAN** is a *de novo* method, that operates on multiple samples, and that seeks to resolve haplotypes and their abundance.

StrainPhlan (Truong et al., 2017) is a tool for strain-level population genomics using markers as references. When bacterial communities are not sufficiently supported by existing reference genomes or by using existing marker-based approaches, it is difficult, if not impossible, to profile strains from metagenomes. **StrainPhlan** offers a strain-level resolution profiling for each sample from a set of metagenomic samples.

Before using **StrainPhlan**, **MetaPhlan2** is used to identify species-specific markers from reference genomes. The sequencing reads from each metagenomic sample are mapped onto those markers. From the mapping data and by using a majority rule on each nucleotide of the marker, a consensus sequence is built for each marker. For each species and each sample, the consensus sequences are aligned against reference genomes and concatenated into larger alignments. The presence of multiple strains from a species in a single sample is revealed by the evidence of polymorphic sites on the alignments, suggesting multiple alleles. However, while multiple strains can be detected, due to the use of dominant consensus sequences, only the dominant strain can be further described in the next steps of the workflow. **RAxML** (Ott et al., 2007), a maximum-likelihood phylogenetic inference software, processes the previously mentioned concatenated alignments and infer the corresponding phylogenetic tree. **StrainPhlan** also generates heatmaps of strain-level genetic relation.

In summary, the key characteristics are that **StrainPhlan** operates on a set of metagenomic samples and with references, and provides a strain-level phylogeny for the dominant strains of each analyzed species for each single sample. It is also worth noting that

StrainPhlan is often used alongside **PanPhlan** (Scholz et al., 2016). **PanPhlan** uses a similar approach compared to **StrainPhlan** but aims at identifying strain-specific gene content whereas **StrainPhlan** is based on nucleotide substitutions.

StrainEST (Albanese and Donati, 2017) is a reference-based tool for identifying strains and determining their abundance in metagenomic samples. While marker-based tools often assume or are able to only detect the presence of a single dominant strain, **StrainEST** is based on a set of reference genomes from a species and uses the single-nucleotide variant (SNV) profiles from them to access all strains. In fact, two reference databases are required. One for the SNV profiling and one for the metagenomic reads mapping. As stated in its corresponding paper, the choice of the sets depends on the goals of the study.

During the SNV profiling step, the Mash (Ondov et al., 2016) distance (an approximation of the mutation rate) between each pair of representative genomes is computed. Among the set of representative genomes, one is selected as a *species representative*, and other representative genomes too distant from the species representative are discarded. All remaining representative genomes are aligned to the species representative, describing a matrix of all variable positions, a SNV profile.

The Mash distance matrix is re-used to define the set of representative genomes for reads mapping through a complete linkage hierarchical clustering. For each cluster, the selected representative genome is selected as the one with the lowest average distance from the other genomes of the cluster. The number of reference genomes to include depends on the genomic variability of the species and needs specific parameters tuning, that can also be guided by *a priori* knowledge if available. Like in the previous step, the representative genomes are aligned to the species representative. The metagenomic reads are mapped onto those representative genomes.

Finally, the two previous steps are combined. For each SNV position identified in the first step, the frequency of occurrences of each nucleotide is extracted from the mapping output of the second step. The relative abundance profile is inferred by a Lasso regression.

In summary, the key characteristics are that **StrainEST** operates on a single metagenomic sample with two sets of reference genomes, and provides a strain-level abundance profile for each species. Moreover, its methodology is based on strain-specific characteristics of the core genome of a species, and as such, is not able to identify features in the dispensable genome.

DiTASiC (Fischer et al., 2017) is a reference-based tool for abundance estimation and to compute differential abundance of individual taxa in metagenomics samples. As seen in the previous chapter, recent approaches evolved to the use of k-mers over alignments to alleviate the cost of computation. However, they showed a reduction in resolution for strain-level read assignments. For this reason, DiTASiC aims at combining both by relying on pseudo-alignments for faster mapping, and by using a generalized linear model to resolve ambiguities in read assignments.

While not mandatory, DiTASiC strongly recommends a first step of pre-filtering references to start the pipeline with a set of reference genomes of species expected in the sample.

Sequencing reads from the metagenomic sample are mapped to each reference genome and the number of reads assigned to a genome defines its *mapping abundance*. Considering the high similarity between genomes from strains of the same species, some reads might map equally well on multiple genomes.

Then, a generalized linear model is used to predict the mapping abundances (response variable) from the references similarity matrix (predictor), resulting in a corrected abundance that takes into account the read count ambiguities. The differential abundance between two samples will not be further detailed here as it does not strictly relate to the thesis project. Briefly, the corrected abundances are formulated as distributions and compared assuming Poisson distributions. For each taxa, a p-value reports the significance of the difference.

In summary, the key characteristics are that DiTASiC operates on a single metagenomic sample with a set of reference genomes, and provides a strain-level abundance profile for each taxa. However, missing or unknown taxa may introduce bias, hence the recommendation of an initial step of adequate selection and pre-filtering of the references used.

`mixtureS` (X. Li et al., 2020) is a tool for *de novo* identification of bacterial strains from shotgun reads of a metagenomic sample. The rationale behind `mixtureS` is that most of the existing methods depends on known strains and/or does not work on a single sample.

Before using `mixtureS`, it is assumed that a sample has been sequenced, and the resulting reads have been mapped to a species genome. Here, it is important to clarify that, similarly to DESMAN that uses known core genes, `mixtureS` uses a type of reference, yet the *de novo* nature of the methodology refers to using sequencing reads to infer

non-identified haplotypes, and does not mean it is completely reference-free. Moreover, in order to simplify the computations behind the algorithm used, two assumptions are made. It is assumed that different strains of a species have different abundances and that a polymorphic site can only be biallelic.

mixtureS operates in three steps. Firstly, all positions in the reference revealing variations according to the mapped reads are identified. Secondly, among the detected positions, those of low-coverage are removed. Finally, an expectation-maximization algorithm is applied to infer the strains from those variations positions.

In summary, the key characteristics are that **mixtureS** is a *de novo* method, that operates on a single sample, and that seeks to resolve haplotypes and their abundance.

Several other strain-oriented tools could have been cited. For instance, **ConStrains** (Luo et al., 2015) that, similarly to **StrainPhlan**, uses **MetaPhlan2** to work on marker genes. However, **StrainPhlan** proved to produce better results in terms of overall strain-tracking accuracy compared to **ConStrains**. **MetaSNV** (Costea et al., 2017) and **inStrain** (Olm et al., 2021) compare SNVs across several samples to conclude on the existence of different strains across populations. However, they do not provide extended profiles as described for the tools previously presented. **StrainsGE** (Dijk et al., 2021) is not published yet. While it aims to characterize the strain abundances in microbial communities, the first estimated strain abundance is always biased, making further comparisons difficult. Finally, others tools like **StrainSeeker** (Roosaare et al., 2017) are used for strain identification but only on isolates, and are not applicable to complex metagenomic samples.

Finally, tools that are not specifically dedicated to strain resolution analysis may be used.

Kraken2 (Wood, Lu, et al., 2019) is one of the tools, as previously mentioned, that uses k-mer approaches to accelerate read assignments. **Kraken2** is a reference-based tool, that requires a set of reference genomes and their taxon information from the NCBI taxonomy database that form a classification tree. The reference database is compressed into a k-mers database, and the sequencing reads from a metagenomic sample are also decomposed into k-mers. Each k-mer of the read is mapped to the lowest common ancestor of the genomes in the reference database. Each node of the classification tree has a weight equal to the k-mers that mapped the associated sequence taxon. Finally, each root-to-leaf paths has a score equal to the sum of the traversed nodes weight. The classification

selected corresponds to the leaf with the maximal score path.

KrakenUniq (Breitwieser et al., 2018) works in a similar way, but focuses only on unique k-mers identified for each taxon, allowing for a better distinction between false-positive and true-positive matches.

Kraken2 and **KrakenUniq**, as they are, only provide a classification output, that is to say an information of presence/absence. To perform abundance estimation, the same authors have developed **Bracken** (Lu et al., 2017), that is complementary to **Kraken**. Unfortunately, **Bracken** can only estimate species/genus-level abundances.

This review on the state of the art related to the strain-level resolution issue, associated with their year of publication, shows how this is still an active field of research. While **Kraken2** is the popular tool for classification, there is no similar consensus tool for strain-level profiling as there is still a variety of inputs used and concerns to address (identification of known strains, inference of haplotypes, etc).

3.3 Thesis rational

In conclusion to this state of the art, graph representations are well defined, and tools to build and manipulate graphs are still under active development. A graph structure allows to reduce the data redundancy and to highlight variations, key advantages compared to the current approaches that are biased towards the references, that bias read mapping because of the high similarity of closely related sequences, or that discard sequences or some variations. Moreover, as seen with the existing strain-level profiling tools, some tools focus on the dominant strain only, whereas it has been shown that, for instance, the human microbiota is often a complex mixture of strains (Oh et al., 2016). In other cases, some tools need a set of multiple metagenomic samples, while for this thesis we are interested in profiling samples independently.

To our knowledge, no strain-level profiling tool uses graph structure despite their obvious advantages when working with similar sequences. In parallel, those new graph frameworks also arise new challenges: updating a graph with novel sequences, adapting existing efficient algorithms for read mapping, and, directly related to the concern of exploring strain-level resolution, developing new ways to analyse sequence-to-graph mapping results for downstream analyses.

For these reasons, and in order to address the thesis objectives, we developed **StrainFLAIR**, a tool that uses variation graph representation of gene sequences for strain identification and quantification.

STRAINFLAIR: STRAIN-LEVEL PROFILING OF METAGENOMICS SAMPLE USING VARIATION GRAPHS

The fourth chapter of this thesis aims at describing the major contribution of my work. This has been published and the chapter is adapted from the resulting scientific paper (Da Silva et al., 2021).

We present **StrainFLAIR**, a novel method and its implementation that uses variation graph representation of gene sequences for strain identification and quantification. We proposed novel algorithmic and statistical solutions for managing ambiguous alignments and computing an adequate abundance metric at the graph node level. Results on simulated data and on real sequencing data have shown that we could correctly identify and quantify strains present in a sample. Notably, in the controlled experimental design that we investigated, we could also detect the existence of a strain close to, but absent from those in the reference.

StrainFLAIR is available at <http://github.com/kevsilva/StrainFLAIR>

4.1 Pipeline

4.1.1 Overview

We propose here a description of our tool **StrainFLAIR** (STRAIN-level proFiLing using vArIation gRaph). This method exploits various state-of-the-art tools and proposes novel algorithmic solutions for indexing bacterial genomes at the strain-level. It also permits to query metagenomes for assessing and quantifying their content, in regards to the indexed genomes. An overview of the index and query pipelines are presented on Figure 4.1.

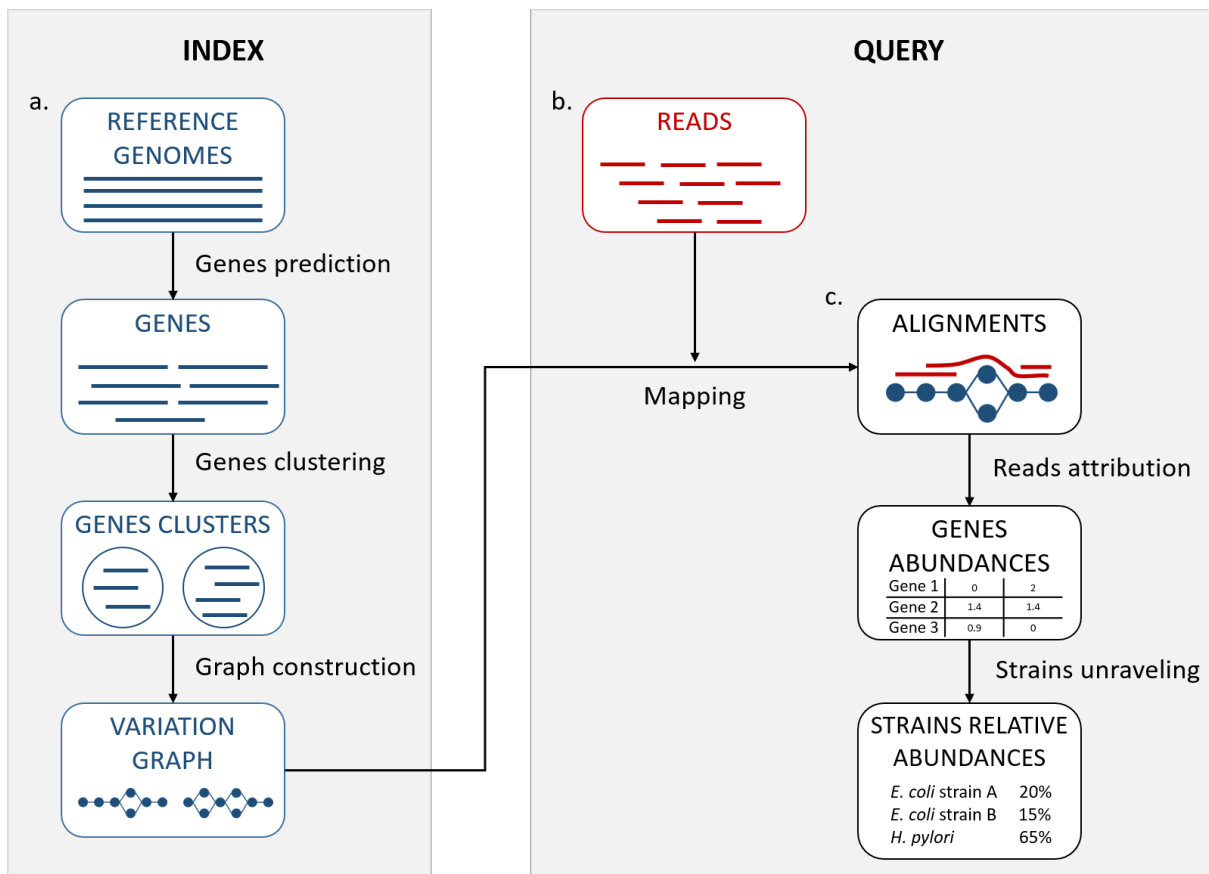


Figure 4.1 – **StrainFLAIR overview.**

a. Indexing. Input is a set of known reference genomes of various bacterial species and strains. **StrainFLAIR** uses a graph for indexing genes of those reference genomes. **b. Read mapping** on the previously mentioned graph. **c. Mapped reads analysis.** **StrainFLAIR** assigns and estimates species and strain abundances of a bacterial metagenomic sample represented as short reads.

In a few words, **StrainFLAIR** works as follows: First, it indexes genes of input reference genomes. Similar genes from several genomes are grouped into a gene family. Each gene family is represented as a part (a connected component) of a variation graph. The path described in this variation graph by the sequence of any gene of any indexed genome is called a “colored-path”. Note that, conversely, any path of the variation graph does not necessarily correspond to an indexed gene. At query time, the mapping of a queried read on the graph results on a subset of the graph in which each mapped nodes is associated with a mapping score. This set of nodes is called a “*multipath-alignment*”. From a multipath-alignment we extract a set of so called “*single-path-alignments*” that are paths with a mapping score higher than a threshold.

Then, in a step called “*colored-path attribution*”, each of the previously determined single-path-alignments is, when possible, attributed to the most probable colored-path of the variation graph, hence determining to which input genome the mapped read belongs to. Once all read are mapped, the careful analysis of mapped colored-paths enables to draw a profile to the queried metagenomic sample.

We now provide more details on each of the **StrainFLAIR** steps.

4.1.2 Indexing strains

Gene prediction

As non-coding DNA represents 15% in average of bacterial genomes and is not well characterized in terms of structure, **StrainFLAIR** focuses on protein-coding genes in order to characterize strains by their gene content and nucleotidic variations of them. Moreover, non-coding DNA regions can be highly variable (Thorpe et al., 2017) and taking into account complete genomes would then lead to highly complex graphs, and combinatorial explosions when mapping reads. Additionally, complete genomes are not always available. Focusing on the genes allows to use also drafts and metagenome-assembled genomes or a pre-existing set of known genes (Qin et al., 2010; J. Li et al., 2014).

Hence, **StrainFLAIR** indexes genes instead of complete genomes in graphs.

Genes are predicted using **Prodigal** (Hyatt et al., 2010), a tool for prokaryotic protein-coding genes prediction.

Knowing that some reads map at the junction between the gene and intergenic regions, by conserving only gene sequences, mapping results are biased towards deletions and drastically lower the mapping score. In order to alleviate this situation, we extend the predicted gene sequences at both ends. Hence, **StrainFLAIR** conserves predicted genes plus their surrounding sequences. By default, and if the sequence is long enough, we conserve 75 bp on the left and on the right of each gene.

Gene clustering

Genes are clustered into gene families using **CD-HIT** (W. Li and Godzik, 2006), similar to the pipeline used in the IGC construction seen in Chapter 2. For the clustering step, the genes without extensions are used in order to strictly cluster according to the exact gene sequences and no parts of intergenic regions. **CD-HIT-EST** is used to realize the clustering with an identity threshold of 0.95 and a coverage of 0.90 on the shorter sequence. The local sequence identity is calculated as the number of identical bases in alignment divided by the length of the alignment. Sequences are assigned to the best fitting cluster verifying these requirements.

Graph construction

Each gene family is represented as a variation graph (see Figure 4.2). As a reminder, variation graphs are bidirected DNA sequence graphs that represent multiple sequences, including their genetic variations. Each node of the graph contains sub-sequences of the input sequences, and successive nodes draw paths on the graph. Paths corresponding to reference sequences are specifically called “colored-paths”. Each colored-path corresponds to the original sequences of a gene in the cluster.

In the case of a cluster composed of only one sequence, **vg toolkit** (Garrison, A. Novak, et al., 2017) is used to convert the sequence into a flat graph. Alternatively, when a cluster is composed of two sequences or more, **minimap2** (H. Li, 2018) is used to generate pairwise sequence alignments. Then **seqwish** (Garrison, 2022) is used to convert these pairwise sequence alignments into a variation graph.

vg toolkit allows to modify the graph including a normalization step. Normalization consists in deleting redundant nodes (nodes containing the same sub-sequence and having the same parent and child nodes), removing edges that do not introduce new paths, and merging nodes separated by only one edge. For each cluster, if the colored paths of the

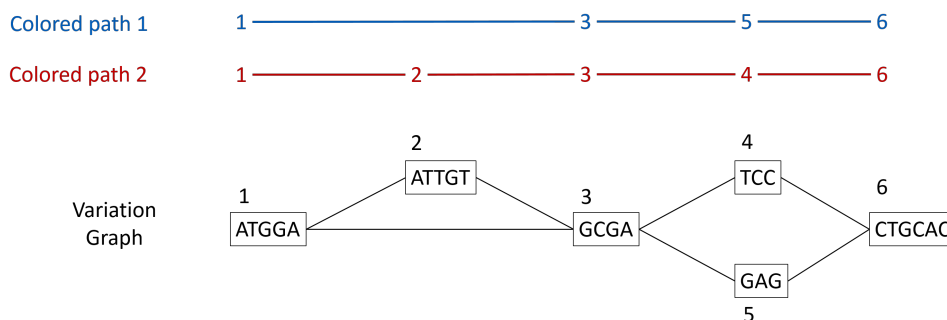


Figure 4.2 – **Illustration of a variation graph structure and colored-paths.** Each node of the graph contains a sub-sequence of the input sequences and is integer-indexed. A path corresponding to an input sequence is called a colored-path, and is encoded by its succession of node ids, e.g. 1,3,5,6 for the colored-path 1 in this example.

corresponding graph still describe their respective input sequences, the graph is normalized.

All the so-computed graphs (one per input cluster) are then concatenated to produce a single variation graph where each cluster of genes is a connected component.

After the concatenation of all computed graphs (one for each cluster), the final single variation graph is indexed using `vg toolkit`. Indexing a graph allows a fast querying of the graph when mapping reads. Indexing uses two file formats: *XG*, which is a succinct graph index which presents a static index of nodes, edges and paths of a variation graph, and *GCSA*, a generalized FM-index to directed acyclic graphs. A *SNARLS* file is also generated, describing snarls (a generalization of the superbubble concept; Paten, Eizenga, et al., 2018) in the variation graph and similarly allowing faster querying.

The index is created once for a set of reference genomes. Afterward, any set of sequenced reads can be queried at the strain-level based on this index.

4.1.3 Querying variation graphs

The so-created variation graphs is queried by sequencing reads. Each read is mapped on the graph. Then each mapped read is associated, when possible, to a gene of one of the indexed genome. This is the “*read attribution*” step, itself composed of the “*single-*

path-alignments attribution” and the “*colored-path attribution*” steps, detailed below.

Mapping reads

To map reads on the previously described reference graph, we used the sequence-to-graph mapper `vg mpmc` from `vg toolkit`. It produces a so-called “multipath-alignment”. A multipath-alignment is a graph of partial alignments and can be seen as a sub-graph (a subset of edges and vertices) of the whole variation graph (see Figure 4.3). The mapping result describes, for each read, the nodes of the variation graph traversed by the alignment and the potential mismatches or indels between the read and the sequence of each traversed node.

The mapping results are given in *GAMP* format, then converted into *JSON* format with `vg toolkit`, describing, for each read, the nodes of the graph traversed by the alignment.

Reads attribution

When mapping a read on a graph with colored-paths, two key issues arise, as illustrated on Figure 4.3. As mapping generates a sub-graph per mapped read, the most probable mapped path(s) have to be defined. Meanwhile, the most probable mapped path(s) corresponding to a colored-path also have to be defined.

Hence we developed an algorithm to analyse and convert, when possible, a mapping result into one or several single-path-alignments (successive nodes joined by only one edge) per mapped read. In addition, we propose an algorithm to attribute each such single-path-alignment to most probable colored-path(s).

Single-path-alignments attribution.

A breadth first search on the multipath-alignment is proposed. It starts at each node of the alignment with a user-defined threshold on the mapping score. A single-path-alignment with a mapping score below this threshold is ignored, and the single-path-alignment with the best mapping score is retained. Additionally, for each alignment, nodes are associated with a so-called “horizontal coverage” value. The horizontal coverage of a node by a read corresponds to the proportion of bases of the node covered by the read.

Hence, a node has an horizontal coverage of 1 if all its nucleotides are covered by the read with or without mismatches or indels.

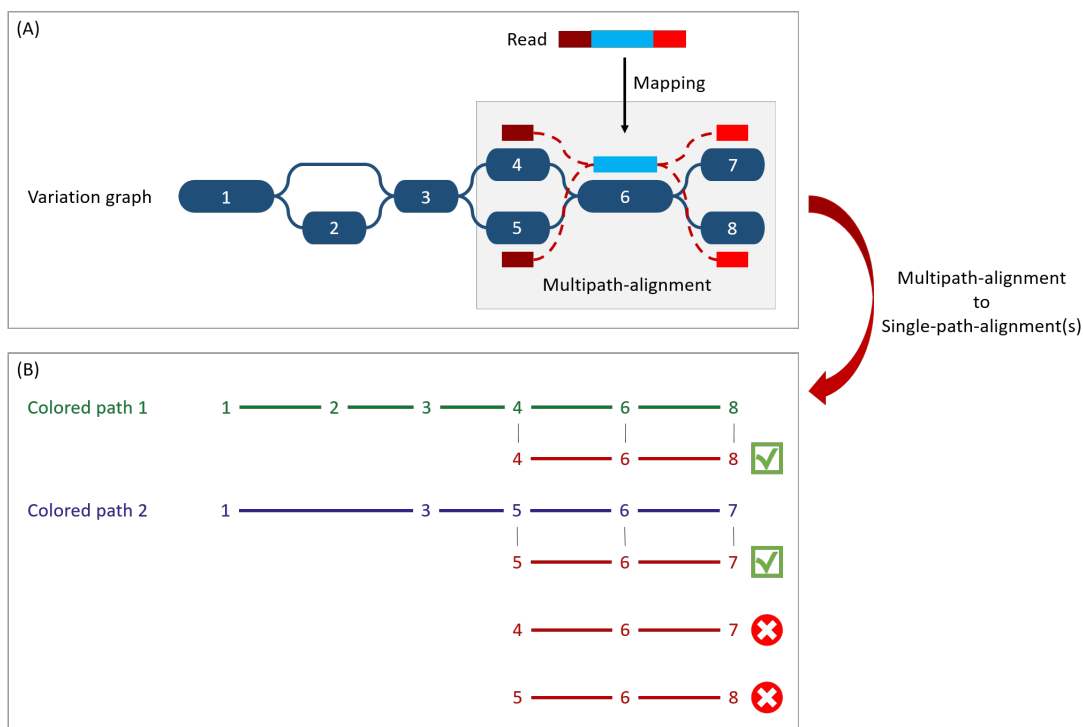


Figure 4.3 – **Illustration of the multipath-alignment concept and the read attribution process.** The region of the read in blue aligns un-ambiguously to a node of the graph while the dark and light red parts can either align to the top or the bottom nodes of their respective mapping localization (due to mismatches that can align on both nodes for example), drawing an alignment as a sub-graph of the reference variation graph, and thus opening the possibility of four single-path-alignments. **(A) Single-path-alignments attribution.** First, from the multipath-alignment (all four read sub-paths), the breadth search finds the possible corresponding single-path-alignment(s) while respecting the mapping score threshold imposed by the user. Here, for the example, all four possible paths are considered valid. **(B) Colored-path attribution.** Second, each single-path-alignment is compared to the colored-paths from the reference variation graph. Two single-path-alignments matched the colored-paths (4-6-8 and 5-6-7). As it mapped equally more than one colored-path, this read is not processed during the first step of the algorithm which focuses on reads mapping uniquely on a single colored-path, but falls in the multiple mapped reads case which is processed during the second step and will be considered shared by both matched colored-paths.

Because of possible ties in mapping score, the search can result in multiple single-path-alignments, as illustrated Figure 4.3(A). This situation corresponds to a read with a sequence found in several different genes or to a read mapping onto the similar region of different versions of a gene.

To take into account ambiguous mapping affectations, as shown below, the parsing of the mapping output is decomposed into two steps. The first step processes the reads that mapped only a unique colored-path (called “unique mapped reads” here), corresponding to a single gene. The second step processes the reads with multiple alignments (called “multiple mapped reads” here).

Colored-path attribution.

Once a read is assigned to one or several single-path-alignments, it still has to be attributed, if possible, to a colored-path.

The following process attributes each mapped read to a colored-path and various metrics for downstream analyses are computed. In particular, an absolute abundance for each node of the variation graph, called the “node abundance”, is computed, first focusing on **unique mapped reads** (first step). For a given single-path-alignment, the successive nodes composing this path are compared to the existing colored-paths of the variation graph.

If the alignment matches part of a colored-path, the number of mapped reads on this path is incremented by one (i.e. reads raw count). The node abundance for each node of the alignment is incremented with its horizontal node coverage defined by this alignment. Alignments with no matching colored-paths are skipped.

Then, we focus on **multiple mapped reads** (second step), as illustrated Figure 4.3(B).

During this step, a single-path-alignment matches multiple colored-paths. Hence, the abundance is distributed to each matching colored-path relatively to the ratio between them. This ratio is determined from the reads raw count of each path from the first step. For example, if 70 unique mapped reads were found for path1 and 30 for path2 during the first step, a read matching ambiguously both path1 and path2 during the second step counts as 0.7 for path1 and 0.3 for path2. This ratio is applied to increment both the raw count of reads and the coverage of the nodes.

Gene-level and strain-level abundances

StrainFLAIR output is decomposed into an intermediate result describing the queried sample and gene-level abundances, and the final result describing the strain-level abundances.

Gene-level.

After parsing the mapping result, the first output provides information for each colored-path, i.e. each version of a gene. Thereby, this first result proposes gene-level information including abundances. For each colored path, **StrainFLAIR** provides the following items:

- The corresponding gene identifier.
- For each reference genome, the number of copies of the gene. Since each unique version of a gene is represented once in the graph, whereas it can exist in several copies in the genome (duplicate genes), the counts and abundances computed correspond to the sum of those copies. Keeping track of the number of copies is important to normalize the counts.
- The cluster identifier to which the colored path belongs.
- For unique mapped reads: their raw number and their number normalized by the sequence length (further detailed below).
- For unique plus multiple mapped reads: their raw number and their number normalized by the sequence length (further detailed below).
- The mean abundance of the nodes composing the path.
- The mean abundance without the nodes of the path never covered by a read.
- The ratio of covered nodes.

We further detail here the three major metrics outputted by **StrainFLAIR**.

The **mean abundance of the nodes composing the path**. Instead of solely counting reads, we make full use of the graph structure and we propose abundances computation for each node as previously explained, and as already done for haplotype resolution (Baaijens et al., 2019). Hence, for each colored-path, the gene abundance is estimated by the mean of the nodes abundance.

In order to not underestimate the abundance in case of a lack of sequencing depth (which could result in certain nodes not being traversed by sequencing reads), the **mean abundance without the nodes of the path never covered by a read** is also outputted.

The mean abundance with and without these non-covered nodes are computed using unique mapped reads only or all mapped reads.

The **ratio of covered nodes**, defined as the proportion of nodes from the path with an abundance strictly greater than zero.

Strain-level.

A colored-path associated to only one strain is called “strain-specific”. Strain-level abundances are obtained by exploiting the results of reads mapped on strain-specific colored-paths.

First, for each genome, the proportion of detected genes is computed, as the proportion of specific genes on which at least one read maps. Then, the global abundance of the genome is computed as the mean or median of all its specific gene abundances. However, if the proportion of detected genes is less than a user-defined threshold, the genome is considered absent and hence its abundance is set to zero.

StrainFLAIR final output is a table where each line corresponds to one of the reference genomes, containing in columns the proportion of detected specific genes, and our proposed metrics to estimate their abundances (using mean or median, with or without never covered nodes as described for the gene-level result).

As presented in Section 4.2.3, we validated and motivated the proposed abundance metric by comparing it to the expected abundances and other estimations using linear models.

4.2 Validation

We validated our method on both a simulated and a real dataset. All computations were performed using **StrainFLAIR**, version 0.0.1, with default parameters. The relative abundances estimation was based on the mean of the specific gene abundances, computed by taking into account all the nodes (including non-covered nodes), and using a 50% threshold on the proportion of detected specific genes.

The presented results are compared to **Kraken2** considered as one of the state-of-the-art tool dedicated to the characterization of read set content, and based on flat sequences as references. Read counts given by **Kraken2** were normalized by the genome length and con-

verted into relative abundances. Other tested tools either suffer from unfair comparisons as their features differ from **StrainFLAIR** or show weaker results than those obtained by **Kraken2**. Considering **StrainFLAIR** was designed to query a single sample, **DESMAN** was not suitable for this work as it needs multiple samples in order to compute variant co-occurrences. Similarly, considering **StrainFLAIR** was designed to compute strain relative abundances, **PanPhlan** and **StrainPhlan** were not suitable as they do not provide such output. **StrainEst**, **DiTASiC**, **KrakenUniq** and **mixtureS** had similar inputs and outputs compared to **StrainFLAIR**. Those tools were tested on two of the simulated datasets described in the following. It was enough to highlight their main differences with **StrainFLAIR** (see Section 4.2.1).

Here we present a proof of concept of the variation graph application for the microbial strain detection. While the aim is not to provide a benchmark of the state-of-the-art tools, computing setup and performances are also indicated (see Section 4.2.4).

4.2.1 Validation on a simulated dataset

We first validated our method on simulated data, focusing on a single species with multiple strains. Our aim was to validate the **StrainFLAIR** ability to identify and quantify strains given sequencing data from a mixture of several strains of uneven abundances, and with one of them absent from the index.

Results presented in this section can be reproduced using data and commands available from the github website.

Reference variation graph

We selected complete genomes of *Escherichia coli*, a predominant aerobic bacterium in the gut microbiota (Tenailon et al., 2010), and a species known for its phenotypic diversity (pathogenicity, antibiotics resistance) mostly resulting from its high genomic variability (Dobrindt, 2005).

Eight strains of *E. coli* were selected for this experiment from the NCBI¹. Seven were used to construct a variation graph (*E. coli* IAI39, O104:H4 str. 2011C-3493, str. K-12

1. [https://www.ncbi.nlm.nih.gov/genome/?term=txid562\[orgn\]](https://www.ncbi.nlm.nih.gov/genome/?term=txid562[orgn])

substr. MG1655, SE15, O157:H16 str. Santai, O157:H7 str. Sakai, O26 str. RM8426), and one was used as an unknown strain in a strains mixture (*E. coli* BL21-DE3).

For ease of reading, in the following, K-12 substr. MG1655 is simply designed by “K12” and BL21-DE3 is designed by “BL21”.

Distance between the selected genomes

We estimated the distance between the complete genomes of the selected strains using **fastANI** (Average Nucleotide Identity; Jain, Rodriguez-R, et al., 2018). **FastANI** uses an alignment-free algorithm to estimate the average nucleotide identity between pairs of sequences.

	K-12	IAI39	O104:H4	Sakai	SE15	Santai	BL21-DE3	RM8426
K-12	100	97.0652	98.3769	97.8703	96.8716	98.0362	98.9365	98.3657
IAI39	97.037	100	96.9742	96.7417	97.1289	96.9295	97.0197	96.8987
O104:H4	98.3059	96.9521	100	97.4788	96.8007	97.8896	98.249	98.7212
Sakai	97.7497	96.8627	97.5094	100	96.6657	98.1523	97.7455	97.6125
SE15	96.8453	97.1064	96.9211	96.7362	100	96.7575	96.8141	96.7763
Santai	98.0073	97.0372	97.9584	98.1797	96.8199	100	97.9279	97.9077
BL21-DE3	98.9983	97.1721	98.4048	97.8227	96.8448	97.9616	100	98.3204
RM8426	98.306	96.9037	98.6801	97.5815	96.6907	97.8353	98.2567	100

Table 4.1 – **Distance between each pair of complete genome sequences from eight strains of *E. coli* as computed by fastANI.**

All pairs showed a distance at least greater than 95%, highlighting the strong similarities between the strains. As a threshold, we although considered that beyond 99%, sequences were too similar to be considered and distinguished, additionally to the effect of sequencing errors. The **fastANI** results showed that none of the pairs exceeded this similarity threshold.

The strain BL21 was chosen as the unknown strain while the seven others would be used to build the reference variation graph. According to the results of **fastANI**, the strain BL21 closest genome in the present references is the strain K-12 with a similarity of 98.9%. Hence we expected to find evidences of the strain K-12 while analyzing a sample containing the unknown strain BL21.

Mixtures and sequencing simulations

Our aim was to simulate the co-presence of several *E. coli* strains. Mixtures of three strains were used to mimic complex single species composition in metagenomic samples. We simulated short sequencing reads of 150 bp using `vg sim` from `vg toolkit` with a probability of sequencing errors set to 0.1%. Two batches of simulations were conducted in order to highlight the detection and quantification of strains in the mixture. The first simulation was a mixture composed of strains indexed in the reference graph (O104:H4, IAI39 and K-12) while the second simulation (O104:H4, IAI39 and BL21) had one absent from the reference variation graph (BL21) thus simulating a strain absent from the reference graph to be identified and quantified. For each simulation, we tested our `StrainFLAIR` with various read coverage (see Table 4.2), with K-12 or BL21 in equal abundance of IAI39, potentially making it more difficult to distinguish, or in lower abundance, potentially making it more difficult to detect at all.

Samples	O104:H4	IAI39	K-12 or BL21
1	300,000 (8.5x)	200,000 (5.8x)	200,000 (6.5x)
2			100,000 (3x)
3			50,000 (1.6x)
4			25,000 (0.8x)
5			10,000 (0.3x)
6			5,000 (0.2x)
7			1,000 (0.03x)

Table 4.2 – **Composition of the mixtures described in number of reads simulated and the corresponding coverage (in parentheses).** For each simulation (including either K-12, indexed in the variation graph, or BL21, not indexed), seven mixtures were simulated.

Strain-level abundances

As explained in Section 4.1, we computed the strain-level abundances using the specific gene-level abundance table obtained by mapping the simulated reads onto the variation graph. We compared our results to the expected simulated relative abundances.

Simulation 1: mixtures with K-12, present in the reference graph

`StrainFLAIR` successfully estimated the relative abundances of the three strains present in the mixture (see Table 4.3), the sum of squared errors between the estimation given

#reads K-12	Method	O104:H4	IAI39	K-12	Sakai	SE15	Santai	RM8426
1,000	Expected	59.88	39.92	0.2	0	0	0	0
	StrainFLAIR	56.47 (0.995)	43.53 (0.989)	0 (0.309)	0 (0.189)	0 (0.151)	0 (0.188)	0 (0.212)
	Kraken2	38.91	60.72	0.22	0.04	0.07	0.03	0.02
25,000	Expected	57.14	38.1	4.76	0	0	0	0
	StrainFLAIR	52.14 (0.994)	40.58 (0.989)	7.27 (0.878)	0 (0.208)	0 (0.153)	0 (0.215)	0 (0.234)
	Kraken2	37.23	58.1	4.51	0.04	0.07	0.03	0.02
200,000	Expected	42.86	28.57	28.57	0	0	0	0
	StrainFLAIR	38.12 (0.993)	29.81 (0.988)	32.08 (0.99)	0 (0.211)	0 (0.159)	0 (0.219)	0 (0.237)
	Kraken2	28.31	44.18	27.35	0.04	0.08	0.03	0.02

Table 4.3 – **Reference strains relative abundances expected and computed by StrainFLAIR or Kraken2 for each simulated experiment with variable coverage of the K-12 strain.** Best results are shown in bold. For StrainFLAIR, the proportion of specific genes detected is shown in parentheses. Complete results are presented in the Appendix, Table A1.

by our tool and the expected relative abundance was between 25 and 45 for all the experiments. However, it did not detect the very low abundant strain in the case of the mixture with 1,000 simulated reads for K-12 (coverage of $\approx 0.03x$).

With our methodology, the threshold on the proportion of detected genes (see Section 4.1) lead to set relative abundance to zero of likely absent strains. This reduces both the underestimation of the relative abundances of the present strains and the overestimation of the absent strains.

In comparison, Kraken2 did not provide this resolution. Applied to our simulated mixtures, while Kraken2 was slightly better for K-12 abundance estimation, it overestimated IAI39 relative abundance and underestimated O104’s one, leading to an overall higher sum of squared errors (between 456 and 872) compared to the expected abundances. Moreover, it set relative abundances to all the seven reference strains whereas four of them were absent from the mixture. This was expected as some reads (from intergenic regions for example) can randomly be similar to regions of genes from absent strains.

Simulation 2: mixtures with BL21, absent from the reference graph

Here, BL21 was considered an unknown strain, not contributing to the variation graph. The closest strain of BL21 in the graph, according to fastANI, was K-12 (98.9% of identity, see Table 4.1). Thus we expected to find signal of BL21 through the results on K-12.

#reads BL21-DE3	Method	O104:H4	IAI39	K-12	Sakai	SE15	Santai	RM8426
1,000	Expected	59.88	39.92	0.2*	0	0	0	0
	StrainFLAIR	56.48 (0.995)	43.52 (0.989)	0 (0.254)	0 (0.189)	0 (0.151)	0 (0.192)	0 (0.214)
	Kraken2	38.93	60.76	0.11	0.05	0.08	0.04	0.03
25,000	Expected	57.14	38.1	4.76*	0	0	0	0
	StrainFLAIR	54.12 (0.995)	41.72 (0.989)	4.16 (0.584)	0 (0.266)	0 (0.177)	0 (0.282)	0 (0.298)
	Kraken2	37.75	58.93	2.16	0.28	0.34	0.25	0.29
200,000	Expected	42.86	28.57	28.57*	0	0	0	0
	StrainFLAIR	46.96 (0.993)	35.32 (0.988)	17.72 (0.711)	0 (0.318)	0 (0.211)	0 (0.346)	0 (0.351)
	Kraken2	31.14	48.83	13.53	1.57	1.67	1.58	1.68

Table 4.4 – **Reference strain relative abundances expected and computed by StrainFLAIR or Kraken2 for each simulated experiment with variable coverage of the BL21 strain, absent from the reference variation graph.** BL21 strain expected abundances are followed by an asterisk in the K-12 column. Best results are shown in bold. For StrainFLAIR, the proportion of specific genes detected is shown in parentheses. Complete results are presented in the Appendix, Table A2.

As with the K-12 mixtures, StrainFLAIR successfully estimated the relative abundances of the two known strains present in the mixture (see Table 4.4), the sum of squared errors between the estimation given by our tool and the expected relative abundance was between 22 and 180 for all the experiments. Labelled as K-12, it also gave close estimations for BL21 in this controlled experimental design. Again, it did not detect the very low abundant strain in the case of the mixture with 1,000, 5,000, and 10,000 simulated reads for BL21. Also similarly to the K-12 mixtures experiments, Kraken2 overestimated IAI39 relative abundance and underestimated O104’s one (sum of squared errors between 751 and 873), even less precisely than in the previous experiment. With sufficient coverage (here from the 0.8x for BL21), StrainFLAIR was closer to the expected values for all the reference strains than Kraken2.

Interestingly, the proportion of detected specific genes for each strain (see Figure 4.4) seems to highlight a pattern allowing to distinguish - in this specific experiment - present strains, absent strains and likely new strains close to the reference in the graph. According to the experiments with enough coverage (from 25,000 simulated reads for BL21), three groups of proportions could be observed: proportion of almost 100% (O104:H4 and IAI39 : strains present in the mixtures and in the reference graph), proportion under 30-35% (Sakai, SE15, Santai, and RM8426 : strains absent from the mixtures), and an in-between proportion around 60-70% for K-12 (closest strain to BL21).

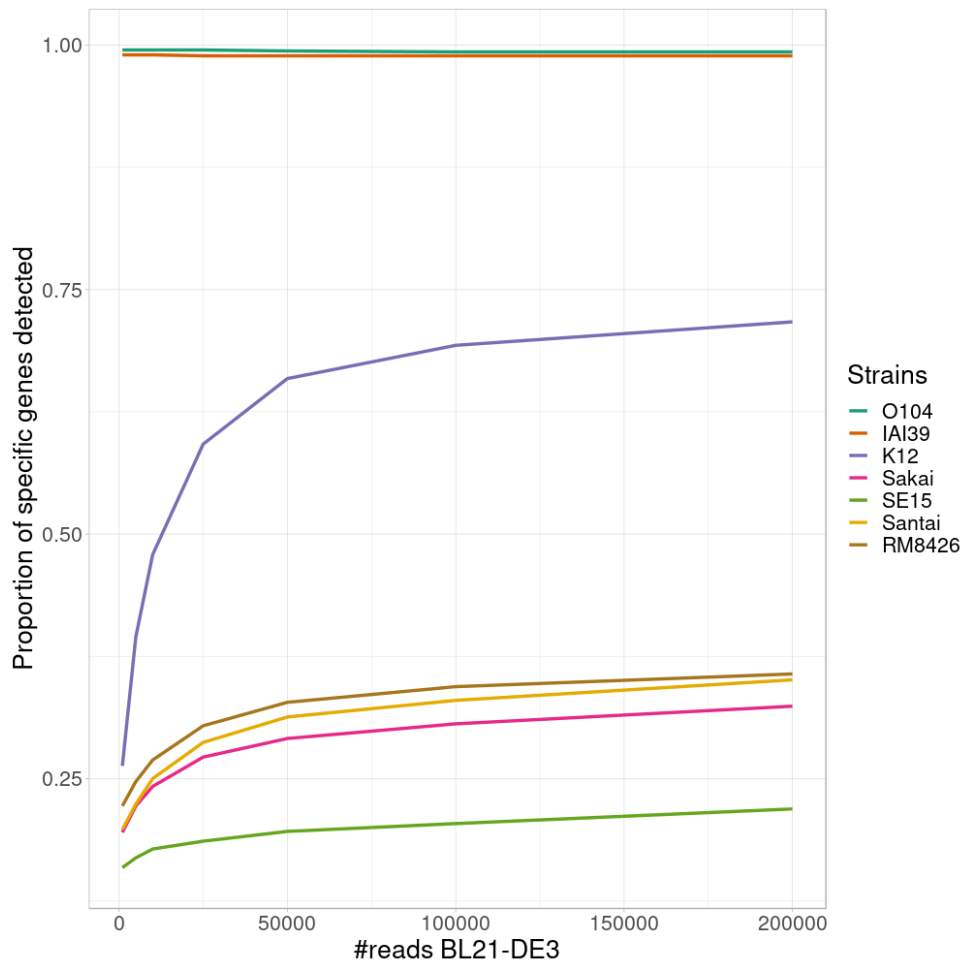


Figure 4.4 – Proportion of detected specific genes for each simulated experiment with variable coverage of the BL21 strain, absent from the reference graph.

It was expected that an absent strain would have specific genes detected as **StrainFLAIR** detects a gene once only one read mapped on it. However, all absent strains had a proportion at around 30% except K-12 which proportion was twice higher. Conjointly with the non-null abundance estimated for the reference K-12, this suggests the presence of a new strain whose genome is highly similar to K-12.

Comparison to other tools

As previously mentioned, other strain-oriented tools have been tested.

Mixture	Method	O104:H4	IAI39	K-12	Sakai	SE15	Santai	RM8426
with K-12	Expected	50	33.33	16.67	0	0	0	0
	StrainFLAIR	44.66	35.05	20.29	0	0	0	0
	StrainEst	48.64	32.97	18.39	0	0	0	0
	DiTASiC	50.27	33.38	16.35	0	0	0	0
	Kraken2	32.8	51.19	15.85	0.04	0.07	0.03	0.02
	KrakenUniq	38.27 (0.99)	26.14 (0.50)	15.28 (0.93)	5.08 (0.0017)	5.08 (0.0017)	5.08 (0.0017)	5.08 (0.0014)
with BL21	Expected	50	33.33	16.67*	0	0	0	0
	StrainFLAIR	50.47	38.64	10.89	0	0	0	0
	StrainEst	56.65	36.71	0	0	0	0	6.64
	DiTASiC	53.34	34.72	8.52	0.66	0.03	1.06	1.67
	Kraken2	34.53	54.03	7.68	0.91	0.98	0.91	0.96
	KrakenUniq	27.9 (0.99)	19.24 (0.50)	11.12 (0.34)	10.1 (0.02)	10.42 (0.02)	10.28 (0.03)	10.94 (0.04)

Table 4.5 – **Reference strains relative abundances expected and computed by StrainFLAIR or other tools for each simulated experiment.** BL21-DE3 being similar at 98.9% to K-12 strain, we expect that reads from BL21-DE3 will map this strain, hence its expected value is followed by an asterisk, as it corresponds to BL21-DE3 strain abundance and not K-12. For **KrakenUniq**, additionally to the relative abundances computed from the average number of times each unique k-mer has been seen, the coverage value of the k-mers of the clade in the database was added in parenthesis. Best results are shown in bold.

StrainEst

Similarly to **StrainFLAIR**, **StrainEst** uses a set of reference genomes. *E. coli* K-12 MG1655 was used as the species reference needed in the **StrainEst** pipeline. It was also added for the clustering step of the representative genomes. The output is a relative abundance associated to each reference genome. Results are presented in Table 4.5.

While **StrainEst** gives slightly closer relative abundance estimations to the expected ones when the three strains from the mixture are represented in the references, it does not

perform well with the mixture composed of an unknown strain (BL21-DE3). Aside from the relative abundances values that are farther than the ones provided by **StrainFLAIR**, the main issue is that **StrainEst** assigns an abundance to the strain RM8426 and not K-12 which is the closest strain to BL21-DE3 and thus expected to capture the signal.

DiTASiC

Similarly to **StrainFLAIR**, **DiTASiC** uses a set of reference genomes. The output is a table of read count estimates for each reference genomes associated with a standard error and p-value for those estimates. Read counts have been converted into relative abundances (percentages). Results are presented in Table 4.5.

While **DiTASiC** gives accurate relative abundance estimations when the three strains from the mixture are represented in the references, it does not perform well with the mixture composed of an unknown strain (BL21-DE3). Although in lower abundance than the three present strains in the sample, the absent strains are considered present even considering the p-values associated with the read count estimates, except for the strain SE15 (p-value = 0.55).

KrakenUniq

KrakenUniq assesses the coverage of unique k-mers found in each species in a dataset. It has been used by building a custom database containing the same set of reference genomes as with **StrainFLAIR**. The output is a table of, among others, the average number of times each unique k-mer has been seen, and the coverage of the k-mers of the clade in the database, for each reference genome and their higher taxonomic levels. The number of times each unique k-mer has been seen has been converted into relative abundances (percentages). Results are presented in Table 4.5 with the coverage in parentheses.

Coverage values show a high discrimination between present and absent strains, with absent strains being in less than 0.1% in coverage. By using a threshold on this coverage, discarding the false-positive strains (Sakai, SE15, Santai and RM8426), the relative abundances computed are close to expected. However, IAI39 has a coverage of 0.5 while the two other present strains are at over 0.9, which could mislead the conclusion of IAI39 being the exact strain present in the sample, as it can be observed for the simulation with BL21-DE3 reads, the coverage associated with K-12 (0.34) is also higher than the absent strains and lower compared to present strains.

KrakenUniq was also used on the mock dataset and showed similar results compared to

Kraken2 (sum of squared errors around 16 between **KrakenUniq** and **Kraken2**) except for two genomes which were drastically lower in abundance and close to abundances estimated by **Kraken2** for absent strains. *Desulfovibrio piger* ATCC 29098 estimated abundance was around 1,000 times lower with **KrakenUniq** compared to **Kraken2**, and *Methanobrevibacter oralis* DSM 7256 around 60 times lower.

mixtureS

mixtureS uses a single reference genome. The output is the inference of the number of haplotypes and an estimate of their relative abundance. Inferred haplotypes are not associated with known references. For both simulated datasets, **mixtureS** gave similar results with 5 haplotypes predicted with abundances between 11 and 31% overall. Thus, those results could not be matched with the ones given by **StrainFLAIR**, **StrainEst** or **DiTASiC**, and consequently did not allowed accurate estimations in terms of number of strains in the mixtures nor in terms of abundances.

4.2.2 Validation on a real dataset

We used a mock dataset available on EBI-ENA repository under accession number PRJEB42498, in order to validate our method on real sequencing data from samples composed of various species and strains. The mock dataset is composed of 91 strains of bacterial species for which complete genomes or sets of contigs are available, including plasmids. Among the species, two of them contained each two different strains. Three mixes had been generated from the mock, and we used the “Mix1A” in the following results.

Even though 20 out of 91 strains were absents in this mix, we indexed the full set of 91 genomes. This was done in order to mimic a controlled **StrainFLAIR** use case where the reference graph contains a mix of strains present and absent in the queried data. The metagenomic sample was sequenced using Illumina HiSeq 3000 technology and resulted in 21,389,196 short paired-end reads.

We compared our results to the expected abundances of each strain in the sample defined as the theoretical experimental DNA concentration proportion. As such, it has to be noted that potential contamination and/or experimental bias could have occurred and affected the expected abundances.

Strain detection

Among the 91 strains used in the reference variation graph, **StrainFLAIR** detected 65 strains. All of these 65 strains were indeed sequenced in Mix1A. Hence, **StrainFLAIR** produced no false positive. From the 26 strains considered absent by **StrainFLAIR**, 20 were not present in the sample (true negatives) and 6 should have been detected (false negatives). However, the term false negative has to be softened as the ground truth remains uncertain. Among those 6 undetected strains, all of them had theoretical abundance below 0.1%.

More precisely, among the 6 strains undetected by **StrainFLAIR**, 5 had some detected genes, but below the 50% threshold. In this case, by default, **StrainFLAIR** discards these strains. Finally, only one of the undetected strains (*Desulfovibrio desulfuricans* ND 132) should have been theoretically detected (even if its expected coverage was below 0.1%), but no specific gene was identified. Considering that **StrainFLAIR** uses a permissive definition of detected gene (at least one read maps on the gene), having strictly no specific genes detected for *Desulfovibrio desulfuricans* ND 132 suggests that this strain might in fact be absent from Mix1A. This is also supported by the result from **Kraken2** which estimated a relative abundance of $\approx 9e-5$, almost 500 times lower than the theoretical result.

As in the simulated dataset validation, **Kraken2** affected non-null abundances to all the references.

Strain relative abundances

For the estimated relative abundances, **StrainFLAIR** gave more similar results compared to the state-of-the-art tool **Kraken2** than the experimental values (see Figure 4.5). The sum of squared error between **StrainFLAIR** and **Kraken2** was around 11. **StrainFLAIR** and **Kraken2** gave similar results compared to the experimental values, with sum of squared errors of around 209 and 211 respectively.

Interestingly, *Thermotoga petrophila* RKU-1 is the only case where results from **StrainFLAIR** and **Kraken2** differs greatly, with, in addition, the theoretical abundance being in-between. Moreover, *Thermotoga* sp. RQ2 is the strain expected to be absent that **Kraken2** estimates with the highest relative abundance among the other expected absent strains, and the only one exceeding the relative abundances of two present strains. Considering the previous results on the simulated mixtures and that *Thermotoga petrophila* RKU-1 and *Thermotoga* sp. RQ2 are close species (**fastANI** around 96.6%) it could be an additional indicator of

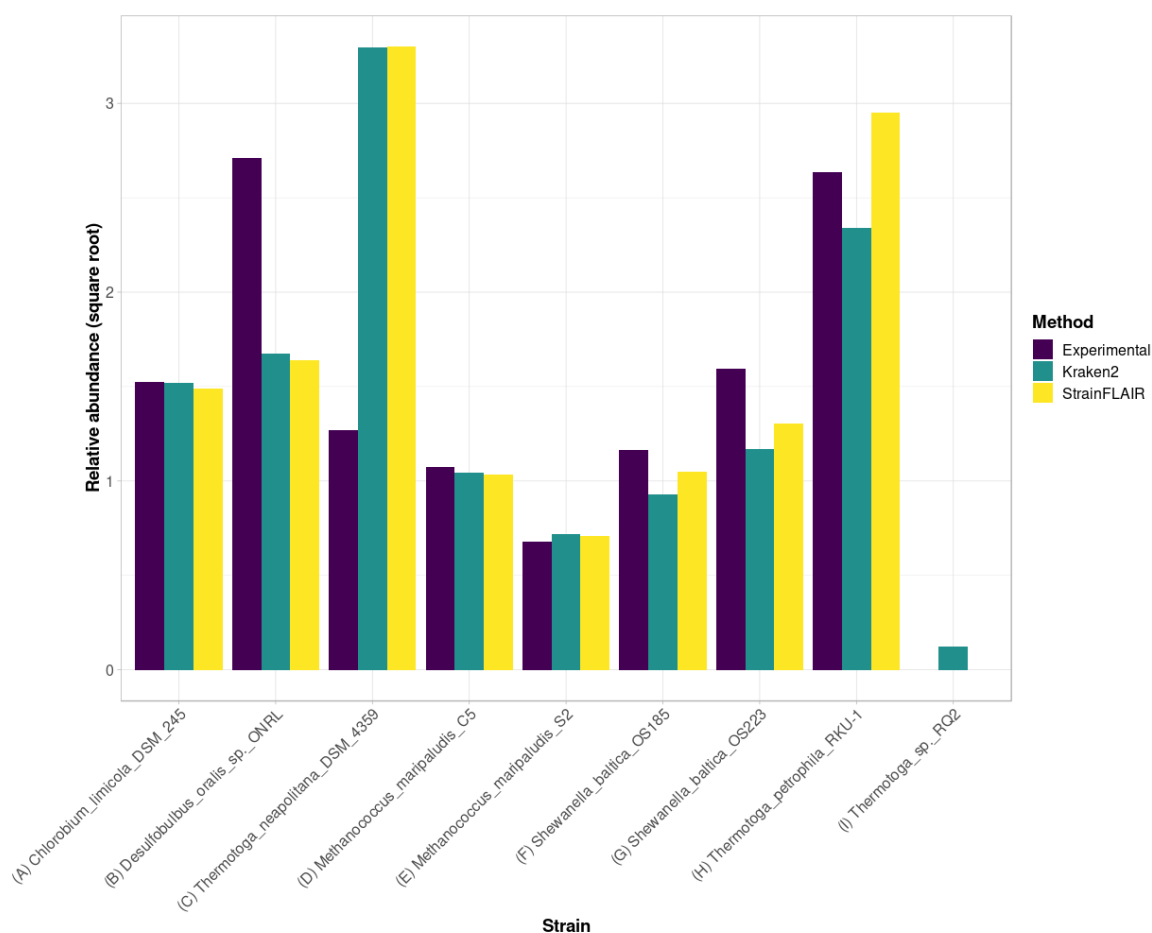


Figure 4.5 – Experimental relative abundance compared to relative abundance as computed by **StrainFLAIR** and **Kraken2**. A selection of relevant results is shown here, see the Appendix, Figure A1 for the complete results. **(A)** Represents a case where **StrainFLAIR** and **Kraken2** give similar results to the experimental value (18 cases over 91). **(B)** Represents a case where **StrainFLAIR** and **Kraken2** give similar results, but lower than the experimental value (26 cases over 91). **(C)** Represents a case where **StrainFLAIR** and **Kraken2** give similar results, but greater than the experimental value (16 cases over 91). **(D, E, F, G)** Represent the two species represented by two strains each. **(H, I)** Represent two atypical cases.

how tools like **Kraken2** can be misled by too close species or strains.

In the sample, the species *Methanococcus maripaludis* was represented by two strains (S2 and C5) and the species *Shewanella baltica* likewise (OS223 and OS185).

StrainFLAIR successfully distinguished and estimated the relative abundances of each strain of these two genomes. In this very situation and contrary to results on *E. coli* strains, **Kraken2** was also able to correctly estimate the abundances.

4.2.3 Abundance metrics validation

The output of **StrainFLAIR** provides several metrics to estimate the abundance of the genes detected in the sample.

For validation, we used a combination of LASSO (least absolute shrinkage and selection operator) model and linear model on the simulated dataset to estimate the abundances at the strain-level, as the abundance of a gene is a linear combination of the abundances of the strains it belongs to. As such, we expect no intercept value for those models and have forced the intercept at zero for the following modeling.

First, a LASSO model was used to perform strain selection. The response variable of the model was the presence or absence of the genes according to the selected metric while the strains, described as their genes content (number of copies), were the predictors. Then, a linear model was constructed with the raw selected metric as the response variable, and only the strains selected by the LASSO model as the predictors. The estimate of the strains relative abundance was thus the coefficients of the linear model associated to the strains and transformed into relative values. For each metric, the sum of squared errors between the real relative abundances and the estimated relative abundances from the linear model was computed. The best metric was then defined as the one minimizing this sum of squared errors.

For the mixtures containing *E. coli* K-12 MG1655, the three expected strains were selected and thus detected using LASSO, except for the mixture containing only 1,000 reads of K-12 MG1655 (representing 0.002% of the mixture, hence very negligible). For all the mixtures, the best metric was the mean abundance computed from the node abundances and by taking into account the multiple mapped reads.

For the mixtures containing *E. coli* BL21-DE3, BL21 being absent from the reference but very close to K-12, we expected to get some detection of K-12 in the results.

The three expected strains were selected and thus detected using LASSO, except for the mixture containing only 1,000 reads of BL21 (representing 0.002% of the mixture, hence very negligible). For the mixtures at 200,000, 100,000, and 50,000 reads of BL21, the best metric was the mean abundance computed from the node abundances without the abundances at zero, and by taking into account the multiple mapped reads. While for the others, the best metric was the mean abundance computed from the node abundances (including the abundances at zero), and by taking into account the multiple mapped reads.

This approach using linear models was particularly appropriate for this situation where the reference variation graph and the sample contained a small number of strains and thus a small number of predictors for the model. However, this can hardly transpose to a whole metagenomic sample with various species and various strains that would lead to too many predictors and probably confusing the heuristics behind the models. This was confirmed by applying the same methodology above on the mock dataset leading to abundances estimation hardly comparable to expected. Compared to `Kraken2` results, the sum of squared errors of our methodology was approximately 6 whereas for the results with the LASSO model it was around 236. Nevertheless, those results highlighted the relevance of (i) using a metric taking into account the multiple mapped reads and not only the unique mapped reads, and (ii) using our metric of abundance based on the node abundances over raw read counts.

4.2.4 Performances

Our benchmarks were performed on the GenOuest platform on a machine with 48 Xeon E5-2670 2.30 GHz with 500 GB of memory and 16 CPUs. Time results (see Table 4.6) are the wall-clock times. We provided rough computation time, mainly in the purpose to show that `StrainFLAIR` can be applied on usual datasets.

Dataset	Step	Items processed	Time	Disk used (GB)	Max mem. (GB)
Simulated	Gene prediction	7 genomes	0m20	0	1.2
	Gene clustering	34,011 genes	0m22	0	0.36
	Graph construction	8,596 clusters	2m44	0.04	1.31
	Graph concatenation	8,596 graphs	0m51	0	0.25
	Indexing graph	1 graph	6m23	0.16	4.24
	Mapping reads	350,000 short reads	15m15	0.16	0.99
	JSON conversion	1 GAMP file	3m58	4.2	0.03
	JSON parsing	1 JSON file + 1 GFA file + 1 pickle file	12m44	0	0.55
	Abundance computing	1 Gene abundances table	0m2	0	0.04
Mock	Gene prediction	91 genomes	1m43	1.02	6.7
	Gene clustering	280,174 genes	3m38	0.14	0.98
	Graph construction	270,712 clusters	41m54	1.12	9.1
	Graph concatenation	270,712 graphs	14m38	0	1.05
	Graph indexing	1 graph	75m19	1.98	30.4
	Mapping reads	21,389,196 short read pairs	147m28	7	17.5
	JSON conversion	1 GAMP file	53m21	75	0.12
	JSON parsing	1 JSON file + 1 GFA file + 1 pickle file	110m44	0	5.7
	Abundance computing	1 Gene abundances table	0m4	0	0.68

Table 4.6 – **StrainFLAIR** performances on simulated and mock datasets.

4.3 Conclusion

We developed **StrainFLAIR**, a tool for downstream analysis using a graph data structure. Our approach permits strain-level profiling of metagenomic samples, using variation graphs to represent multiple reference genomes. We validated our method on simulated datasets and a mock, and **StrainFLAIR** showed expected results compared to the theoretical ones. Results and perspectives are fully discussed in the Conclusion of the thesis.

However, besides the intended perspectives, two immediate additions to **StrainFLAIR** were considered. As seen during the description of the pipeline, sequencing reads may be attributed to multiple colored-paths, while being in fact from a single strain or a smaller subset of strains represented by the colored-paths. This introduces noise in the data, giving abundance to absent strains in the sample and/or underestimating the abundance of present strains. We expected that paired-end reads could decrease the paths ambiguities and lower the bias on the abundance estimations.

The second addition was to make use of the reads that could not match colored-paths but still match paths in the graph. Those reads are likely to originate from new strains present in the sample and unknown in the reference graph. Therefore they are crucial information to guide towards the inference of new strains.

The next chapter details those additions to **StrainFLAIR**.

TOWARDS PATHS DISAMBIGUATION AND STRAINS INFERENCE

The first release of **StrainFLAIR** set the basis of our approach using variation graphs to profile metagenomic samples at the strain level. Two immediate new developments were considered. Firstly, using paired-end reads information instead of considering them as single-end, and secondly, using the mapped reads that could not match any existing colored-path.

The fifth and last chapter of this thesis describes those new developments.

5.1 Path attributions disambiguation

5.1.1 Rational

All previous results presented in Chapter 4 were obtained using simulated single-end reads or Illumina paired-end reads each treated as single-end in **StrainFLAIR**'s algorithm. However, as seen in Section 1.1.2, paired-end reads are a way to get longer breadth of coverage and can be used to resolve ambiguities.

In this work, those ambiguities appeared during the colored-path attribution step. Reads may be shared by several colored-paths and therefore follow the shared count heuristic (second pass of **StrainFLAIR**'s algorithm). Eventually, some of those shared paths might be false positive attributions due to the high similarity between strain sequences, and would decrease the accuracy of the subsequent relative abundance computation.

By taking into account the colored-path attributions of both reads of the same pair, the list of shared colored-paths may be reduced, or all ambiguities may even be completely resolved by obtaining only one possible colored-path.

The next section details the algorithm I used in order to take into account paired-end sequencing reads.

5.1.2 Algorithm

In order to resolve path attribution ambiguities, the detailed algorithm intersects the information yield by both reads of a pair. Similarly to the single-end version, a breadth first search on the multipath-alignment is realized for both reads of a pair, still with a user-defined mapping score threshold. Then, the algorithm has different steps to search and resolve ambiguities depending on the combination of attributions held by each read (see Figure 5.1).

The first step depends on the reads alignments mapping score.

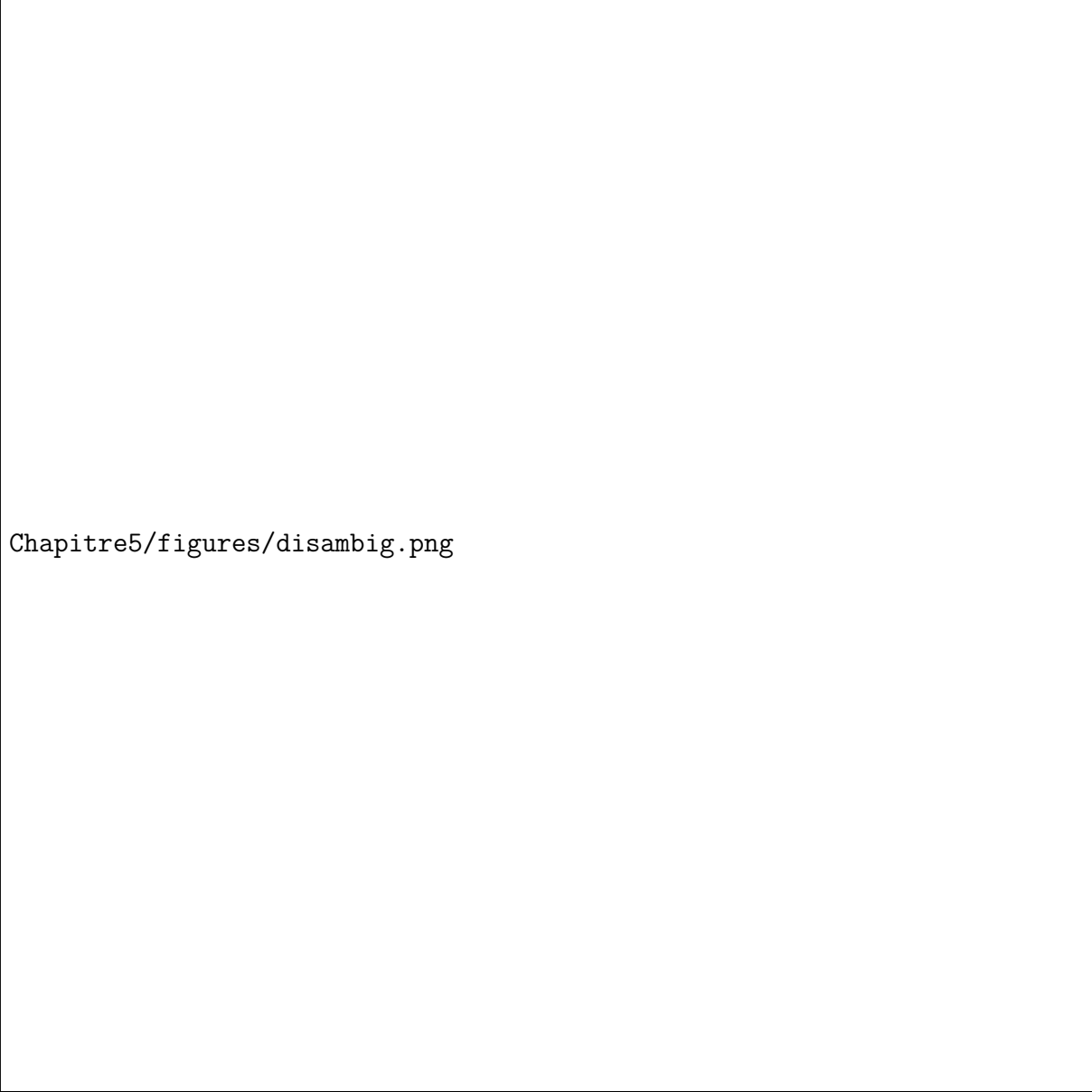
If both reads mapping score are below the user-defined threshold, the reads are removed, like in the single-end version. Since **StrainFLAIR** only indexes genes, this case especially occurs for reads corresponding to intergenic regions, misannotated genes, or unindexed genes (if some genes could not be predicted or genes other than coding DNA sequences).

If only one of the two reads does not pass the threshold, only this read is removed while the other follows the single-end version of the algorithm, exactly as described in Chapter 4. This case might occur for reads localized at the junction between an intergenic region and a gene, for example. However, we cannot exclude that it might also correspond to a pair of reads originating from a new strain not referenced in the variation graph. In this case, one read could be highly similar to a close strain while the other could have a low mapping score.

The second step of the algorithm is explored when both reads have one or more single-path-alignments exceeding the mapping score threshold.

Here, I followed the same “colored-path attribution” step previously described, in which each of the determined single-path-alignments is, when possible, attributed to the matching colored-path of the variation graph. As a result, to each read of the pair corresponds a list of matching colored-path(s).

If none of the reads matches colored-paths, they are designated as *unassigned*. Those sequencing reads might correspond to intergenic region sequences similar enough to gene sequences to pass the mapping score threshold. However, more interestingly, they could



Chapitre5/figures/disambig.png

Figure 5.1 – **Colored-path attributions disambiguation pipeline.** This approach can be decomposed into four steps. The first step checks the mapping score of the single-path-alignments, determining if the reads are removed, considered as single-end or continue to the next step. The second step checks the matching colored-paths of each read of the pair, determining if the reads are considered unassigned, as single-end or continue to the next step for potential disambiguation. The third step tries a first disambiguation by checking the intersection of the matching colored-paths from both reads. This intersection may reduce the number of potential colored-paths considered for the rest of **StrainFLAIR**'s pipeline. The fourth and last step tries another disambiguation in case of empty intersection in the previous step. For each read, the list of strains associated to the matching colored-paths are retrieved, then, the disambiguation is done by checking the intersection of these lists from both reads. This intersection may reduce the number of potential strains considered. The matching colored-paths used for the rest of **StrainFLAIR**'s pipeline are only the ones associated with the strains found in the intersection.

also correspond to sequences from new strains not referenced in the variation graph, and as such, are a key information for the inference of new strains. Those unassigned reads were used in the second feature added to **StrainFLAIR** and further detailed in Section 5.2 below. Unassigned reads are not used for relative abundances computation.

If only one of the two reads has no colored-paths, the situation is similar to the case of one of the two reads not reaching the mapping score threshold. The one with no matching colored-paths is considered as unassigned, while the other read follows the single-end version of the algorithm. It is then used for the relative abundances computation. Again, such reads might originate from intergenic regions or from a new strain.

The third step of the algorithm is performed if both reads have matching colored-paths. At this stage, the first attempt for colored-path attribution disambiguation is realized and depends on the intersection between the lists of colored-paths of both reads. Indeed, we assumed that, the reads being from the same pair and, as such, originate from the same DNA fragment, the most likely accurate colored-paths (genes) were the ones common to both reads. Hence, “false positive” colored-paths for a single read are discarded.

If the previously mentioned intersection has more than one colored-path, reads follow the second pass of the algorithm as detailed in Chapter 4, dedicated to multiple mapped reads. Once the lists of colored-paths have been updated, each read follow the same steps as single-end reads in the case of multiple mapped reads. It must be noted that, if both reads have exactly the same matching colored-paths in common, the initial list of colored-paths for each read might not change, leading to no disambiguation at all.

If the intersection has only one colored-path, there is no ambiguity on colored-path attribution and the reads are used for the relative abundances computation in the exact same way as single-end reads.

The fourth step of the algorithm is performed if the reads have no colored-paths in common, i.e. the intersection of the matching colored-paths is empty. This situation may occur if the reads originate from different genes physically close enough in the genome to be both covered by the paired-end reads. However, here also, the pair may come from a new strain. This step of the algorithm takes advantage of the references used. Since the colored-paths cannot be used for disambiguation, this means that the reads do not come from the same genes. We then consider that the reads instead come from the same strain.

For each read, the strains corresponding to their matching colored-paths are retrieved.

Hence, instead of lists of colored-paths, each read has a list of matching strains. Then, the disambiguation depends on the intersection between the lists of matching strains of both reads.

If the previously mentioned intersection has at least one strain, only the colored-paths related to the strains present in the intersection are kept. Here again, it must be noted that the initial list of colored-paths for each read might not change, leading to no disambiguation at all. This may occur if both reads have exactly the same matching strains in common. Once the lists of colored-paths have been updated (with no changes or reduced), each read follow the same steps as single-end reads.

If the intersection is empty, the reads are called *discordant* since they are considered to originate from different strains, which should not be possible. Those reads are discarded.

5.1.3 Validation

I validated my method with the same reference variation graphs, on a similar simulated dataset, and the same real dataset as used in Chapter 4. I also used the same default parameters. The relative abundance estimation was based on the mean of the strain-specific gene abundances, computed by taking into account all the nodes (including non-covered nodes), and using a 50% threshold on the proportion of detected specific genes.

The presented results are compared to the results obtained with the first release of **StrainFLAIR**.

Validation on a simulated dataset

I first validated my method on simulated data, focusing on the single species *E. coli* with multiple strains. I simulated short sequencing paired-end reads of 150 bp and given a fragment length of 500 pb using `vg sim` from `vg toolkit` with a probability of sequencing errors set to 0.1%. Like in the previous experiments, the first simulation was a mixture composed of strains indexed in the reference graph (O104:H4, IAI39 and K-12) while the second simulation (O104:H4, IAI39 and BL21) had one absent from the reference variation graph (BL21) thus simulating a novel strain to be identified and quantified.

Here, the simulations tested were generated as mixtures of 300,000 reads (150,000 pairs) for O104:H4, 200,000 reads (100,000 pairs) for IAI39, and 100,000 reads (50,000 pairs) for K-12 or BL21.

Proportion of disambiguations

The first results I wanted to highlight were the proportion of reads involved in disambiguations that could be addressed.

For the mixture containing K-12:

- 600,000 reads were processed (300,000 pairs). Among them, 92,602 reads did not meet the mapping score threshold. Also, 980 reads were dropped for being discordant. As a reminder, reads are defined as discordant when the two reads of the same pair has no common strains attributed. Moreover, 2,190 reads were unassigned. As a reminder, it corresponds to reads for which no matching colored-path was found. Finally, 556 reads could not be used because of multiple possible alignments that could not be resolved.
- In total, 503,672 reads (83.9%) were used. The percentages presented below are based on this total.
- For 70,901 reads (14.1%), their colored-path attribution ambiguities were resolved. Here, “resolved” means that, initially, the reads matched more than one colored-path and my method could reduce this list to a single colored-path.
- For 16,571 reads (3.3%), their strain attribution ambiguities were resolved. Again, “resolved” means that the lists of matching strains was reduced to one strain.
- For 10,265 reads (2%), their strain attribution ambiguities were reduced but not resolved.

For the mixture containing BL21:

- 600,000 reads were processed (300,000 pairs). Among them, 99,789 reads did not meet the mapping score threshold. Also, 5,456 reads were dropped for being discordant. Moreover, 10,738 reads were unassigned. Finally, 501 reads could not be used because of multiple possible alignments that could not be resolved.
- In total, 483,516 reads (80.6%) were used. The percentages presented below are based on this total.
- For 66,213 reads (13.4%), their colored-path attribution ambiguities were resolved.
- For 15,193 reads (3.1%), their strain attribution ambiguities were resolved.
- For 10,261 reads (2.1%), their strain attribution ambiguities were reduced but not resolved.

Except for the number of reads defined as unassigned that was higher in the case of the mixture with BL21 (which was expected since it is the mixture containing an unknown strain, thus more reads would not be able to map onto the indexed reference sequences), both experiments showed similar results in terms of proportion of disambiguations. This

likely suggests that the existence of those ambiguities and their amount are not dependent on whether or not unknown strains are present in the queried sample, but rather depends on the reference strains indexed and their sequence similarity.

Interestingly, a non-negligible amount of reads (more than 10%) had their ambiguities completely resolved, avoiding sharing counts of reads. To a lesser extent (less than 5%), some ambiguities could also be reduced. The next sections present the impact of these disambiguations on **StrainFLAIR**'s primary outputs.

Proportion of detected genes

I tested the impact of my updated methodology on the proportion of specific genes detected for each indexed genomes. Those proportions were computed exactly the same way as described in Chapter 4. Similarly, close to 100% of the genes detected for the strains present in the sample were expected and close to 0% for the referenced strains absent in the sample, if the coverage was sufficient (which was the case considering the results obtained using the first release of **StrainFLAIR** with the selected samples). The results are presented in Table 5.1.

Mixture	Method	O104:H4	IAI39	K-12	Sakai	SE15	Santai	RM8426
K-12	StrainFLAIR	0.994	0.989	0.979	0.202	0.152	0.207	0.229
	StrainFLAIR-PE	0.993	0.982	0.982	0.045	0.023	0.046	0.06
BL21	StrainFLAIR	0.993	0.988	0.687	0.3	0.196	0.324	0.338
	StrainFLAIR-PE	0.994	0.983	0.62	0.15	0.058	0.18	0.188

Table 5.1 – **Reference strains proportion of specific genes detected computed by the first release of **StrainFLAIR** and the new version (**StrainFLAIR-PE**) for each simulated experiment.** Best results are shown in bold. Is considered best if the proportion is closer to the expected value, that is to say higher for the strains present in the mixture and lower for the strains absent.

For the present strains in the mixture (O104:H4, IAI39, and K-12 for the mixture containing K-12), the proportion of specific genes detected were already very close to 100%, thus my new methodology had little impact on those results. However, for the referenced strains absent in the mixture, the benefits of using the paired information were clearly visible. Notably for the mixture with K-12 (all strains in the mixture are referenced in the variation graph), almost all the expected absent strains had less than 5% of their genes detected, whereas this proportion was around 20% with the first release of **StrainFLAIR**. This gain in accuracy was also visible for the mixture with BL21. The expected absent strains (except K-12) had between 6% and 19% of genes detected, whereas

it was between 19% and 34% with the first release of **StrainFLAIR**.

Strain-level abundances

I tested the impact of my updated methodology on the estimated relative abundances. Those abundances were computed exactly the same way as described in Chapter 4. The results are presented in Table 5.2.

Mixture	Method	O104:H4	IAI39	K-12	Sakai	SE15	Santai	RM8426
K-12	Expected	50	33.33	16.67	0	0	0	0
	StrainFLAIR	44.67	35.04	20.29	0	0	0	0
	StrainFLAIR-PE	47.96	32.21	19.83	0	0	0	0
BL21	Expected	50	33.33	16.67*	0	0	0	0
	StrainFLAIR	50.5	38.63	10.87	0	0	0	0
	StrainFLAIR-PE	54.72	35.63	9.64	0	0	0	0

Table 5.2 – **Reference strains relative abundances expected and computed by **StrainFLAIR** or the new methodology based on paired-end reads (**StrainFLAIR-PE**) for each simulated experiment.** BL21’s expected abundance is followed by an asterisk in the K-12 column. Best results are shown in bold. Is considered best if the abundance is closer to the expected value.

For the mixture with only known strains, all estimations were closer to the expected relative abundances compared to the first release of **StrainFLAIR**, demonstrating the gain in accuracy by using paired-end reads and my disambiguation approach. The sum of squared differences compared to the expected values were of 44.4 for the first release of **StrainFLAIR** and decreased to 15.4 with my new methodology.

For the mixture with one unknown strain, only the estimation for IAI39 relative abundance improved, while the others were less close than the expected values compared to the first release of **StrainFLAIR**. The sum of squared differences compared to the expected values were of 62 for the first release of **StrainFLAIR** and increased to 77 with my new methodology. Hence, the loss in accuracy was slightly lower in this experiment than the gain in accuracy observed with the mixture containing only known strains.

Validation on a real dataset

I used the same mock dataset as in Chapter 4 (available on EBI-ENA repository under accession number PRJEB42498), in order to evaluate the impact of taking into account the reads as pairs in a more complex mixture.

Proportion of disambiguations

Again, I first wanted to highlight the proportion of reads involved in disambiguations that could be addressed:

- 42,778,364 reads were processed. This corresponded to 21,389,168 pairs, however `vg` could not map all pairs and 28 reads were mapped as single-end reads. Among all the reads processed, 9,952,234 reads did not reach the mapping score threshold. Also, 29,198 reads were dropped for being disconcordant. Moreover, 22,050 reads were unassigned. Finally, 42,789 reads could not be used because of multiple possible alignments that could not be resolved.
- In total, 32,732,093 reads (76.5%) were used. The percentages presented below are based on this total.
- For 311,392 reads (0.95%), their colored-path attribution ambiguities were resolved.
- For 40,942 reads (0.13%), their strain attribution ambiguities were resolved.
- For 2,514 reads (0.008%), their strain attribution ambiguities were reduced but not resolved.

Here, while my approach still allowed to resolve some ambiguities, it corresponded to a negligible amount of reads (less than 1%). Whereas the mock dataset is a more complex mixture with several species and some strains of the same species mixed together, it probably lacks similarity complexity such as this approach could actually benefit this sample.

Proportion of detected genes

I tested the impact of my updated methodology on the proportion of specific genes detected. Those proportions were computed exactly the same way as described in Chapter 4.

As expected by the analysis of the proportion of disambiguation that showed a limited impact on the number of reads concerned, this new approach did not alter the results on the proportion of detected genes. For this reason, the detailed results are not showed, the sum of squared errors between the proportions computed in Chapter 4 and the proportions obtained from this updated method was equal to 0.17.

Nevertheless, interestingly, one genome had a notable change. *Thermotoga* sp. RQ2 had 42.3% of specific genes detected with the first release of **StrainFLAIR**. As a reminder, *Thermotoga* sp. RQ2 genome was used to construct the reference variation graph but was

absent from the queried sample. Moreover, *Thermotoga* sp. RQ2 was the strain expected to be absent that **Kraken2** estimated with the highest relative abundance among the other expected absent strains, and the only one exceeding the relative abundances of two present strains. Since 42.3% is close to the threshold of 50% of detected genes to consider the strain present, and considering the results from **Kraken2**, this could question the possibility of a contamination of the sample with *Thermotoga* sp. RQ2. However, with my paired-end reads approach, the proportion of specific genes detected decreased to 26%, strengthening the fact that *Thermotoga* sp. RQ2 is absent from the sample as expected from the theoretical values.

Strain-level abundances

I tested the impact of my updated methodology on the estimated relative abundances. Those abundances were computed exactly the same way as described in Chapter 4.

As expected by the analysis of the proportion of disambiguation that showed a limited impact on the number of reads concerned, this new approach did not alter the results on the estimated abundances. For this reason, the detailed results are not showed, the sum of squared errors between the abundances computed in Chapter 4 and the abundances obtained from this updated method was equal to 0.004. Hence, the difference was so negligible that it would not be visible in figures using the same representations as in Chapter 4.

5.1.4 Conclusion

As seen by comparing the results between the simulated datasets and the mock dataset, using paired-end information is particularly relevant for complex mixtures of highly similar genomes. My approach had almost no impact on the mock dataset, probably because it was less complex in terms of strains mixture than the simulated datasets. However, it highlighted that it did not negatively alter the estimated abundances either and strengthened the results for absent strains as seen with *Thermotoga* sp. RQ2.

While the results were improved with the simulated dataset containing only known strains, a slight decrease was observed with an unknown strain. This is probably due to the increase of accuracy and the use of relative abundance. Indeed, because paired-end reads consideration permits less noise in the data, fewer reads from the strain BL21 would be attributed to the closest strain K-12, leading to a decrease of its estimated abundance and

therefore, because they are relative abundance, an increase of the other strain abundances.

Eventually, this paired-end approach is now the default parameter for **StrainFLAIR**, however the user can still choose to force a single-end approach, as it may be interesting to compare both depending on the study.

5.2 Inference of new strains

5.2.1 Rational

One of the main advantage of the graph structure, is that reads can map to non-colored-paths, revealing new sequences absent from the references without being discarded due to a low mapping score. In this work, those reads have been called *unassigned* and were not used in the first release of **StrainFLAIR**.

Although they were considered unassigned, those reads still matched paths in the graph and more specifically paths in particular connected components of the variation graph, that represent clusters of genes. By analyzing the reads at the cluster level, I assumed that it would be possible to determine the presence of one or more new strains depending on the compatibility or incompatibility (defined below) of the reads present in each cluster. Moreover, those reads might also be used to infer the abundance of those new strains.

Here, the terms *compatibility* and *incompatibility* describe how two reads are related. Two reads are said to be compatible if the paths they describe (the successive nodes they traverse in the graph) are overlapping with identical nodes in the overlapped part. On the contrary, two reads are said to be incompatible if their lists of nodes are overlapping on at least one node but with at least one distinct node (see Figure 5.2). In practice, among the common nodes between the two reads, one is used as a seed. For each position of this seed in the reads list of nodes, the overlapped part corresponds to the seed plus its left and right successive nodes until the end of one of the two reads. Of note, a node can be traversed twice or more by a read and thus appear several times in the list of nodes. The reads are compatible if among those different positions, at least one shows identical overlapped part (identical list of nodes). If the two reads have no node in common (no overlap), there is no conclusion on whether they are compatible or incompatible. Eventually, two compatible reads might originate from the same gene or strain, whereas two incompatible reads cannot originate from the same gene.

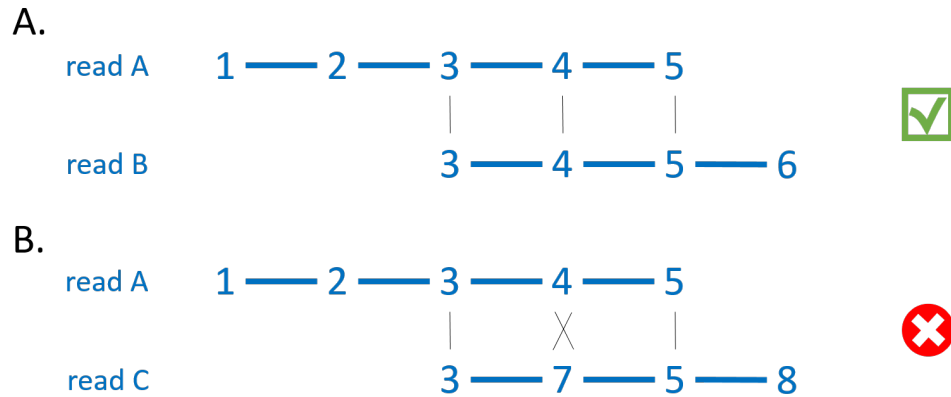


Figure 5.2 – **Compatibility and incompatibility between reads.** **A.** The read A traverses the nodes 1-2-3-4-5 in the variation graph, while the read B traverses 3-4-5-6. Thus those two reads overlap without any distinct node (3-4-5) and are considered compatible. **B.** The read A traverses the nodes 1-2-3-4-5 in the variation graph, while the read C traverses 3-7-5-8. Thus those two reads overlap on the nodes 3 and 5, however the node 4 from read A does not match the node 7 from read C, hence they are considered incompatible.

The algorithm uses a graph to represent the incompatibilities between all unassigned reads associated to a cluster of genes. The length of the maximal clique (subset of nodes such that every two distinct nodes in the clique are connected) then provides the minimum number of new genes and therefore potential new strains. For example, if two reads are incompatible (leading to a clique of length 2 in the graph of incompatibilities), as already mentioned they cannot originate from the same gene, indicating the existence of at least two genes.

The following sections present the detailed algorithm and the results on simulated datasets.

5.2.2 Algorithm

The algorithm detailed below characterizes the compatibility or incompatibility of each pair of reads within a connected component (cluster of genes). Moreover, due to the sequencing of the whole-genome, reads that originate from intergenic regions might map with a score good enough to pass the user-defined threshold and be considered as an unassigned read.

This would lead to incorrect incompatibility conclusions and an overestimation of the number of new strains. This might also happen with reads that originate from genes but

because of sequencing errors, they map on non-colored-paths. Therefore, the algorithm also includes a crucial filter step to remove the reads that likely introduce noise.

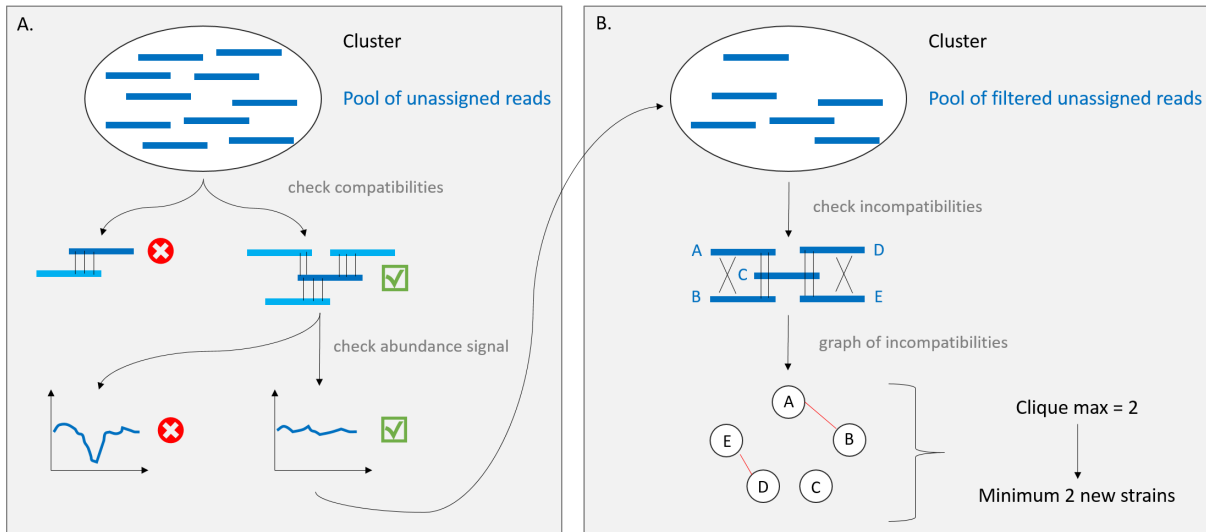


Figure 5.3 – Inference of new strains pipeline. **A.** The unassigned reads of each cluster are filtered. First by the number of their compatible reads, and then according to the homogeneity of the abundance signal computed from the overlapping compatible reads. **B.** The remaining reads after the filter step are represented into a graph of incompatibilities. In the illustration, the read C is compatible with all the other reads, however, reads A and B are incompatible, as well as reads D and E. The graph of incompatibilities then reveals two cliques of length 2, implying at least two new strains in the sample. Indeed, there is no combination of those reads allowing for only one solution. Although, the result provided is a minimum, as more than 2 combinations could be realized with this configuration (ACD, ACE, BCD, BCE).

As previously mentioned, the algorithm operates on each cluster of genes (see Figure 5.3), after the reads attribution step of the pipeline. Each cluster is associated with a pool of reads that could not match a colored-path but matched a non-colored-path of the cluster.

Firstly, for each read, the list of its compatible reads among the other unassigned ones is determined. In order to exclude isolated reads that might originate from intergenic regions or due to sequencing errors, a first filter is applied on the number of compatible reads. For this first filter only, other reads identical or representing a subset of the traversed nodes of the current read are not considered compatible. Hence, the read is removed if it has not at least two compatible reads. Then, a second filter is applied inspired by the signal processing field. The same way the first release of **StrainFLAIR** applies an abundance to each node of a path when a read maps on it, here, each node of a read is given a abundance which is the count of the compatible reads traversing it. The abundances along the nodes

traversed by the read can thus be viewed as a signal (as schematized in Figure 5.3A). In order to remove noise, reads that do not display an homogeneous signal are discarded. For this, I used the signal processing function `find_peaks` from the Scipy module to detect drops in the read abundance signal.

Secondly, the remaining reads that have passed both filters are used to build a graph representation of their incompatibilities. Here, the nodes are the reads and the edges correspond to the existence of an incompatibility between two reads.

Finally, all cliques from the incompatibility graph are determined, and the length of the maximal clique determines the minimum number of new strains represented in the cluster. Actually, it determines the minimum number of new genes and it cannot be excluded that it may correspond to two genes (likely paralogs) from the same new strain that are similar enough to map to the same genes family. However, I assumed that it is a minor event compared to the number of clusters analyzed, and for simplification, the following sections will mention new strains instead of new genes. This will also be further discussed in the Conclusion of the thesis.

The final estimation of novel strains for the queried sample is computed as the maximum among all estimated minimum numbers of new strains inferred for each cluster.

5.2.3 Validation

I validated my method with the same reference variation graph as used in Chapter 4 for the simulated experiments, and used some identical and new simulated datasets, focusing on the single species *E. coli* with multiple strains. I also used the same default parameters for **StrainFLAIR**. I simulated paired-end reads and used my new algorithm as described in Section 5.1. For all the presented mixtures, I simulated short sequencing paired-end reads of 150 bp and given a fragment length of 500 pb using `vg sim` from `vg toolkit` with a probability of sequencing errors set to 0.1%.

No new strains

I simulated a mixture with only referenced strains, identical to one of the mixtures used in Chapter 4: 300,000 reads (150,000 pairs) for O104:H4, 200,000 reads (100,000 pairs) for IAI39, and 100,000 reads (50,000 pairs) for K-12. Thus, I expected that no new strain would be inferred.

As presented in Table 5.3, I first highlighted the relevance of using the filters described in Section 5.2.2. Without any filters, my algorithm inferred up to minimum 4 new strains while none was expected. The filters had drastically decreased the number of clusters inferring more than zero strains.

	No new strain	1 new strain	2 new strains	3 new strains	4 new strains
No filters	7,470	1,059	62	4	1
With filters	8,388	203	5	0	0

Table 5.3 – Number of clusters that infers a certain minimum number of new strains, for a mixture with no unknown strain.

There were still 208 clusters remaining that inferred more than zero strains. However, two points should be mentioned to mitigate those results. Firstly, those 208 clusters represented only 2% of the total number of clusters analyzed (8,596) and could be considered as negligible, orienting towards the development of another filter at the clusters level. Secondly, most of those unexpected inferences were realized with a small number of reads. Only 10 clusters among the 208 had more than 10 reads mapped on them, while the maximum number of reads mapped among all clusters was 27 (see Figure 5.4). This could also orient to the development of another filter on the number of reads mapped to a cluster.

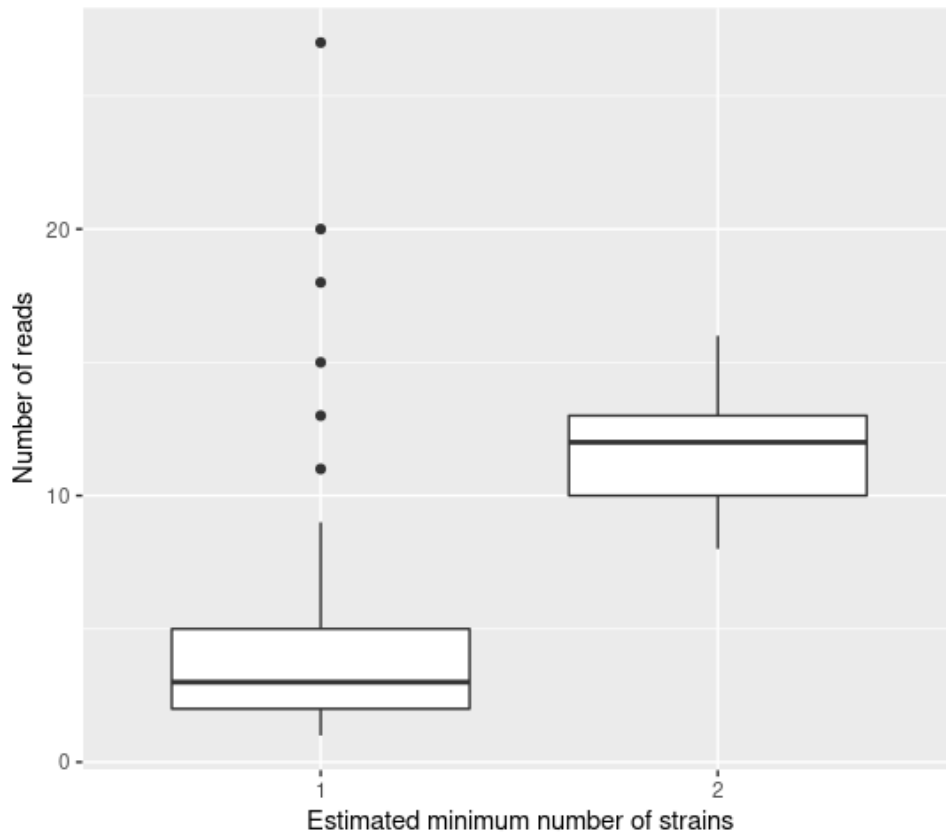


Figure 5.4 – Boxplot of the number of reads mapped to each cluster according to the minimum number of new strains inferred for a mixture with no unknown strains.

One new strain

I simulated a mixture of two referenced strains and one unknown strain, identical to one of the mixtures used in Chapter 4: 300,000 reads (150,000 pairs) for O104:H4, 200,000 reads (100,000 pairs) for IAI39, and 100,000 reads (50,000 pairs) for BL21. Thus, I expected that only one new strain would be inferred.

As presented in Table 5.4, I again highlighted the relevance of using filters. Without any filters, my algorithm inferred up to minimum 4 new strains while only one was expected. The filters had drastically decreased the number of clusters inferring more than one strain.

	No new strain	1 new strain	2 new strains	3 new strains	4 new strains
No filters	6,412	1,931	243	9	1
With filters	7,184	1,374	38	0	0

Table 5.4 – Number of clusters that infers a certain minimum number of new strains, for a mixture with one unknown strain.

There were still 38 clusters remaining that inferred more than one strain. Again, 38 clusters represented a negligible proportion of the total number of clusters analyzed (0.4%). It also represented a negligible amount of the total number of clusters inferring more than zero strains (2.7%). Moreover, while less significant than in the previous experiment, those 38 clusters had between 4 and 26 reads mapped on them, with 18 having more than 10 reads (max 46 reads, see Figure 5.5).

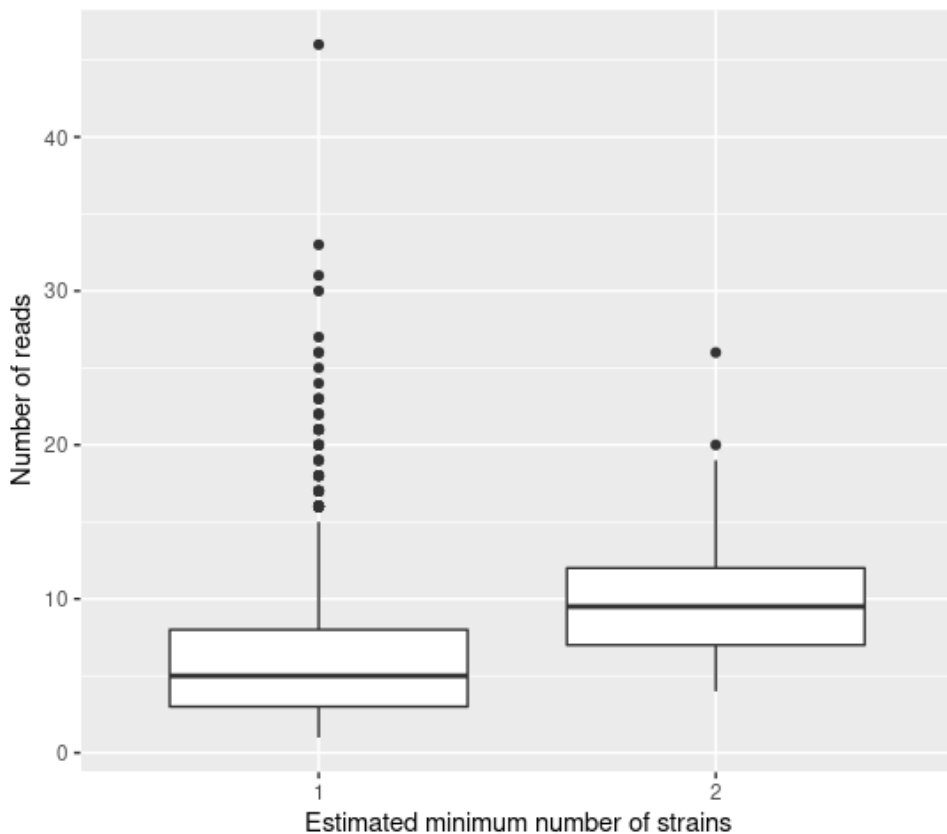


Figure 5.5 – Boxplot of the number of reads mapped to each cluster according to the minimum number of new strains inferred for a mixture with one unknown strain

Two new strains

I simulated a mixture of two referenced strains and two unknown strains: 300,000 reads (150,000 pairs) for O104:H4, 200,000 reads (100,000 pairs) for IAI39, 100,000 reads (50,000 pairs) for BL21, and 100,000 reads (50,000 pairs) for UTI89. Thus, I expected that two new strains would be inferred.

The sequence from *Escherichia coli* UTI89 (NC_007946.1) was a new strain added in the mixture as an unknown strain additionally to BL21. In order to be able to infer that at least two new strains were present in the sample, I choose a strain that was not too similar to the referenced strains or to BL21, and not the most similar to K-12 like BL21. All similarity scores from `fastANI` with the other strains were between 96.4 and 98.5. Among the referenced strains, SE15 was the closest to UTI89 according to this similarity score.

As presented Table 5.5, I again highlighted the relevance of using filters. Without any filters, my algorithm inferred up to minimum 4 new strains while only two were expected. The filters had drastically decreased the number of clusters inferring more than two strains.

	No new strain	1 new strain	2 new strains	3 new strains	4 new strains
No filters	5,131	2,307	1,057	96	5
With filters	5,749	2,428	415	4	0

Table 5.5 – **Number of clusters that infers a certain minimum number of new strains, for a mixture with two unknown strains.**

There were still 4 clusters remaining that inferred more than two strains. Again, 4 clusters represented a negligible proportion of the total number of clusters analyzed (0.05%). It also represented a negligible amount of the total number of clusters inferring more than one strain (1%). Moreover, those 4 clusters had between 8 and 35 reads mapped to them, with 3 having more than 10 reads (max 70 reads, see Figure 5.6).

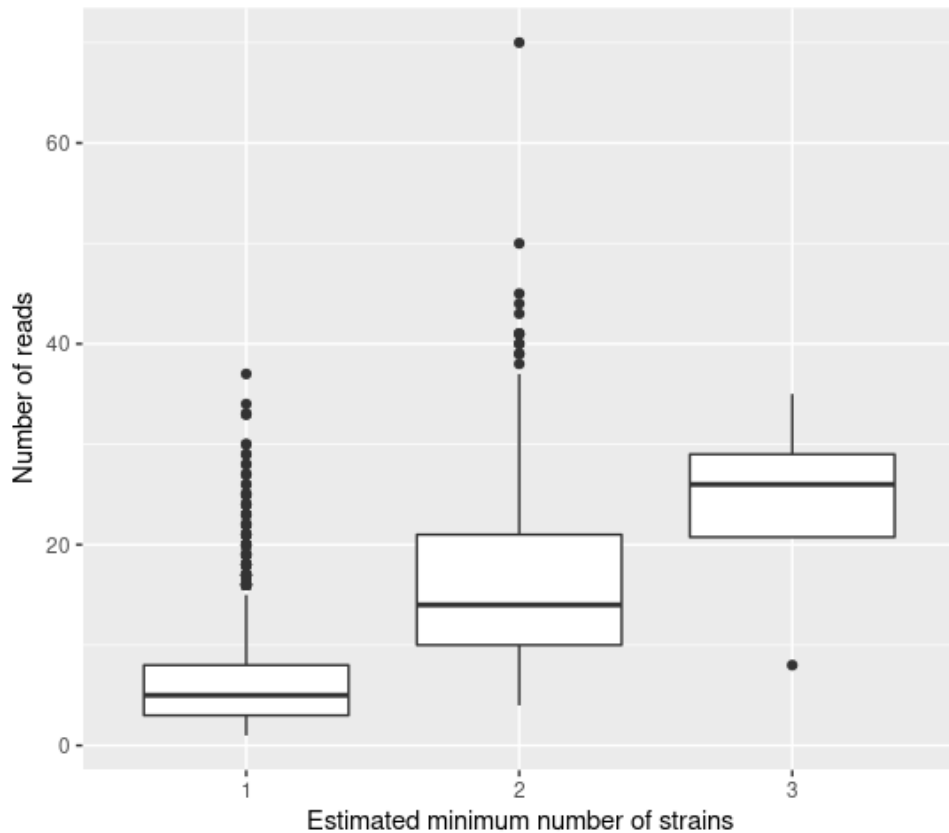


Figure 5.6 – Boxplot of the number of reads mapped to each cluster according to the minimum number of new strains inferred for a mixture with two unknown strains

Three new strains

I simulated a mixture of two referenced strains and three unknown strains: 300,000 reads (150,000 pairs) for O104:H4, 200,000 reads (100,000 pairs) for IAI39, 100,000 reads (50,000 pairs) for BL21, 100,000 reads (50,000 pairs) for UTI89, and 100,000 reads (50,000 pairs) for LF82. Thus, I expected that three new strains would be inferred.

The sequence from *Escherichia coli* LF82 (NC_011993.1) was a new strain added in the mixture as an unknown strain additionally to BL21 and UTI89. All similarity scores from *fastANI* with the other strains were between 96.65 and 98.9, assuring that the strain would still be distinguishable among the others. However, as opposed to the previous experiment, the new strain LF82 was the closest to another unknown strain (UTI89) instead of a referenced one.

As presented Table 5.6, I again highlighted the relevance of using filters. Without any filters, my algorithm inferred up to minimum 5 new strains while only three were expected. The filters have drastically decreased the number of clusters inferring more than three strains.

	No new strain	1	2	3	4	5 new strains
No filters	4,933	1,944	1,325	375	17	2
With filters	5,386	2,353	818	39	0	0

Table 5.6 – Number of clusters that infers a certain minimum number of new strains, for a mixture with three unknown strains.

This experiment fully gave the expected results. While 39 clusters inferring a minimum of three new strains represent a small proportion of the total number of clusters analyzed (0.5%), they had between 8 and 95 reads mapped on them, with only one having less than 10 reads (max 95 reads, see Figure 5.7). Therefore, despite their small proportion, the number of reads mapped strengthen the robustness of this result.

However, it is important to note that this favorable result might be due to LF82 being too close to UTI89, leading to the algorithm to not be able to distinguish them and cancelling the difference of 1 strain between the observed and expected results found in the previous experiments.

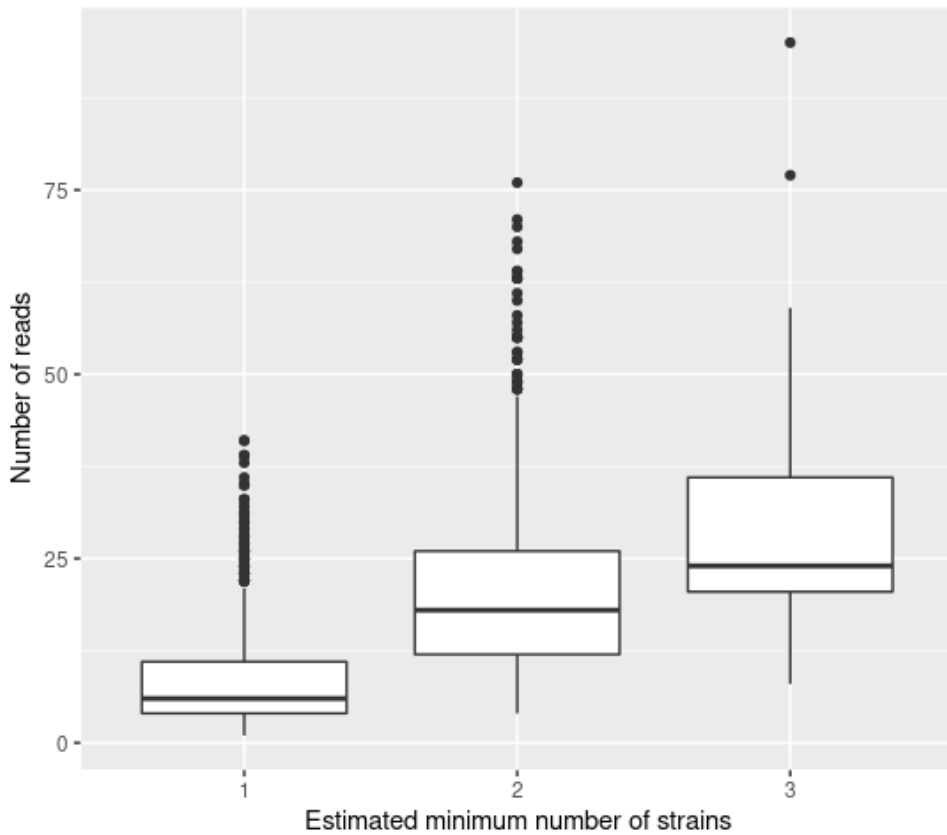


Figure 5.7 – Boxplot of the number of reads mapped to each cluster according to the minimum number of new strains inferred for a mixture with three unknown strains

5.2.4 Conclusion

The results on the various simulated datasets emphasize the relevance of this approach. However, it also showed the importance of a cleaning step to remove noisy reads. The current strategy has demonstrated important improvements on the results compared to the results without any filters, although not completely as expected from the simulations. Yet, the remaining incorrect results represent a small fraction of the whole set of results, suggesting that some minor additional filters should be required to correct them.

While the results are promising, this observation is mainly based on the decrease of the number of clusters overestimating the minimum number of new strains after applying the filters. However, there is also an important decrease of the number of clusters estimating

the expected minimum number of new strains, especially in the case of mixtures with two or three new strains. For the mixture with two new strains, 243 clusters estimated two new strains without any filters, while it decreased to 38 with filters. For the mixture with three new strains, 375 clusters estimated three new strains without any filters, while it decreased to 39 with filters, and 1,325 clusters estimated two new strains without any filters, while it decreased to 818 with filters. More exploration is needed to determine if the filters have induced the loss of clusters that correctly estimated the minimum number of new strains or, on the contrary, if they contributed to also correct the estimation from those clusters.

CONCLUSION

Challenges and objectives

Metagenomics studies the genomic composition of microorganisms present in a sample. Recent advances in sequencing technologies have provided large resources from diverse environments. The main advantage of metagenomics lies in allowing to sample all microorganisms, and more importantly those that cannot be cultured. Metagenomics analyses have notably been applied to gain insights on human health and diseases, mainly by revealing the species composition of samples and its association to phenotypes. And while those species-level analysis are well established, from the construction of metagenome-assembled genomes or from the construction and use of genes and metagenomic species catalogs, new methodological approaches are needed to characterize metagenomics samples at the strain level.

Microbial communities are usually highly diverse, representing multiple taxonomic levels. The next challenge of the metagenomics field is then to capture the variations at the strain level in order to even more accurately describe the composition of a sample. New associations with diseases or with efficiency/toxicity of drugs, for instance, may be highlighted by considering the strain composition of an individual's microbiota whereas they are currently masked by considering only the species level. Targeting specific bacterial strains will open the field of precision medicine (Albanese and Donati, 2017; Marchesi et al., 2016).

The main limitations with the current species-level approaches lie in the reduction of the redundancy, a unique sequence is considered as the representative for other similar sequences, getting rid of the variations that could characterize strains of the same species. Metagenomics allows to capture the whole diversity of a sample but can hardly process all this information, mostly because of the lack of a computationally efficient representation. Representation and integration of multiple genome are under active development and led to graph-based frameworks and softwares. Integrating highly similar genomes from the same species provides new opportunities to represent and analyse strains.

The existing strain-level tools are either specialized in haplotype inference or in inferring the number of strains, rely on multiple samples, or only identify strains from the references used. Eventually, no tool combines the needed requirements for a strain-level profiling of a metagenomic sample: able to operate on a single sample, identify known present and new strains, identify all of them and not only the dominant ones, and able to provide relative abundances of each strain. Moreover, while graph representations are becoming popular, there is still a need to develop downstream analysis using them.

Contributions

We developed **StrainFLAIR**, a tool for strain-level profiling of metagenomic samples that uses variation graphs. **StrainFLAIR** had two main objectives. Firstly, to test the feasibility of characterizing a metagenomic sample at the strain-level by indexing highly similar genomes in a variation graph. Secondly, to offer a tool allowing to perform the indexing of genomes and the query of a metagenomic sample by the analysis of sequencing reads mapped onto the graph. **StrainFLAIR** exploits state-of-the-art tools additionally to novel algorithmic solutions.

In controlled experiments, we have demonstrated that **StrainFLAIR** was able to identify and estimate the abundance of strains in metagenomic samples, even when the graph was built using references of strains absent from the sample, that could have generated false positive calls like the results obtained with **Kraken2**. From the validation on simulated datasets, we notably showed that we were also able to suggest the presence of a novel strain (absent from the references) close to one of the reference used to build the graph, as well as to estimate its relative abundance. Furthermore, we have demonstrated that we could use paired-end reads information in order to resolve or reduce ambiguities existing when attributing reads to colored-paths of the graph, that led to less noise in the data and resulting in better abundance estimations when the strains present in the sample are represented by a reference in the graph, and a slight decrease when novel strains exist in the sample.

Moreover, due to the use of a threshold on the proportion of specific genes detected, **StrainFLAIR** tended to set to zero the predicted abundance of low abundant strains (whereas a tool like **Kraken2** was able detect them). Therefore, it appears that currently **StrainFLAIR** is not able to detect very low abundant strains. However, in the presented

simulations, it was still relevant to maintain such strategy as this situation corresponded to coverages of 0.03x or less, representing a strain for which not all genomic content was present. Hence, it might be better suited to consider such a strain absent.

Detecting strains as absent in the queried sample is of great interest in metagenomics. No prior knowledge is usually available on the sample, leading to the use of the most exhaustive references, including unnecessary genomes. Strain-level profiling tools then require to avoid false positive calls that would biased the downstream analysis. **StrainFLAIR** operates in this direction.

Additionally to the abundance estimation, we started to explore the unassigned reads at the cluster level in order to add a novel strain inference feature to **StrainFLAIR**. While the first release of **StrainFLAIR** allowed to imply the presence of at least one new strain close to one the indexed reference in the graph, there was still a need to better characterize those new strains, at least in terms of number. We showed in simulated experiments that we estimate a number of strains very close to the actual number of unknown strains, and that the incorrect estimations were supported by a limited amount of gene clusters.

In conclusion, and considering the results presented in Chapter 4 and in Section 5.1, our approach that takes into account single and multiple mapped reads and that imposes a threshold allowing for some strains abundances to be set to zero seems more adequate and closer to what is expected (experimental data or ground truth) compared to other tools. Furthermore, our implementations presented in Chapter 4 and in Section 5.2 set promising paths to fully detect, distinguish and estimate the relative abundance of novel strains.

Perspectives

Firstly, I detail the perspectives towards direct and short-term developments for **StrainFLAIR**, considering the results and limitations presented in this thesis.

Although **StrainFLAIR** showed convincing results on simulated and real datasets, exploring more complex situations is still required. Notably, the mock dataset was a controlled sample, and while we also included reference genomes absent from the queried sample, we still had prior knowledge of the sample composition to build the reference set. Actually, this can be reproduced for real situations by pre-filtering a genome database,

with **Kraken2** for example. Nevertheless, overall, further work is needed to evaluate the scalability of our method with larger reference sets in terms of species and strains. This may include to construct a variation graph for a single species like in the simulated experiments but with variable proximity between the selected strains, to construct a graph with several strains for each species known to be present in the mock dataset instead of only using the genomes provided, or test the scalability by constructing a variation graph with thousands of references.

In this same direction, **StrainFLAIR** needs to be used in the context of a real biological application. For instance, Solé *et al.* used the genes catalog methodology as described in Section 2.2 and showed that the progression of the disease in patients with cirrhosis was associated with a decrease of gene and species richness. Various species were found to be associated with different specific symptoms of the disease or related complications (Solé *et al.*, 2021). And they also built a model to predict the mortality 3 months after hospitalization using the gut microbiota species as predictors. An analysis on strain level with **StrainFLAIR** might highlight new or more specific associations and/or improve predictive models.

StrainFLAIR integrates a threshold on the proportion of specific genes detected that is required to estimate abundances at zero and conclude to the absence of a strain. However, this threshold needs to be further explored to refine which strain abundances are set to zero as it is still challenging to distinguish between low abundant strains, insufficient sequencing depth, and reads from intergenic regions or other genes randomly matching genes.

Currently, **StrainFLAIR** uses only the strain-specific genes in order to compute relative abundances. New strategies need to be developed to take into account non strain-specific genes (core and shell genome), hence the potential ties between this thesis project and pangenomics. For each cluster, **StrainFLAIR** outputs the number of genes from each reference strain indexed in the variation graph. This information might be used to refine the abundances and the detection of new strains. Additionally, existing pangenomic tools like **PPanGGOLiN**, which conserves the genomic organization along genomes (synteny), could be used to add another layer on resolving ambiguities as seen in Section 5.1 by taking into account the distance between the genes if the paired-end reads map different genes.

StrainFLAIR provides two distinct insights for detection of novel strains. The first one, present in the first release of the tool, is the proportion of specific genes detected

for the references. Neither close to 100% like the expected present strains, nor under our 50% threshold like the absent strains (as seen in Section 4.2.1), this output indicates that there is at least one new strain close to the reference concerned, with an estimated relative abundance. Consequently, in case of more than one novel strain close to the same reference, this estimated abundance will correspond to the sum of the abundances of those strains. With only the first released version, distinguishing those strains is not yet possible. On the other hand, our new developments after the first release of **StrainFLAIR** have permitted to estimate the minimum number of novel strains in the sample (as seen in Section 5.2.3). Moreover, an estimated abundance from the reads used to infer this number of strains was computed and remains to be exploited. Eventually, these results need to be combined to output a more characterized profile of the novel strains. In addition, as a reminder, we simplified the fact that our approach infers the minimum number of strains whereas it would be more accurate to describe it as the inference of the minimum number of genes, since several genes from the same novel strain could map the same cluster. Similarly to the previous paragraph, the number of genes from each reference strain indexed and a pangenomic approach might help to statistically untangle these situations.

StrainFLAIR can analyse million reads in a few hours, which is sufficient for routine analyses of small read sets. Nevertheless, new development are needed to reduce the computation time in order to scale-up to very large datasets. Worth noting, after **StrainFLAIR**'s first release, the authors of the **vg toolkit** released a new mapper **Giraffe** (Sirén et al., 2020, not published yet). **Giraffe** is an haplotype-aware mapper that is ten times faster than their original mapper **vg map**. While **StrainFLAIR** needs the **vg mpmmap** mapper to operate, this shows the rapid development in the field and foreshadows new opportunities to scale-up.

Secondly, I detail more long-term perspectives, either for **StrainFLAIR** or for the field in general.

StrainFLAIR does not use a pangenomic approach. However, as mentioned above, this direction might help for the current limitations observed and, overall, variation graphs seem to be an adequate framework to explore the pangenome of a species. Genomic plasticity and diversity is of increasing importance in microbiology, hence the interest in pangenomics. Pangenomics is usually explored in two ways. First, from the gene presence/absence perspective, also allowing to characterize core and accessory genome of a species. Then, from fine analysis of genomic variations. From the use of variation graphs

to index clusters of genes, **StrainFLAIR** has the potential to cover both of those aspects. Graph structures that represent a set of similar sequences allow to capture all information on presence/absence of genes and variation/similarity of sequences, leading to new highlights on genome organization and regions of plasticity in a species. In other words, **StrainFLAIR** might have the potential to integrate pangenomics and metagenomics for microbial community profiling.

Despite the limitations raised in Section 2.2.3, the number of MAGs reconstructed and available is growing and gut databases, for example, are alimeted by MAGs. As a result, some reference genomes will be represented by MAGs, more especially for sub-dominant strains. It will then be crucial to explore the impact of using MAGs as references for **StrainFLAIR**.

Another aspect that as yet to be fully used in our approach is the variability provided by the sequencing reads from new genomes. Although we used the reads that did not match any colored-path which is a first step to consider new genomes, the mismatches found in the alignments have not been used. This variability need to be integrated into the graphs, which assumes a dynamic structure. Therefore, a natural continuation of the thesis project would be related to the dynamical update of the reference graph used with **StrainFLAIR** when novel species or strains are detected. Additionally, all along this work, we saw how sequencing reads with different behaviour towards mapping have been used, notably in order to use all information available from them: reads matching a single colored-path, reads matching multiple colored-paths, and reads not matching colored-paths but still mapping the graph. The next step, which is also related to the detection of novel strains, would be to also use the non-mapped reads. Reads from these so-detected novel species or strains may be assembled using third-party haplotype-aware assemblers and the assembled sequences of genes will have to be added to the reference variation graph, updating clusters and path colors.

This thesis work focused exclusively on the use of short sequencing reads, as short-read sequencing technologies are still the most used, particularly in metagenomics. However, as described in Section 1.1.2, long reads are becoming more popular with decreasing rates of sequencing errors. Therefore, a long-term perspective points towards a future shift to long-read sequencing technologies or at least to the need for profiling tools capable to operate with long reads. In the field of strain-level profiling using long reads, **MetaMaps** (A. T. Dilthey et al., 2019) or **ORI** (Siekanić et al., 2021) have been developed, a lesser range of tools compared to what exists for short-reads. Despite the existence

of long-reads sequence-to-graph alignment tools (like the previously cited **GraphAligner** and **PaSGAL**), to our knowledge, no tool for strain-level analysis using variation graphs and long reads has been developed.

To conclude, **StrainFLAIR** and the use of variation graphs overall have still many opportunities for new developments and new advances in pangenomics and metagenomics.

APPENDIX

Mentioned in Section 4.2.1, Table A1 provides exhaustive results on simulated datasets when all queried strains are indexed in the variation graph. Table A2 provides exhaustive results on simulated datasets when one of the queried strain (BL21-DE3) is not indexed and highly similar to strain K-12.

Mentioned in Section 4.2.2, Figure A1 shows full results obtained on the mock dataset.

#reads K-12	Method	O104:H4	IAI39	K-12	Sakai	SE15	Santai	RM8426
1,000	Expected	59.88	39.92	0.2	0	0	0	0
	StrainFLAIR	56.47 (0.995)	43.53 (0.989)	0 (0.309)	0 (0.189)	0 (0.151)	0 (0.188)	0 (0.212)
	Kraken2	38.91	60.72	0.22	0.04	0.07	0.03	0.02
5,000	Expected	59.41	39.6	0.99	0	0	0	0
	StrainFLAIR	54.93 (0.995)	42.46 (0.989)	2.6 (0.546)	0 (0.202)	0 (0.153)	0 (0.2)	0 (0.227)
	Kraken2	38.61	60.25	0.99	0.04	0.07	0.03	0.02
10,000	Expected	58.82	39.22	1.96	0	0	0	0
	StrainFLAIR	54.12 (0.994)	41.96 (0.989)	3.92 (0.709)	0 (0.21)	0 (0.155)	0 (0.211)	0 (0.234)
	Kraken2	38.26	59.69	1.9	0.04	0.07	0.03	0.02
25,000	Expected	57.14	38.1	4.76	0	0	0	0
	StrainFLAIR	52.14 (0.994)	40.58 (0.989)	7.27 (0.878)	0 (0.208)	0 (0.153)	0 (0.215)	0 (0.234)
	Kraken2	37.23	58.1	4.51	0.04	0.07	0.03	0.02
50,000	Expected	54.55	36.36	9.09	0	0	0	0
	StrainFLAIR	49.25 (0.994)	38.5 (0.989)	12.24 (0.949)	0 (0.203)	0 (0.15)	0 (0.208)	0 (0.23)
	Kraken2	35.63	55.6	8.62	0.04	0.07	0.03	0.02
100,000	Expected	50	33.33	16.67	0	0	0	0
	StrainFLAIR	44.67 (0.994)	35.04 (0.989)	20.29 (0.979)	0 (0.202)	0 (0.152)	0 (0.207)	0 (0.229)
	Kraken2	32.8	51.19	15.85	0.04	0.07	0.03	0.02
200,000	Expected	42.86	28.57	28.57	0	0	0	0
	StrainFLAIR	38.12 (0.993)	29.81 (0.988)	32.08 (0.99)	0 (0.211)	0 (0.159)	0 (0.219)	0 (0.237)
	Kraken2	28.31	44.18	27.35	0.04	0.08	0.03	0.02

Table A1 – Reference strains relative abundances expected and computed by StrainFLAIR or Kraken2 for each simulated experiment with variable coverage of the K-12 MG1655 strain. Best results are shown in bold. For StrainFLAIR, the proportion of specific genes detected is shown in parentheses.

#reads BL21-DE3	Method	O104:H4	IAI39	K-12	Sakai	SE15	Santai	RM8426
1,000	Expected	59.88	39.92	0.2*	0	0	0	0
	StrainFLAIR	56.48 (0.995)	43.52 (0.989)	0 (0.254)	0 (0.189)	0 (0.151)	0 (0.192)	0 (0.214)
	Kraken2	38.93	60.76	0.11	0.05	0.08	0.04	0.03
5,000	Expected	59.41	39.6	0.99*	0	0	0	0
	StrainFLAIR	56.46 (0.995)	43.54 (0.989)	0 (0.387)	0 (0.216)	0 (0.16)	0 (0.218)	0 (0.239)
	Kraken2	38.72	60.42	0.5	0.09	0.13	0.08	0.07
10,000	Expected	58.82	39.22	1.96*	0	0	0	0
	StrainFLAIR	56.46 (0.995)	43.54 (0.989)	0 (0.471)	0 (0.236)	0 (0.169)	0 (0.243)	0 (0.262)
	Kraken2	38.47	60.05	0.92	0.14	0.19	0.12	0.13
25,000	Expected	57.14	38.1	4.76*	0	0	0	0
	StrainFLAIR	54.12 (0.995)	41.72 (0.989)	4.16 (0.584)	0 (0.266)	0 (0.177)	0 (0.282)	0 (0.298)
	Kraken2	37.75	58.93	2.16	0.28	0.34	0.25	0.29
50,000	Expected	54.55	36.36	9.09*	0	0	0	0
	StrainFLAIR	52.77 (0.994)	40.62 (0.989)	6.61 (0.652)	0 (0.284)	0 (0.187)	0 (0.307)	0 (0.321)
	Kraken2	36.59	57.17	4.15	0.51	0.57	0.48	0.53
100,000	Expected	50	33.33	16.67*	0	0	0	0
	StrainFLAIR	50.5 (0.993)	38.63 (0.988)	10.87 (0.687)	0 (0.3)	0 (0.196)	0 (0.324)	0 (0.338)
	Kraken2	34.53	54.03	7.68	0.91	0.98	0.91	0.96
200,000	Expected	42.86	28.57	28.57*	0	0	0	0
	StrainFLAIR	46.96 (0.993)	35.32 (0.988)	17.72 (0.711)	0 (0.318)	0 (0.211)	0 (0.346)	0 (0.351)
	Kraken2	31.14	48.83	13.53	1.57	1.67	1.58	1.68

Table A2 – Reference strains relative abundances expected and computed by StrainFLAIR or Kraken2 for each simulated experiment with variable coverage of the BL21-DE3 strain, absent from the reference graph. BL21-DE3 being similar at 98.9% to K-12 strain (highest similarity compared to the other references), we expect that reads from BL21-DE3 will map this strain, hence its expected values are followed by an asterisk, as they correspond to BL21-DE3 strain abundances and not K-12. Best results are shown in bold. For StrainFLAIR, the proportion of specific genes detected is shown in parentheses.

BIBLIOGRAPHY

- Akopyants, Natalia S. et al. (1998), « Analyses of the *cag* pathogenicity island of *Helicobacter pylori* », *in: Molecular microbiology* 28.1, pp. 37–53, ISSN: 0950-382X, DOI: 10.1046/J.1365-2958.1998.00770.X.
- Albanese, Davide and Claudio Donati (Dec. 2017), « Strain profiling and epidemiology of bacterial species from metagenomic sequencing », *in: Nature Communications* 8.1, ISSN: 20411723, DOI: 10.1038/s41467-017-02209-5.
- Alneberg, Johannes et al. (Oct. 2014), « Binning metagenomic contigs by coverage and composition », *in: Nature methods* 11.11, pp. 1144–1146, ISSN: 1548-7105, DOI: 10.1038/NMETH.3103.
- Altschul, Stephen F., Warren Gish, Webb Miller, Eugene W. Myers, and David J. Lipman (Oct. 1990), « Basic local alignment search tool », *in: Journal of Molecular Biology* 215.3, pp. 403–410, ISSN: 0022-2836, DOI: 10.1016/S0022-2836(05)80360-2.
- Armstrong, Joel et al. (Nov. 2020), « Progressive Cactus is a multiple-genome aligner for the thousand-genome era », *in: Nature* 2020 587:7833 587.7833, pp. 246–251, ISSN: 1476-4687, DOI: 10.1038/s41586-020-2871-y.
- Baaijens, Jasmijn A., Bastiaan Van Der Roest, Johannes Köster, Leen Stougie, and Alexander Schönhuth (Dec. 2019), « Full-length de novo viral quasispecies assembly through variation graph construction », *in: Bioinformatics* 35.24, pp. 5086–5094, ISSN: 1367-4803, DOI: 10.1093/BIOINFORMATICS/BTZ443.
- Ballouz, Sara, Alexander Dobin, and Jesse Gillis (Jan. 2019), « Is it time to change the reference genome? », *in: bioRxiv*, p. 533166, DOI: 10.1101/533166.
- Beller, Timo and Enno Ohlebusch (July 2016), « A representation of a compressed de Bruijn graph for pan-genome analysis that enables search », *in: Algorithms for Molecular Biology* 11.1, pp. 1–17, ISSN: 17487188, DOI: 10.1186/S13015-016-0083-7/TABLES/7, arXiv: 1602.03333.
- Ben-David, Avishai and Charles E. Davidson (Dec. 2014), « Estimation method for serial dilution experiments », *in: Journal of microbiological methods* 107, pp. 214–221, ISSN: 1872-8359, DOI: 10.1016/J.MIMET.2014.08.023.

-
- Bentley, David R. et al. (Nov. 2008), « Accurate whole human genome sequencing using reversible terminator chemistry », *in: Nature 2008 456:7218* 456.7218, pp. 53–59, ISSN: 1476-4687, DOI: 10.1038/nature07517.
- Borderes, Marianne, Cyrielle Gasc, Emmanuel Prestat, Mariana Ferrarini, Susana Vinga, Lilia Boucinha, Marie-France Sagot, Mariana Galvão Galv~, and Galvão Ferrarini (2021), « A comprehensive evaluation of binning methods to recover human gut microbial species from a non-redundant reference gene catalog », *in: NAR Genomics and Bioinformatics* 2021.1, p. 1, DOI: 10.1093/nargab/lqab009.
- Breitwieser, F. P., D. N. Baker, and S. L. Salzberg (Dec. 2018), « KrakenUniq: confident and fast metagenomics classification using unique k-mer counts », *in: Genome Biology* 19.1, p. 198, ISSN: 1474-760X, DOI: 10.1186/s13059-018-1568-0.
- Brenner, Don J., James T. Staley, and Noel R. Krieg (Sept. 2015), « Classification of Prokaryotic Organisms and the Concept of Bacterial Speciation », *in: Bergey's Manual of Systematics of Archaea and Bacteria*, pp. 1–9, DOI: 10.1002/9781118960608.BM00006.
- Břinda, Karel, MacIej Sykulski, and Gregory Kucherov (Nov. 2015), « Spaced seeds improve k-mer-based metagenomic classification », *in: Bioinformatics* 31.22, pp. 3584–3592, ISSN: 1367-4803, DOI: 10.1093/BIOINFORMATICS/BTV419, arXiv: 1502.06256.
- Brockhurst, Michael A., Ellie Harrison, James P.J. Hall, Thomas Richards, Alan McNally, and Craig MacLean (Oct. 2019), « The Ecology and Evolution of Pangenomes », *in: Current biology : CB* 29.20, R1094–R1103, ISSN: 1879-0445, DOI: 10.1016/J.CUB.2019.08.012.
- Carding, Simon, Kristin Verbeke, Daniel T. Vipond, Bernard M. Corfe, and Lauren J. Owen (Feb. 2015), « Dysbiosis of the gut microbiota in disease », *in: Microbial Ecology in Health and Disease* 26.0, ISSN: 0891-060X, DOI: 10.3402/MEHD.V26.26191.
- Chen, Sai et al. (Dec. 2019), « Paragraph: A graph-based structural variant genotyper for short-read sequence data », *in: Genome Biology* 20.1, pp. 1–13, ISSN: 1474760X, DOI: 10.1186/S13059-019-1909-7/FIGURES/5.
- Chikhi, Rayan, Antoine Limasset, and Paul Medvedev (June 2016), « Compacting de Bruijn graphs from sequencing data quickly and in low memory », *in: Bioinformatics* 32.12, pp. i201–i208, ISSN: 1367-4803, DOI: 10.1093/BIOINFORMATICS/BTW279.
- Clemente, Jose C., Luke K. Ursell, Laura Wegener Parfrey, and Rob Knight (Mar. 2012), *The impact of the gut microbiota on human health: An integrative view*, DOI: 10.1016/j.cell.2012.01.035.

-
- Coelho, Luis Pedro et al. (Dec. 2021), « Towards the biogeography of prokaryotic genes », *in: Nature 2021 601:7892* 601.7892, pp. 252–256, ISSN: 1476-4687, DOI: 10.1038/s41586-021-04233-4.
- Coico, Richard (Feb. 2006), « Gram Staining », *in: Current Protocols in Microbiology* 00.1, A.3C.1–A.3C.2, ISSN: 1934-8533, DOI: 10.1002/9780471729259.MCA03CS00.
- Colquhoun, Rachel M. et al. (Sept. 2021), « Pandora: nucleotide-resolution bacterial pan-genomics with reference graphs. », *in: Genome Biology* 22.1, pp. 267–267, ISSN: 1474-7596, DOI: 10.1186/S13059-021-02473-1.
- Comin, Matteo, Barbara Di Camillo, Cinzia Pizzi, and Fabio Vandin (Jan. 2021), « Comparison of microbiome samples: methods and computational challenges », *in: Briefings in Bioinformatics* 22.1, pp. 88–95, ISSN: 14774054, DOI: 10.1093/BIB/BBAA121.
- Costea, Paul Igor, Robin Munch, Luis Pedro Coelho, Lucas Paoli, Shinichi Sunagawa, and Peer Bork (2017), « metaSNV: A tool for metagenomic strain level analysis », *in: PLoS ONE*, ISSN: 19326203, DOI: 10.1371/journal.pone.0182392.
- Da Silva, Kévin, Nicolas Pons, Magali Berland, Florian Plaza Oñate, Mathieu Almeida, and Pierre Peterlongo (Aug. 2021), « StrainFLAIR: strain-level profiling of metagenomic samples using variation graphs », *in: PeerJ* 9, e11884, ISSN: 21678359, DOI: 10.7717/PEERJ.11884/SUPP-1.
- Darling, Aaron E., Guillaume Jospin, Eric Lowe, Frederick A. Matsen, Holly M. Bik, and Jonathan A. Eisen (2014), « PhyloSift: phylogenetic analysis of genomes and metagenomes », *in: PeerJ* 2.1, ISSN: 2167-8359, DOI: 10.7717/PEERJ.243.
- Derosa, Lisa et al. (2020), « Gut Bacteria Composition Drives Primary Resistance to Cancer Immunotherapy in Renal Cell Carcinoma Patients », *in: European Urology* 78, pp. 195–206, DOI: 10.1016/j.eururo.2020.04.044.
- Dijk, Lucas R. van et al. (May 2021), « StrainGE: A toolkit to track and characterize low-abundance strains in complex microbial communities », *in: bioRxiv*, p. 2021.02.14.431013, DOI: 10.1101/2021.02.14.431013.
- Dilthey, Alexander T., Chirag Jain, Sergey Koren, and Adam M. Phillippy (July 2019), « Strain-level metagenomic assignment and compositional estimation for long reads with MetaMaps », *in: Nature Communications* 2019 10:1 10.1, pp. 1–12, ISSN: 2041-1723, DOI: 10.1038/s41467-019-10934-2.
- Dobrindt, Ulrich (Oct. 2005), *(Patho-)Genomics of Escherichia coli*, DOI: 10.1016/j.ijmm.2005.07.009.

-
- Dohm, Juliane C, Philipp Peters, Nancy Stralis-Pavese, and Heinz Himmelbauer (June 2020), « Benchmarking of long-read correction methods », *in: NAR Genomics and Bioinformatics 2.2*, DOI: 10.1093/NARGAB/LQAA037.
- Eggertsson, Hannes P. et al. (Dec. 2019), « GraphTyper2 enables population-scale genotyping of structural variation using pangenome graphs », *in: Nature Communications 10.1*, pp. 1–8, ISSN: 20411723, DOI: 10.1038/s41467-019-13341-9.
- Ehrlich, S. Dusko (2011), « MetaHIT: The European Union project on metagenomics of the human intestinal tract », *in: Metagenomics of the Human Body*, Springer New York, pp. 307–316, ISBN: 9781441970893, DOI: 10.1007/978-1-4419-7089-3_15.
- Eisen, Jonathan A. (Mar. 2007), « Environmental Shotgun Sequencing: Its Potential and Challenges for Studying the Hidden World of Microbes », *in: PLOS Biology 5.3*, e82, ISSN: 1545-7885, DOI: 10.1371/JOURNAL.PBIO.0050082.
- Eren, A. Murat, Özcan C. Esen, Christopher Quince, Joseph H. Vineis, Hilary G. Morrison, Mitchell L. Sogin, and Tom O. Delmont (Oct. 2015), « Anvi'o: an advanced analysis and visualization platform for 'omics data », *in: PeerJ 3*, e1319, ISSN: 2167-8359, DOI: 10.7717/peerj.1319.
- Esposito, Alfonso and Matthias Kirschberg (Feb. 2014), « How many 16S-based studies should be included in a metagenomic conference? It may be a matter of etymology », *in: FEMS microbiology letters 351.2*, pp. 145–146, ISSN: 1574-6968, DOI: 10.1111/1574-6968.12375.
- Fischer, Martina, Benjamin Strauch, and Bernhard Y. Renard (July 2017), « Abundance estimation and differential testing on strain level in metagenomics data », *in: Bioinformatics*, vol. 33, 14, Oxford University Press, pp. i124–i132, DOI: 10.1093/bioinformatics/btx237.
- Fleischmann, Robert D. et al. (1995), « Whole-genome random sequencing and assembly of *Haemophilus influenzae* Rd », *in: Science (New York, N. Y.) 269.5223*, pp. 496–512, ISSN: 0036-8075, DOI: 10.1126/SCIENCE.7542800.
- Fouts, Derrick E., Lauren Brinkac, Erin Beck, Jason Inman, and Granger Sutton (Dec. 2012), « PanOCT: automated clustering of orthologs using conserved gene neighborhood for pan-genomic analysis of bacterial strains and closely related species », *in: Nucleic acids research 40.22*, ISSN: 1362-4962, DOI: 10.1093/NAR/GKS757.
- Fraser-Liggett, Claire M. (Dec. 2005), « Insights on biology and evolution from microbial genome sequencing », *in: Genome research 15.12*, pp. 1603–1610, ISSN: 1088-9051, DOI: 10.1101/GR.3724205.

-
- Garrison, Erik (2019), *Untangling graphical pangenomics*.
- (2022), *GitHub - ekg/seqwish: alignment to variation graph inducer*.
- Garrison, Erik, Adam Novak, Glenn Hickey, Jordan Eizenga, Eric Dawson, William Jones, Orion Buske, and Mike Lin (2017), « Sequence variation aware references and read mapping with vg : the variation graph toolkit », *in: bioRxiv*, ISSN: 14779226, DOI: 10.1101/234856.
- Garrison, Erik, Jouni Sirén, et al. (2018), « Variation graph toolkit improves read mapping by representing genetic variation in the reference », *in: Nat Biotechnol* 36.9, pp. 875–879, DOI: 10.1038/nbt.4227.
- Gautreau, Guillaume et al. (Mar. 2020), « PPanGGOLiN: Depicting microbial diversity via a partitioned pangenome graph », *in: PLOS Computational Biology* 16.3, ed. by Christos A. Ouzounis, e1007732, ISSN: 1553-7358, DOI: 10.1371/journal.pcbi.1007732.
- Goodwin, Sara, John D. McPherson, and W. Richard McCombie (May 2016), « Coming of age: ten years of next-generation sequencing technologies », *in: Nature Reviews Genetics* 2016 17:6 17.6, pp. 333–351, ISSN: 1471-0064, DOI: 10.1038/nrg.2016.49.
- Grimont, Patrick A D and François-Xavier Weill (2007), *ANTIGENIC FORMULAE OF THE SALMONELLA SEROVARS*, 9th editio, WHO Collaborating Centre for Reference and Research on Salmonella.
- Handelsman, Jo, Michelle R. Rondon, Sean F. Brady, Jon Clardy, and Robert M. Goodman (Oct. 1998), « Molecular biological access to the chemistry of unknown soil microbes: a new frontier for natural products », *in: Chemistry & Biology* 5.10, R245–R249, ISSN: 1074-5521, DOI: 10.1016/S1074-5521(98)90108-9.
- Heather, James M. and Benjamin Chain (Jan. 2016), « The sequence of sequencers: The history of sequencing DNA », *in: Genomics* 107.1, p. 1, ISSN: 10898646, DOI: 10.1016/J.YGENO.2015.11.003.
- Helgason, Erlendur, Ole Andreas Økstad, Dominique A. Caugant, Henning A. Johansen, Agnes Fouet, Michèle Mock, Ida Hegna, and Anne Brit Kolstø (June 2000), « *Bacillus anthracis*, *Bacillus cereus*, and *Bacillus thuringiensis*—one species on the basis of genetic evidence », *in: Applied and environmental microbiology* 66.6, pp. 2627–2630, ISSN: 0099-2240, DOI: 10.1128/AEM.66.6.2627-2630.2000.
- Hughenoltz, Philip, Brett M. Goebel, and Norman R. Pace (1998), « Impact of Culture-Independent Studies on the Emerging Phylogenetic View of Bacterial Diversity », *in:*

-
- Journal of Bacteriology* 180.18, p. 4765, ISSN: 00219193, DOI: 10.1128/jb.180.18.4765-4774.1998.
- Huson, Daniel H., Alexander F. Auch, Ji Qi, and Stephan C. Schuster (Mar. 2007), « MEGAN analysis of metagenomic data », *in: Genome research* 17.3, pp. 377–386, ISSN: 1088-9051, DOI: 10.1101/GR.5969107.
- Hyatt, Doug, Gwo Liang Chen, Philip F. LoCascio, Miriam L. Land, Frank W. Larimer, and Loren J. Hauser (Mar. 2010), « Prodigal: Prokaryotic gene recognition and translation initiation site identification », *in: BMC Bioinformatics* 11, p. 119, ISSN: 14712105, DOI: 10.1186/1471-2105-11-119.
- Imelfort, Michael, Donovan Parks, Ben J. Woodcroft, Paul Dennis, Philip Hugenholtz, and Gene W. Tyson (2014), « GroopM: An automated tool for the recovery of population genomes from related metagenomes », *in: PeerJ* 2014.1, ISSN: 21678359, DOI: 10.7717/peerj.603.
- Iqbal, Zamin, Mario Caccamo, Isaac Turner, Paul Flicek, and Gil McVean (Jan. 2012), « De novo assembly and genotyping of variants using colored de Bruijn graphs », *in: Nature Genetics* 2012 44:2 44.2, pp. 226–232, ISSN: 1546-1718, DOI: 10.1038/ng.1028.
- Jain, Chirag, Alexander Dilthey, Sanchit Misra, Haowen Zhang, and Srinivas Aluru (May 2019), « Accelerating Sequence Alignment to Graphs », *in: bioRxiv*, p. 651638, ISSN: 2692-8205, DOI: 10.1101/651638.
- Jain, Chirag, Luis M. Rodriguez-R, Adam M. Phillippy, Konstantinos T. Konstantinidis, and Srinivas Aluru (Dec. 2018), « High throughput ANI analysis of 90K prokaryotic genomes reveals clear species boundaries », *in: Nature Communications* 9.1, pp. 1–8, ISSN: 20411723, DOI: 10.1038/s41467-018-07641-9.
- Jovel, Juan et al. (Apr. 2016), « Characterization of the gut microbiome using 16S or shotgun metagenomics », *in: Frontiers in Microbiology* 7.APR, p. 459, ISSN: 1664302X, DOI: 10.3389/fmicb.2016.00459.
- Kang, Dongwan D., Feng Li, Edward Kirton, Ashleigh Thomas, Rob Egan, Hong An, and Zhong Wang (2019), « MetaBAT 2: an adaptive binning algorithm for robust and efficient genome reconstruction from metagenome assemblies », *in: PeerJ* 7.7, ISSN: 2167-8359, DOI: 10.7717/PEERJ.7359.
- Katsir, Leron, Ruan Zhepu, Alon Piasezky, Jiandong Jiang, Noa Sela, Shiri Freilich, and Ofir Bahar (Jan. 2018), « Genome Sequence of " Candidatus Carsonella ruddii" Strain BT from the Psyllid *Bactericera trigonica* », *in: Genome announcements* 6.4, ISSN: 2169-8287, DOI: 10.1128/GENOMEA.01466-17.

-
- Kiełbasa, Szymon M., Raymond Wan, Kengo Sato, Paul Horton, and Martin C. Frith (Mar. 2011), « Adaptive seeds tame genomic sequence comparison », *in: Genome research* 21.3, pp. 487–493, ISSN: 1549-5469, DOI: 10.1101/GR.113985.110.
- Kim, Daehwan, Joseph M. Paggi, Chanhee Park, Christopher Bennett, and Steven L. Salzberg (Aug. 2019), « Graph-based genome alignment and genotyping with HISAT2 and HISAT-genotype », *in: Nature biotechnology* 37.8, pp. 907–915, ISSN: 1546-1696, DOI: 10.1038/S41587-019-0201-4.
- Kim, Daehwan, Li Song, Florian P. Breitwieser, and Steven L. Salzberg (Dec. 2016), « Centrifuge: rapid and sensitive classification of metagenomic sequences », *in: Genome research* 26.12, pp. 1721–1729, ISSN: 1549-5469, DOI: 10.1101/GR.210641.116.
- Koonin, Eugene V., Kira S. Makarova, and Yuri I. Wolf (July 2021), « Evolution of Microbial Genomics: Conceptual Shifts over a Quarter Century », *in: Trends in Microbiology* 29.7, pp. 582–592, ISSN: 0966-842X, DOI: 10.1016/J.TIM.2021.01.005.
- Koonin, Eugene V. and Yuri I. Wolf (2008), « Genomics of bacteria and archaea: the emerging dynamic view of the prokaryotic world », *in: Nucleic acids research* 36.21, pp. 6688–6719, ISSN: 1362-4962, DOI: 10.1093/NAR/GKN668.
- Kultima, Jens Roat et al. (Oct. 2012), « MOCAT: A Metagenomics Assembly and Gene Prediction Toolkit », *in: PLOS ONE* 7.10, e47656, ISSN: 1932-6203, DOI: 10.1371/JOURNAL.PONE.0047656.
- Langmead, Ben and Steven L. Salzberg (Apr. 2012), « Fast gapped-read alignment with Bowtie 2 », *in: Nature methods* 9.4, p. 357, ISSN: 15487091, DOI: 10.1038/NMETH.1923.
- Le Chatelier, Emmanuelle et al. (Aug. 2013), « Richness of human gut microbiome correlates with metabolic markers », *in: Nature* 2013 500:7464 500.7464, pp. 541–546, ISSN: 1476-4687, DOI: 10.1038/nature12506.
- Lefébure, Tristan and Michael J. Stanhope (May 2007), « Evolution of the core and pan-genome of *Streptococcus*: positive selection, recombination, and genome composition », *in: Genome biology* 8.5, ISSN: 1474-760X, DOI: 10.1186/GB-2007-8-5-R71.
- Letcher, Brice, Martin Hunt, and Zamin Iqbal (Dec. 2021), « Gramtools enables multiscale variation analysis with genome graphs », *in: Genome Biology* 22.1, pp. 1–27, ISSN: 1474760X, DOI: 10.1186/S13059-021-02474-0/FIGURES/9.
- Li, Heng (Sept. 2018), « Minimap2: pairwise alignment for nucleotide sequences », *in: Bioinformatics* 34.18, ed. by Inanc Birol, pp. 3094–3100, ISSN: 1367-4803, DOI: 10.1093/bioinformatics/bty191.

-
- Li, Heng and Richard Durbin (July 2009), « Fast and accurate short read alignment with Burrows–Wheeler transform », *in: Bioinformatics* 25.14, pp. 1754–1760, ISSN: 1367-4803, DOI: 10.1093/BIOINFORMATICS/BTP324.
- Li, Heng, Xiaowen Feng, and Chong Chu (Dec. 2020), « The design and construction of reference pangenome graphs with minigraph », *in: Genome Biology* 21.1, pp. 1–19, ISSN: 1474760X, DOI: 10.1186/s13059-020-02168-z.
- Li, Junhua et al. (2014), « An integrated catalog of reference genes in the human gut microbiome », *in: Nature Biotechnology* 32.8, pp. 834–841, ISSN: 15461696, DOI: 10.1038/nbt.2942.
- Li, Ruiqiang et al. (Feb. 2010), « De novo assembly of human genomes with massively parallel short read sequencing », *in: Genome research* 20.2, pp. 265–272, ISSN: 1549-5469, DOI: 10.1101/GR.097261.109.
- Li, W. and A. Godzik (July 2006), « Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences », *in: Bioinformatics* 22.13, pp. 1658–1659, ISSN: 1367-4803, DOI: 10.1093/bioinformatics/bt1158.
- Li, Xin, Haiyan Hu, and Xiaoman Li (Aug. 2020), « mixtureS: a novel tool for bacterial strain genome reconstruction from reads », *in: Bioinformatics*, ed. by Inanc Birol, ISSN: 1367-4803, DOI: 10.1093/bioinformatics/btaa728.
- Lindgreen, Stinus, Karen L. Adair, and Paul P. Gardner (Jan. 2016), « An evaluation of the accuracy and speed of metagenome analysis tools », *in: Scientific Reports* 2016 6:1 6.1, pp. 1–14, ISSN: 2045-2322, DOI: 10.1038/srep19233.
- Liu, Bo, Hongzhe Guo, Michael Brudno, and Yadong Wang (Nov. 2016), « deBGA: read alignment with de Bruijn graph-based seed and extension », *in: Bioinformatics (Oxford, England)* 32.21, pp. 3224–3232, ISSN: 1367-4811, DOI: 10.1093/BIOINFORMATICS/BTW371.
- Logan, Niall A. (2009), *Bacterial Systematics*, London: Blackwell Scientific Publications, pp. 1–263, ISBN: 9781444313949, DOI: 10.1002/9781444313949.
- Loman, Nicholas J. et al. (Apr. 2013), « A Culture-Independent Sequence-Based Metagenomics Approach to the Investigation of an Outbreak of Shiga-Toxigenic *Escherichia coli* O104:H4 », *in: JAMA* 309.14, p. 1502, ISSN: 0098-7484, DOI: 10.1001/jama.2013.3231.
- Lu, Jennifer, Florian P. Breitwieser, Peter Thielen, and Steven L. Salzberg (Jan. 2017), « Bracken: Estimating species abundance in metagenomics data », *in: PeerJ Computer Science* 2017.1, e104, ISSN: 23765992, DOI: 10.7717/PEERJ-CS.104/SUPP-5.

-
- Luo, Chengwei, Rob Knight, Heli Siljander, Mikael Knip, Ramnik J Xavier, and Dirk Gevers (2015), « ConStrains identifies microbial strains in metagenomic datasets », *in*: DOI: 10.1038/nbt.3319.
- Madigan, MT, JM Martinko, KS Bender, DH Buckley, DA Stahl, and T Brock (2014), *Brock biology of microorganisms*, 14th ed., Boston: Benjamin Cummings.
- Mahmoud, Medhat, Nastassia Gobet, Diana Ivette Cruz-Dávalos, Ninon Mounier, Christophe Dessimoz, and Fritz J. Sedlazeck (Nov. 2019), « Structural variant calling: The long and the short of it », *in*: *Genome Biology* 20.1, pp. 1–14, ISSN: 1474760X, DOI: 10.1186/S13059-019-1828-7/TABLES/2.
- Makarova, Kira S., Alexander V. Sorokin, Pavel S. Novichkov, Yuri I. Wolf, and Eugene V. Koonin (Nov. 2007), « Clusters of orthologous genes for 41 archaeal genomes and implications for evolutionary genomics of archaea », *in*: *Biology Direct* 2, p. 33, ISSN: 17456150, DOI: 10.1186/1745-6150-2-33.
- Marchesi, Julian R. et al. (Feb. 2016), « The gut microbiota and host health: A new clinical frontier », *in*: *Gut* 65.2, pp. 330–339, ISSN: 14683288, DOI: 10.1136/gutjnl-2015-309990.
- Margulies, Marcel et al. (July 2005), « Genome sequencing in microfabricated high-density picolitre reactors », *in*: *Nature* 2005 437:7057 437.7057, pp. 376–380, ISSN: 1476-4687, DOI: 10.1038/nature03959.
- McCarroll, Steven A. and David M. Altshuler (2007), « Copy-number variation and association studies of human disease », *in*: *Nature genetics* 39.7 Suppl, S37–S42, ISSN: 1061-4036, DOI: 10.1038/NG2080.
- McInerney, James O., Alan McNally, and Mary J. O’Connell (Mar. 2017), « Why prokaryotes have pangenomes », *in*: *Nature Microbiology* 2017 2:4 2.4, pp. 1–5, ISSN: 2058-5276, DOI: 10.1038/nmicrobiol.2017.40.
- Medini, Duccio, Claudio Donati, Hervé Tettelin, Vega Masignani, and Rino Rappuoli (Dec. 2005), « The microbial pan-genome », *in*: *Current opinion in genetics & development* 15.6, pp. 589–594, ISSN: 0959-437X, DOI: 10.1016/J.GDE.2005.09.006.
- Meslier, Victoria et al. (July 2020), « Mediterranean diet intervention in overweight and obese subjects lowers plasma cholesterol and causes changes in the gut microbiome and metabolome independently of energy intake », *in*: *Gut* 69.7, pp. 1258–1268, ISSN: 1468-3288, DOI: 10.1136/GUTJNL-2019-320438.

-
- Mills, Ryan E. et al. (Feb. 2011), « Mapping copy number variation by population scale genome sequencing », *in: Nature* 470.7332, p. 59, ISSN: 14764687, DOI: 10.1038/NATURE09708.
- Minkin, Ilia and Paul Medvedev (Dec. 2020), « Scalable multiple whole-genome alignment and locally collinear block construction with SibeliaZ », *in: Nature Communications* 2020 11:1 11.1, pp. 1–11, ISSN: 2041-1723, DOI: 10.1038/s41467-020-19777-8.
- Minkin, Ilia, Son Pham, and Paul Medvedev (Dec. 2017), « TwoPaCo: an efficient algorithm to build the compacted de Bruijn graph from many complete genomes », *in: Bioinformatics* 33.24, pp. 4024–4032, ISSN: 1367-4803, DOI: 10.1093/BIOINFORMATICS/BTW609, arXiv: 1602.05856.
- Mirkin, Sergei M. (June 2007), « Expandable DNA repeats and human disease », *in: Nature* 2007 447:7147 447.7147, pp. 932–940, ISSN: 1476-4687, DOI: 10.1038/nature05977.
- Myers, E. W. (Sept. 2005), « The fragment assembly string graph », *in: Bioinformatics* 21.Suppl 2, pp. ii79–ii85, ISSN: 1367-4803, DOI: 10.1093/bioinformatics/bti1114.
- Na, Joong Chae, Hyunjoon Kim, Heejin Park, Thierry Lecroq, Martine Léonard, Laurent Mouchard, and Kunsoo Park (July 2016), « FM-index of alignment: A compressed index for similar strings », *in: Theoretical Computer Science* 638, pp. 159–170, ISSN: 03043975, DOI: 10.1016/j.tcs.2015.08.008.
- Needleman, Saul B. and Christian D. Wunsch (Mar. 1970), « A general method applicable to the search for similarities in the amino acid sequence of two proteins », *in: Journal of molecular biology* 48.3, pp. 443–453, ISSN: 0022-2836, DOI: 10.1016/0022-2836(70)90057-4.
- New, Felicia N. and Ilana L. Brito (Sept. 2020), *What Is Metagenomics Teaching Us, and What Is Missed?*, DOI: 10.1146/annurev-micro-012520-072314.
- Nielsen, H. Bjørn et al. (2014), « Identification and assembly of genomes and genetic elements in complex metagenomic samples without using reference genomes », *in: Nature Biotechnology*, ISSN: 15461696, DOI: 10.1038/nbt.2939.
- Noguchi, Hideki, Jungho Park, and Toshihisa Takagi (Nov. 2006), « MetaGene: prokaryotic gene finding from environmental genome shotgun sequences », *in: Nucleic Acids Research* 34.19, p. 5623, ISSN: 03051048, DOI: 10.1093/NAR/GKL723.
- Oh, Julia, Allyson L. Byrd, Morgan Park, Heidi H. Kong, and Julia A. Segre (May 2016), « Temporal Stability of the Human Skin Microbiome », *in: Cell* 165.4, pp. 854–866, ISSN: 1097-4172, DOI: 10.1016/J.CELL.2016.04.008.

-
- Olm, Matthew R., Alexander Crits-Christoph, Keith Bouma-Gregson, Brian A. Firek, Michael J. Morowitz, and Jillian F. Banfield (Jan. 2021), « inStrain profiles population microdiversity from metagenomic data and sensitively detects shared microbial strains », *in: Nature Biotechnology*, pp. 1–10, ISSN: 15461696, DOI: 10.1038/s41587-020-00797-0.
- Oñate Plaza, Florian, Emmanuelle Le Chatelier, Mathieu Almeida, Alessandra C.L. Cervino, Franck Gauthier, Frédéric Magoulès, S. Dusko Ehrlich, and Matthieu Pichaud (May 2019), « MSPminer: Abundance-based reconstitution of microbial pan-genomes from shotgun metagenomic data », *in: Bioinformatics* 35.9, pp. 1544–1552, ISSN: 14602059, DOI: 10.1093/bioinformatics/bty830.
- Ondov, Brian D., Todd J. Treangen, Páll Melsted, Adam B. Mallonee, Nicholas H. Bergman, Sergey Koren, and Adam M. Phillippy (June 2016), « Mash: Fast genome and metagenome distance estimation using MinHash », *in: Genome Biology* 17.1, pp. 1–14, ISSN: 1474760X, DOI: 10.1186/S13059-016-0997-X/FIGURES/5.
- Ott, Michael, Jaroslaw Zola, Srinivas Aluru, and Alexandras Stamatakis (2007), « Large-scale Maximum Likelihood-based phylogenetic analysis on the IBM Bluegene/L », *in: Proceedings of the 2007 ACM/IEEE Conference on Supercomputing, SC'07*, DOI: 10.1145/1362622.1362628.
- Ounit, Rachid, Steve Wanamaker, Timothy J. Close, and Stefano Lonardi (Mar. 2015), « CLARK: fast and accurate classification of metagenomic and genomic sequences using discriminative k-mers », *in: BMC Genomics* 16.1, pp. 1–13, ISSN: 14712164, DOI: 10.1186/S12864-015-1419-2/TABLES/5.
- Outten, Joseph and Andrew Warren (July 2021), « Methods and Developments in Graphical Pangenomics », *in: Journal of the Indian Institute of Science* 101.3, pp. 485–498, ISSN: 00194964, DOI: 10.1007/S41745-021-00255-Z/FIGURES/2.
- Page, Andrew J. et al. (Nov. 2015), « Roary: rapid large-scale prokaryote pan genome analysis », *in: Bioinformatics* 31.22, pp. 3691–3693, ISSN: 1367-4803, DOI: 10.1093/BIOINFORMATICS/BTV421.
- Paten, Benedict, Jordan M. Eizenga, Yohei M. Rosen, Adam M. Novak, Erik Garrison, and Glenn Hickey (July 2018), « Superbubbles, Ultrabubbles, and Cacti », *in: Journal of Computational Biology*, vol. 25, 7, Mary Ann Liebert Inc., pp. 649–663, DOI: 10.1089/cmb.2017.0251.
- Paten, Benedict, Adam M. Novak, Jordan M. Eizenga, and Erik Garrison (2017), *Genome graphs and the evolution of genome inference*, DOI: 10.1101/gr.214155.116.

-
- Perna, Nicole T. et al. (Jan. 2001), « Genome sequence of enterohaemorrhagic *Escherichia coli* O157:H7 », *in: Nature 2001 409:6819* 409.6819, pp. 529–533, ISSN: 1476-4687, DOI: 10.1038/35054089.
- Qian, Jia, Davide Marchiori, and Matteo Comin (Feb. 2017), « Fast and Sensitive Classification of Short Metagenomic Reads with SKraken », *in: Communications in Computer and Information Science* 881, pp. 212–226, ISSN: 18650929, DOI: 10.1007/978-3-319-94806-5_12.
- Qin, Junjie et al. (Mar. 2010), « A human gut microbial gene catalogue established by metagenomic sequencing. », *in: Nature* 464.7285, pp. 59–65, ISSN: 1476-4687, DOI: 10.1038/nature08821.
- Quince, Christopher, Tom O. Delmont, Sébastien Raguideau, Johannes Alneberg, Aaron E. Darling, Gavin Collins, and A. Murat Eren (Dec. 2017), « DESMAN: a new tool for de novo extraction of strains from metagenomes », *in: Genome Biology* 18.1, p. 181, ISSN: 1474-760X, DOI: 10.1186/s13059-017-1309-9.
- Ranjan, Ravi, Asha Rani, Ahmed Metwally, Halvor S. McGee, and David L. Perkins (Jan. 2016), « Analysis of the microbiome: Advantages of whole genome shotgun versus 16S amplicon sequencing », *in: Biochemical and biophysical research communications* 469.4, p. 967, ISSN: 10902104, DOI: 10.1016/J.BBRC.2015.12.083.
- Rappé, Michael S. and Stephen J. Giovannoni (2003), « The uncultured microbial majority », *in: Annual review of microbiology* 57, pp. 369–394, ISSN: 0066-4227, DOI: 10.1146/ANNUREV.MICRO.57.030502.090759.
- Rasko, David A. et al. (Oct. 2008), « The pangenome structure of *Escherichia coli*: Comparative genomic analysis of *E. coli* commensal and pathogenic isolates », *in: Journal of Bacteriology* 190.20, pp. 6881–6893, ISSN: 00219193, DOI: 10.1128/JB.00619-08.
- Rautiainen, Mikko and Tobias Marschall (Sept. 2020), « GraphAligner: rapid and versatile sequence-to-graph alignment », *in: Genome biology* 21.1, p. 253, ISSN: 1474760X, DOI: 10.1186/S13059-020-02157-2/FIGURES/11.
- Rideout, Jai Ram et al. (Aug. 2014), « Subsampled open-reference clustering creates consistent, comprehensive OTU definitions and scales to billions of sequences », *in: PeerJ* 2014.1, e545, ISSN: 21678359, DOI: 10.7717/PEERJ.545/SUPP-1.
- Roosaare, Märt et al. (2017), « StrainSeeker: fast identification of bacterial strains from raw sequencing reads using user-provided guide trees. », *in: PeerJ* 5, e3353, ISSN: 2167-8359, DOI: 10.7717/peerj.3353.

-
- Rossellö -Mora, Ramon and Rudolf Amann (Jan. 2001), « The species concept for prokaryotes », *in: FEMS Microbiology Reviews* 25.1, pp. 39–67, ISSN: 0168-6445, DOI: 10.1111/J.1574-6976.2001.TB00571.X.
- Rothberg, Jonathan M. et al. (July 2011), « An integrated semiconductor device enabling non-optical genome sequencing », *in: Nature* 2011 475:7356 475.7356, pp. 348–352, ISSN: 1476-4687, DOI: 10.1038/nature10242.
- Sahl, Jason W., J. Gregory Caporaso, David A. Rasko, and Paul Keim (2014), « The large-scale blast score ratio (LS-BSR) pipeline: A method to rapidly compare genetic content between bacterial genomes », *in: PeerJ* 2014.1, ISSN: 21678359, DOI: 10.7717/PEERJ.332/SUPP-4.
- Schneiker, Susanne et al. (Nov. 2007), « Complete genome sequence of the myxobacterium *Sorangium cellulosum* », *in: Nature biotechnology* 25.11, pp. 1281–1289, ISSN: 1087-0156, DOI: 10.1038/NBT1354.
- Scholz, Matthias et al. (May 2016), « Strain-level microbial epidemiology and population genomics from shotgun metagenomics », *in: Nature Methods* 13.5, pp. 435–438, ISSN: 15487105, DOI: 10.1038/nmeth.3802.
- Segata, Nicola, Levi Waldron, Annalisa Ballarini, Vagheesh Narasimhan, Olivier Jousson, and Curtis Huttenhower (Aug. 2012), « Metagenomic microbial community profiling using unique clade-specific marker genes », *in: Nature methods* 9.8, p. 811, ISSN: 15487091, DOI: 10.1038/NMETH.2066.
- Shah, Neethu, Haixu Tang, Thomas G. Doak, and Yuzhen Ye (2011), « Comparing bacterial communities inferred from 16S rRNA gene sequencing and shotgun metagenomics », *in: Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, pp. 165–176, ISSN: 2335-6936, DOI: 10.1142/9789814335058_0018.
- Shanson, D.C. (1989), « Classification and pathogenicity of microbes », *in: Microbiology in Clinical Practice*, p. 3, DOI: 10.1016/B978-0-7236-1403-6.50010-7.
- Sibbesen, Jonas Andreas, Lasse Maretty, and Anders Krogh (July 2018), « Accurate genotyping across variant classes and lengths using variant graphs », *in: Nature genetics* 50.7, pp. 1054–1059, ISSN: 1546-1718, DOI: 10.1038/S41588-018-0145-5.
- Siekaniec, Grégoire, Emeline Roux, Téo Lemane, Eric Guédon, and Jacques Nicolas (Nov. 2021), « Identification of isolated or mixed strains from long reads: a challenge met on *Streptococcus thermophilus* using a MinION sequencer », *in: Microbial genomics* 7.11, ISSN: 2057-5858, DOI: 10.1099/MGEN.0.000654.

-
- Sirén, Jouni (Apr. 2016), « Indexing Variation Graphs », *in: Proceedings of the Workshop on Algorithm Engineering and Experiments* 0, pp. 13–27, DOI: 10.1137/1.9781611974768.2, arXiv: 1604.06605v4.
- Sirén, Jouni et al. (Dec. 2020), « Genotyping common, large structural variations in 5,202 genomes using pangenomes, the Giraffe mapper, and the vg toolkit », *in: bioRxiv*, p. 2020.12.04.412486, DOI: 10.1101/2020.12.04.412486.
- Smith, T. F. and M. S. Waterman (Mar. 1981), « Identification of common molecular subsequences », *in: Journal of Molecular Biology* 147.1, pp. 195–197, ISSN: 0022-2836, DOI: 10.1016/0022-2836(81)90087-5.
- Solé, Cristina et al. (Jan. 2021), « Alterations in Gut Microbiome in Cirrhosis as Assessed by Quantitative Metagenomics: Relationship With Acute-on-Chronic Liver Failure and Prognosis », *in: Gastroenterology* 160.1, 206–218.e13, ISSN: 15280012, DOI: 10.1053/j.gastro.2020.08.054.
- Staley, J. T. and A. Konopka (Nov. 2003), « MEASUREMENT OF IN SITU ACTIVITIES OF NONPHOTOSYNTHETIC MICROORGANISMS IN AQUATIC AND TERRESTRIAL HABITATS », *in: https://doi.org/10.1146/annurev.mi.39.100185.001541* 39, pp. 321–346, ISSN: 00664227, DOI: 10.1146/ANNUREV.MI.39.100185.001541.
- Staley, JT and NR Krieg (1984), *Classification of procaryote organisms: an overview*, Krieg NR, Baltimore: Bergey's Manual of systematic bacteriology, pp. 1–4.
- Stankiewicz, Pawel and James R. Lupski (Feb. 2010), « Structural variation in the human genome and its role in disease », *in: Annual review of medicine* 61, pp. 437–455, ISSN: 1545-326X, DOI: 10.1146/ANNUREV-MED-100708-204735.
- Sunagawa, Shinichi et al. (May 2015), « Structure and function of the global ocean microbiome », *in: Science* 348.6237, ISSN: 10959203, DOI: 10.1126/science.1261359.
- Tap, Julien et al. (Dec. 2021), « Diet and gut microbiome interactions of relevance for symptoms in irritable bowel syndrome », *in: Microbiome* 9.1, pp. 1–13, ISSN: 20492618, DOI: 10.1186/S40168-021-01018-9/FIGURES/4.
- Tenaillon, Olivier, David Skurnik, Bertrand Picard, and Erick Denamur (Mar. 2010), *The population genetics of commensal Escherichia coli*, DOI: 10.1038/nrmicro2298.
- Tenover, F. C., R. D. Arbeit, R. V. Goering, P. A. Mickelsen, B. E. Murray, D. H. Persing, and B. Swaminathan (1995), « Interpreting chromosomal DNA restriction patterns produced by pulsed-field gel electrophoresis: criteria for bacterial strain typing », *in: Journal of clinical microbiology* 33.9, pp. 2233–2239, ISSN: 0095-1137, DOI: 10.1128/JCM.33.9.2233-2239.1995.

-
- Tessler, Michael et al. (July 2017), « Large-scale differences in microbial biodiversity discovery between 16S amplicon and shotgun sequencing », *in: Scientific Reports 2017 7:1 7.1*, pp. 1–14, ISSN: 2045-2322, DOI: 10.1038/s41598-017-06665-3.
- Tettelin, Hervé et al. (Sept. 2005), « Genome analysis of multiple pathogenic isolates of *Streptococcus agalactiae*: Implications for the microbial “pan-genome” », *in: Proceedings of the National Academy of Sciences 102.39*, pp. 13950–13955, ISSN: 0027-8424, DOI: 10.1073/PNAS.0506758102.
- Thorpe, Harry A., Sion C. Bayliss, Laurence D. Hurst, and Edward J. Feil (May 2017), « Comparative analyses of selection operating on nontranslated intergenic regions of diverse bacterial species », *in: Genetics 206.1*, pp. 363–376, ISSN: 19432631, DOI: 10.1534/genetics.116.195784.
- Truong, Duy Tin, Adrian Tett, Edoardo Pasolli, Curtis Huttenhower, and Nicola Segata (Apr. 2017), « Microbial strain-level population structure & genetic diversity from metagenomes », *in: Genome Research 27.4*, pp. 626–638, ISSN: 15495469, DOI: 10.1101/gr.216242.116.
- Valouev, Anton et al. (July 2008), « A high-resolution, nucleosome position map of *C. elegans* reveals a lack of universal sequence-dictated positioning », *in: Genome research 18.7*, pp. 1051–1063, ISSN: 1088-9051, DOI: 10.1101/GR.076463.108.
- Van Rossum, Thea, Pamela Ferretti, Oleksandr M. Maistrenko, and Peer Bork (June 2020), « Diversity within species: interpreting strains in microbiomes », *in: Nature Reviews Microbiology 2020 18:9 18.9*, pp. 491–506, ISSN: 1740-1534, DOI: 10.1038/s41579-020-0368-1.
- VanVinh, Le, Tran Van Lang, Le Thanh Binh, and Tran Van Hoai (Jan. 2015), « A two-phase binning algorithm using l-mer frequency on groups of non-overlapping reads », *in: Algorithms for Molecular Biology 10.1*, pp. 1–12, ISSN: 17487188, DOI: 10.1186/S13015-014-0030-4/FIGURES/6.
- Vieira-Silva, Sara et al. (May 2020), « Statin therapy is associated with lower prevalence of gut microbiota dysbiosis », *in: Nature 581.7808*, pp. 310–315, ISSN: 14764687, DOI: 10.1038/s41586-020-2269-x.
- Voskoboynik, Ayelet et al. (July 2013), « The genome sequence of the colonial chordate, *Botryllus schlosseri* », *in: eLife 2013.2*, ISSN: 2050084X, DOI: 10.7554/ELIFE.00569.
- Wang, Yi, Henry C.M. Leung, S. M. Yiu, and Francis Y.L. Chin (Sept. 2012), « Meta-Cluster 5.0: a two-round binning approach for metagenomic data for low-abundance

-
- species in a noisy sample », *in: Bioinformatics* 28.18, pp. i356–i362, ISSN: 1367-4803, DOI: 10.1093/BIOINFORMATICS/BTS397.
- Wen, Chengping et al. (July 2017), « Quantitative metagenomics reveals unique gut microbiome biomarkers in ankylosing spondylitis », *in: Genome Biology* 18.1, pp. 1–13, ISSN: 1474760X, DOI: 10.1186/S13059-017-1271-6/FIGURES/4.
- Wenger, Aaron M. et al. (Aug. 2019), « Accurate circular consensus long-read sequencing improves variant detection and assembly of a human genome », *in: Nature Biotechnology* 2019 37:10 37.10, pp. 1155–1162, ISSN: 1546-1696, DOI: 10.1038/s41587-019-0217-9.
- Westcott, Sarah L. and Patrick D. Schloss (Dec. 2015), « De novo clustering methods outperform reference-based methods for assigning 16S rRNA gene sequences to operational taxonomic units », *in: PeerJ* 2015.12, e1487, ISSN: 21678359, DOI: 10.7717/PEERJ.1487/FIG-5.
- Woese, C.R. (1992), *Prokaryote systematics: the evolution of a science*.
- Wood, Derrick E., Jennifer Lu, and Ben Langmead (Nov. 2019), « Improved metagenomic analysis with Kraken 2 », *in: Genome Biology* 20.1, pp. 1–13, ISSN: 1474760X, DOI: 10.1186/S13059-019-1891-0/FIGURES/2.
- Wood, Derrick E. and Steven L. Salzberg (Mar. 2014), « Kraken: Ultrafast metagenomic sequence classification using exact alignments », *in: Genome Biology* 15.3, pp. 1–12, ISSN: 1474760X, DOI: 10.1186/GB-2014-15-3-R46/FIGURES/5.
- Wu, Yu Wei and Yuzhen Ye (Mar. 2011), « A novel abundance-based algorithm for binning metagenomic sequences using l-tuples », *in: Journal of computational biology : a journal of computational molecular cell biology* 18.3, pp. 523–534, ISSN: 1557-8666, DOI: 10.1089/CMB.2010.0245.
- Zhang, Zheng, Scott Schwartz, Lukas Wagner, and Webb Miller (2000), « A greedy algorithm for aligning DNA sequences », *in: Journal of computational biology : a journal of computational molecular cell biology* 7.1-2, pp. 203–214, ISSN: 1066-5277, DOI: 10.1089/10665270050081478.
- Zhao, Yongbing, Jiayan Wu, Junhui Yang, Shixiang Sun, Jingfa Xiao, and J. Yu (Feb. 2012), « PGAP: pan-genomes analysis pipeline », *in: Bioinformatics* 28.3, pp. 416–418, ISSN: 1367-4803, DOI: 10.1093/BIOINFORMATICS/BTR655.
- Zheng, Grace X.Y. et al. (Feb. 2016), « Haplotyping germline and cancer genomes with high-throughput linked-read sequencing », *in: Nature Biotechnology* 2016 34:3 34.3, pp. 303–311, ISSN: 1546-1696, DOI: 10.1038/nbt.3432.

Zhong, Chaofang, Chaoyun Chen, Lusheng Wang, and Kang Ning (Jan. 2021), « Integrating pan-genome with metagenome for microbial community profiling », *in: Computational and Structural Biotechnology Journal* 19, pp. 1458–1466, ISSN: 2001-0370, DOI: 10.1016/J.CSBJ.2021.02.021.

SCIENTIFIC CONTRIBUTIONS

Publication

Da Silva Kévin, Pons Nicolas, Berland Magali, Plaza Oñate Florian, Almeida Mathieu, Peterlongo Pierre. StrainFLAIR: strain-level profiling of metagenomic samples using variation graphs. PeerJ. 2021 Aug 23;9:e11884.

Communications

Talks

Journée « Microbiomes » 2020 (October 1st, Rennes, France) - Identification and quantification of strains in a strain mixture using variation graphs

SeqBIM 2019 (December 16th-17th, Marne-la-Vallée, France) - Identification and quantification of strains in a metagenomic sample using variation graphs

JOBIM 2019 (July 2nd-5th, Nantes, France) - From genomics to metagenomics: benchmark of variation graphs (flash talk)

Poster

JOBIM 2019 (July 2nd-5th, Nantes, France) - From genomics to metagenomics: benchmark of variation graphs

LIST OF FIGURES

1.1	Genomics, Pangenomics, and Metagenomics	23
1.2	From a chromosome to a DNA molecule	24
1.3	Haplotypes illustration	26
1.4	Whole-genome shotgun sequencing	28
1.5	Single-end and Paired-end sequencing	31
1.6	Within-species stratification	38
1.7	Schematic representation of pangenomes as Venn diagrams	40
1.8	Gene commonality in prokaryote genomes	41
1.9	In-depth understanding of microbial communities	46
2.1	Construction of the integrated gene catalog	57
2.2	Overview of co-abundance clustering	59
2.3	Simplified model behind MSPminer	60
3.1	Example of variation graph	69
4.1	StrainFLAIR overview.	80
4.2	Illustration of a variation graph structure and colored-paths	83
4.3	Illustration of the multipath-alignment concept and the read attribution process	85
4.4	Proportion of detected specific genes	94
4.5	Experimental relative abundance	99
5.1	Colored-path attributions disambiguation pipeline	105
5.2	Compatibility and incompatibility between reads	115
5.3	Inference of new strains pipeline	116
5.4	Boxplot of the number of reads mapped to each cluster according to the minimum number of new strains inferred for a mixture with no unknown strains	119

5.5	Boxplot of the number of reads mapped to each cluster according to the minimum number of new strains inferred for a mixture with one unknown strain	120
5.6	Boxplot of the number of reads mapped to each cluster according to the minimum number of new strains inferred for a mixture with two unknown strains	122
5.7	Boxplot of the number of reads mapped to each cluster according to the minimum number of new strains inferred for a mixture with three unknown strains	124
A1	Experimental relative abundance compared to relative abundance computed by StrainFLAIR and Kraken2	138

LIST OF TABLES

1.1	Sequencing technologies characteristics	33
3.1	Overview of graphs terminology	64
3.2	Summary of the input and output characteristics of the existing strain-level profiling tools	72
4.1	Distance between each pair of complete genome sequences from eight strains of <i>E. coli</i> as computed by fastANI	90
4.2	Composition of the mixtures described in number of reads simulated and the corresponding coverage (in parentheses)	91
4.3	Reference strains relative abundances expected and computed by StrainFLAIR or Kraken2 for each simulated experiment with variable coverage of the K-12 strain	92
4.4	Reference strain relative abundances expected and computed by StrainFLAIR or Kraken2 for each simulated experiment with variable coverage of the BL21 strain, absent from the reference variation graph	93
4.5	Reference strains relative abundances expected and computed by StrainFLAIR or other tools for each simulated experiment	95
4.6	StrainFLAIR performances on simulated and mock datasets	102
5.1	Reference strains proportion of specific genes detected computed by the first release of StrainFLAIR and the new version (StrainFLAIR-PE) for each simulated experiment	110
5.2	Reference strains relative abundances expected and computed by StrainFLAIR or the new methodology based on paired-end reads (StrainFLAIR-PE) for each simulated experiment	111
5.3	Number of clusters that infers a certain minimum number of new strains, for a mixture with no unknown strain	118
5.4	Number of clusters that infers a certain minimum number of new strains, for a mixture with one unknown strain	120

5.5	Number of clusters that infers a certain minimum number of new strains, for a mixture with two unknown strains	121
5.6	Number of clusters that infers a certain minimum number of new strains, for a mixture with three unknown strains	123
A1	Reference strains relative abundances expected and computed by StrainFLAIR or Kraken2 for each simulated experiment with variable coverage of the K- 12 MG1655 strain	136
A2	Reference strains relative abundances expected and computed by StrainFLAIR or Kraken2 for each simulated experiment with variable coverage of the BL21-DE3 strain, absent from the reference graph	137

Titre : Identification et quantification de souches microbiennes dans des échantillons métagénomiques par utilisation de graphes de variations

Mot clés : Métagénomique, Graphes de variations, Abondances au niveau souche, Mapping

Résumé : Les études actuelles se tournent vers l'utilisation de graphes au lieu de références linéaires afin de représenter plusieurs génomes. En parallèle, calculer les abondances des souches dans des échantillons métagénomiques suscite un intérêt croissant. Cela permettrait de mettre en évidence de nouvelles associations entre souches et phénotypes ouvrant des avancées pour le diagnostic et thérapeutiques. Nous avons développé *StrainFLAIR*, démontrant l'utilisation de graphes de variations dans ce contexte en indexant des séquences génomiques similaires telles que retrouvées entre souches d'une même espèce, et nous proposons de nouvelles solutions algorithmiques afin d'iden-

tifier et quantifier les souches à partir d'un ensemble de génomes séquencés en requêtant le graphe. Nous avons validé notre approche sur des données simulées constituées d'un mélange de souches d'une seule espèce. Les résultats montrent que *StrainFLAIR* a pu identifier les souches présentes dans l'échantillon parmi les références utilisées, détecter la présence de nouvelles souches proches de ces références, et estimer les abondances de ces souches. Nous avons également validé notre approche sur un mock composé de plusieurs espèces et souches. Les résultats montrent à nouveau que *StrainFLAIR* a pu profiler correctement l'échantillon même dans une configuration plus complexe.

Title: Identification and quantification of microbial strains in metagenomic samples using variation graphs

Keywords: Metagenomics, Variation graphs, Strain-level abundances, Read mapping

Abstract: Current studies are shifting from the use of single linear references to graph structures in order to represent multiple genomes. In parallel, resolving strain-level abundances within metagenomic samples is of growing interest for microbiome studies, as it would highlight new associations between strain variants and phenotypes that suggest major steps for diagnostic and therapeutic purposes. We developed *StrainFLAIR* that shows the use of variation graphs in this context by indexing highly similar genomic sequences as found with strains of a species, and we propose novel algorithmic solutions to identify

and quantify strains in a set of sequenced genomes by querying this graph. We validated our approach first on simulated datasets which focused on a mixture of strains from a single species. The results show that *StrainFLAIR* was able to identify the present strains among the existing references, to detect new strains close to the existing references, and to estimate their relative abundances. We also validated *StrainFLAIR* on a mock composed of several species and strains. The results show again *StrainFLAIR*'s ability to profile correctly the sample even in this more complex configuration.